

**Федеральное государственное бюджетное образовательное
учреждение высшего образования
МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)**

На правах рукописи

Тюменцев Юрий Владимирович



**НЕЙРОСЕТЕВОЕ МОДЕЛИРОВАНИЕ
АДАПТИВНЫХ ДИНАМИЧЕСКИХ СИСТЕМ**

Специальность 05.13.01

Системный анализ, управление и обработка информации
(авиационная и ракетно-космическая техника)

ДИССЕРТАЦИЯ

на соискание ученой степени
доктора технических наук

Научный консультант:
доктор технических наук,
профессор Брусов В. С.

Москва, 2016

Оглавление

Основные сокращения	7
Введение	8
1 Моделирование управляемого движения динамических систем и проблема адаптивности	22
1.1 Система и среда ее существования	22
1.1.1 Общее понятие динамической системы	22
1.1.2 Факторы неопределенности при моделировании систем	24
1.2 Классы динамических систем	25
1.2.1 Детерминированные системы	26
1.2.2 Системы с неопределенностями	27
1.2.3 Управляемые системы	28
1.2.4 Адаптивные системы	29
1.2.5 Интеллектуальные системы	30
1.3 Классы внешних сред	31
1.3.1 Регулярные среды	31
1.3.2 Среды с неопределенностями	31
1.3.3 Реагирующие среды	32
1.3.4 Адаптивные среды	32
1.3.5 Интеллектуальные среды	33
1.4 Взаимодействие систем и среды	33
1.4.1 Иерархии систем и внешних сред	33
1.4.2 Системы-комплексы	34
1.5 Поведение и деятельность систем	35
1.6 Общий подход к решению проблемы моделирования динамических систем . .	37
1.6.1 Динамическая система как объект исследования	37
1.6.2 Схема процесса моделирования динамической системы	43
1.6.3 Основные проблемы, требующие решения при формировании модели динамической системы	50

1.7	Проблема адаптивности систем	51
1.7.1	Виды адаптации	51
1.7.2	Общая характеристика проблемы адаптивности систем	55
1.7.3	Варианты схем адаптивного управления	56
1.7.4	Роль моделей в проблеме адаптивного управления	60
1.7.5	Нейросетевая реализация алгоритмов адаптивного моделирования и управления	61
2	Нейросетевой подход к задачам моделирования и управления систем	64
2.1	Порождающий подход к формированию НС-моделей	64
2.1.1	Структура порождающего подхода	64
2.1.2	Сетевое представление функциональных разложений	65
2.1.3	Многоуровневые настраиваемые функциональные разложения	68
2.1.4	Функциональные и нейронные сети	69
2.2	Структурная организация НС-моделей	70
2.2.1	Слоистая структурная организация НС-модели	70
2.2.2	Нейроны как элементы, из которых образована НС	75
2.2.3	Структурная организация нейрона	78
2.2.4	Параметризация примитивов и образуемого ими нейрона	80
2.2.5	Пример структурного описания нейронной сети	88
2.2.6	Формирование подхода к автоматизации структурного синтеза НС-моделей	90
2.3	Формирование обучающих наборов для НС-моделирования динамических систем	94
2.3.1	Специфика формирования обучающих наборов для НС-моделирования динамических систем	94
2.3.2	Прямой подход к формированию обучающих наборов для НС-моделирования динамических систем	95
2.3.3	Непрямой подход к формированию обучающих наборов для НС-моделирования динамических систем	106
2.3.4	Формирование набора тестовых маневров	108
2.3.5	Формирование тестового возбуждающего сигнала	109
2.4	Алгоритмы обучения НС-моделей	118

2.4.1	Проблемы, возникающие при обучения НС-моделей динамических систем	118
2.4.2	Алгоритмы обучения динамических НС-моделей	130
2.5	Адаптивность НС-моделей	137
2.5.1	НС-модели со вставочными подсетями	137
2.5.2	Инкрементное формирование НС-моделей	141
3	Нейросетевое моделирование управляемого движения летательных аппаратов — подход на основе моделей типа «черный ящик»	143
3.1	Нейросетевые эмпирические модели динамических систем (модели типа «черный ящик»)	143
3.1.1	Основные виды моделей	144
3.1.2	Подходы к учету возмущений, действующих на ДС	146
3.2	Нейросетевая модель движения ЛА на основе многослойной нейронной сети .	151
3.2.1	Общая структура НС-модели движения ЛА на основе многослойной нейронной сети	151
3.2.2	Обучение нейросетевой модели движения ЛА в пакетном режиме . . .	154
3.2.3	Обучение нейросетевой модели движения ЛА в режиме реального вре- мени	154
3.3	Оценка работоспособности нейросетевой модели движения ЛА на основе многослойной нейронной сети	155
4	Нейросетевая реализация адаптивных динамических систем применительно к управлению полетом ЛА	165
4.1	Адаптивное управление с эталонной моделью	166
4.1.1	Общая схема адаптивного управления с эталонной моделью	166
4.1.2	Синтез нейроконтроллера для адаптивного управления с эталонной мо- делью	169
4.1.3	Компенсирующий контур в схеме адаптивного управления с эталонной моделью	174
4.1.4	Оценка влияния параметров эталонной модели на работоспособность синтезируемой системы управления	177
4.1.5	Адаптивное управление с эталонной моделью применительно к угловому движению ЛА	179

4.2	Адаптивное управление с прогнозирующей моделью	186
4.2.1	Общая схема адаптивного управления с прогнозирующей моделью . . .	186
4.2.2	Синтез нейроконтроллера для адаптивного управления с прогнозирующей моделью	187
4.2.3	Адаптивное управление с прогнозирующей моделью применительно к угловому движению ЛА	190
4.3	Адаптивное управление с обратной динамикой и инверсной моделью	193
4.3.1	Адаптивное управления на основе метода обратной задачи динамики .	193
4.3.2	Адаптивное управление с инверсной моделью	198
4.3.3	Адаптивное управление с обратной динамикой и инверсной моделью для случая углового движения ЛА	203
4.3.4	Адаптивное управление с обратной динамикой и инверсной моделью для случая пространственного движения ЛА	205
4.4	Адаптивное управление угловым движением ЛА в условиях неопределенности	214
4.4.1	Влияние атмосферной турбулентности на работоспособность адаптивной системы управления продольным движением самолета . . .	214
4.4.2	Адаптация к неопределенности в исходных данных	215
4.4.3	Оценка значимости механизмов адаптации в задаче управления угло- вым движением ЛА	219
5	Нейросетевые полуэмпирические модели управляемых динамических систем	222
5.1	Соотношение между эмпирическими и полуэмпирическими моделями динамических систем	222
5.2	Общая схема процесса формирования полуэмпирических НС-моделей	224
5.3	Основные элементы процесса формирования полуэмпирических НС-моделей .	225
5.3.1	Преобразование исходной теоретической модели в конечно-разностную форму	225
5.3.2	Нейросетевое представление модели динамической системы с дискретным временем	229
5.3.3	Структурная корректировка полуэмпирической НС-модели	233
5.4	Пример формирования полуэмпирической НС-модели	234

5.5	Экспериментальная оценка полуэмпирической модели	239
6	Нейросетевое полуэмпирическое моделирование движения и идентификация характеристик летательных аппаратов	246
6.1	Полуэмпирическое моделирование продольного углового движения маневренного самолета	246
6.2	Идентификация аэродинамических характеристик летательного аппарата . . .	256
	Заключение	268
	Список литературы	272
	Приложения	289
А	Результаты вычислительных экспериментов с адаптивными системами	289
Б	Диаграммы информативности обучающих наборов для задачи моделирования продольного углового движения маневренного самолета	423

Основные сокращения

АДТ	—	аэродинамическая труба
АДХ	—	аэродинамические характеристики
БПЛА	—	беспилотный летательный аппарат
ГЗЛА	—	гиперзвуковой летательный аппарат
ДАУ	—	дифференциально-алгебраическое уравнение
ДС	—	динамическая система
КМ	—	композиционная модель
ЛА	—	летательный аппарат
ММ	—	математическое моделирование
НС	—	нейронная сеть
ОДУ	—	обыкновенное дифференциальное уравнение
ОС	—	особая ситуация
ОУ	—	объект управления
ПМ	—	прогнозирующая модель
РМНК	—	рекуррентный метод наименьших квадратов
САХ	—	средняя аэродинамическая хорда
СКО	—	среднеквадратичная ошибка
СМ	—	сетевая модель
СУВД	—	система управления воздушным движением
ФБ	—	функциональный базис
ФК	—	фильтр Калмана
ФС	—	функциональная сеть
ЭМ	—	эталонная модель
BP	—	Back Propagation
BPTT	—	Back Propagation Through Time
EKF	—	Extended Kalman Filter
GS	—	Gain Scheduling
MLP	—	Multi-Layer Perceptron

MPC	— Model Predictive Control
MRAC	— Model Reference Adaptive Control
MSE	— Mean Squared Error
NARMAX	— Nonlinear AutoRegressive network with Moving Average — and eXogeneous inputs
NARX	— Nonlinear AutoRegressive network with eXogeneous inputs
NASP	— National AeroSpace Plane
RMSE	— Root Mean Squared Error
RTRL	— Real-Time Recurrent Learning
TDL	— Time Delay Line

Введение

1. В ходе полета летательного аппарата (ЛА) возможно возникновение различного рода особых (нештатных) ситуаций (ОС). Эти ОС могут быть связаны, в частности, с выходом из строя аэродинамических управляющих поверхностей, их приводов, других элементов системы управления, а также с различными повреждениями планера ЛА, меняющими аэродинамические характеристики и динамические свойства ЛА (например, частичное разрушение крыла или оперения). В этой ситуации необходимо пытаться не допустить нарушения устойчивости движения ЛА и потери его управляемости. В целом ряде случаев это может быть сделано путем соответствующей корректировки (перестройки) законов управления движением ЛА, реализуемых его системой управления. Такая корректировка, призвана, если только это физически возможно, восстановить характеристики устойчивости и управляемости ЛА до уровня, обеспечивающего безопасное продолжение и завершение полета. Это выполняется за счет реконфигурации системы управления, т. е. перераспределения функций между органами управления, оставшимися исправными, отключения неисправных элементов системы управления, а также за счет оперативного учета изменений, которым подверглись аэродинамические и динамические характеристики ЛА (при повреждении элементов планера ЛА).

2. По степени опасности Нормами летной годности [1, 2] особые ситуации, связанные с отказами и повреждениями различных элементов, устройств и систем ЛА, воздействиями внешней среды и т. п., подразделяются следующим образом:

- усложнение условий полета;

- сложная ситуация;
- аварийная ситуация;
- катастрофическая ситуация.

Усложнение условий полета определяется как ОС, для которой характерно незначительное увеличение психофизиологической нагрузки на экипаж, либо незначительное ухудшение характеристик устойчивости и управляемости ЛА или летных характеристик ЛА. ОС данного вида не приводит к необходимости немедленного или не предусмотренного заранее изменения плана полета, а также не препятствует благополучному завершению полета.

Сложная ситуация представляет собой ОС, для которой характерно заметное повышение психофизиологической нагрузки на экипаж, заметное ухудшение характеристик устойчивости и управляемости ЛА или летных характеристик ЛА. При таких ОС один или несколько параметров полета могут выходить за эксплуатационные ограничения, но без достижения предельно-допустимых значений этих параметров. Одна из основных задач при возникновении ОС типа «сложная ситуация» — не допустить перехода ее в аварийную или катастрофическую ситуацию.

Аварийная ситуация — это ОС, при которой необходимой становится вынужденная посадка ЛА, или такая ситуация, предотвращение перехода которой в катастрофическую связано с высокой психофизиологической нагрузкой на экипаж. Предотвращение перехода аварийной ситуации в катастрофическую требует, как правило, высокого профессионального мастерства экипажа.

Катастрофическая ситуация — такая ОС, при которой предотвращение гибели людей и/или потери ЛА практически невозможно.

Следует подчеркнуть, что границы между перечисленными классами ОС являются размытыми. А именно, одна и та же ситуация для двух экипажей с существенно разным уровнем профессиональной подготовки может попасть в категорию «аварийная ситуация» для экипажа лучше подготовленного, и в категорию «катастрофическая ситуация» — для экипажа, подготовленного слабее. В еще большей степени отнесение возникшей ОС к тому или иному классу зависит от реализуемых на борту ЛА алгоритмов управления. В частности, если алгоритмы управления не включают средств переназначения органов управления, то потеря, например, руля высоты может привести к возникновению катастрофической ОС, если же такое переназначение есть, то потеря руля высоты может быть компенсирована, например, управлением тягой двигателей ЛА.

3. К числу возможных причин ОС относятся [1, 2]:

- отказы и повреждения функциональных систем ЛА (элементов планера ЛА, его силовой установки, систем и оборудования);
- отказы и повреждения технических средств наземного комплекса систем управления воздушным движением (СУВД);
- действия функциональных систем ЛА и/или технических средств наземного комплекса СУВД, неадекватные сложившейся ситуации;
- неблагоприятные воздействия внешней среды;
- ошибочные, т. е. неадекватные сложившейся ситуации, действия экипажа ЛА;
- ошибочные, т. е. неадекватные сложившейся ситуации, действия персонала наземного комплекса средств СУВД;
- проявление неблагоприятных особенностей аэродинамики, устойчивости, управляемости и прочности ЛА.

Перечисленные факторы могут действовать как по-отдельности, так и в разнообразных сочетаниях. Применительно ко всем этим видам факторов, обеспечение требуемого уровня безопасности полета ЛА может быть достигнуто сочетанием двух комплексов мер:

- предотвращение ОС с помощью мер как проектного характера, реализуемых в ходе процесса создания ЛА, так и мер организационного характера, реализуемых при изготовлении ЛА (управление качеством), а также при его эксплуатации;
- если возникновение ОС все же не удалось предотвратить, то адаптация ЛА к новым условиям.

Во втором из этих вариантов речь идет, в первую очередь о том, чтобы не допустить повышение степени опасности ОС, например, перехода сложной ситуации в аварийную и, тем более, аварийной в катастрофическую. Далее, необходимо выполнить такой комплекс мер, который позволил бы либо ликвидировать ОС, если это возможно, либо адаптироваться к ней таким образом, чтобы благополучно завершить полет.

Что касается предупреждения возникновения ОС (путем прогнозирования ОС и принятия соответствующих упреждающих мер), то эффективным средством здесь может быть предотвращение ошибочных действий экипажа за счет постоянного мониторинга параметров полета для недопущения их выхода за эксплуатационные ограничения и, тем более, превышения

предельно допустимых значений. Частично подобного рода механизмы уже реализованы в системах управления ЛА. Например, современные электродистанционные системы управления могут автоматически распознавать отказы датчиков и элементов блока выработки управляющих команд, отключая обнаруженные неисправные элементы [3–5].

Однако средствами этих систем практически невозможно справиться с ситуацией, возникающей при повреждениях органов управления ЛА (рули высоты и направления, элероны и т. п.), или, тем более, при частичном разрушении элементов конструкции ЛА, например, горизонтального оперения. Вследствие этого, зачастую особая ситуация перерастает из аварийной в катастрофическую даже тогда, когда располагаемые силы и моменты вполне позволяют благополучно завершить полет, но возможностей системы управления оказывается недостаточно, чтобы перестроить схему управления применительно к новым условиям. Примеры подобных ситуаций можно найти, например, в [7].

4. Процесс перестройки (корректировки) законов управления движением ЛА, предпринимаемый для парирования ОС, принято именовать реконфигурацией [6–11], в ходе которой осуществляется параметрическая перенастройка этих законов и/или (при необходимости) изменение структуры этих законов¹.

Обеспечение безопасности полета ЛА путем реконфигурации системы управления выдвигает ряд задач, которые не могут быть решены средствами только традиционной диагностики и традиционной теории управления. Это обусловлено, главным образом, высоким уровнем неопределенности в условиях решаемых задач: какие органы управления, элементы системы управления и планера выйдут из строя, в каких комбинациях, каковы при этом будут значения параметров полета, каким из доступных в данный момент способов наиболее целесообразно реагировать на возникшую ОС и т. п.

Перечисленные обстоятельства диктуют необходимость поиска новых классов средств, которые позволили бы решать задачи повышения безопасности полета ЛА с учетом отмеченной выше специфики этих задач.

Естественным инструментом реализации концепции реконфигурации законов управления полетом ЛА является адаптация, обеспечивающая возможность оперативно приспособивать-

¹В ряде работ принято, наряду с реконфигурацией, выделять еще и реструктуризацию, основным содержанием которой является корректировка структуры алгоритма управления. В последующих разделах термин «реконфигурация» включает как параметрическую, так при необходимости и структурную корректировку закона управления ЛА.

ся к изменениям в текущей ситуации за счет изменения параметров и/или структуры используемых законов управления.

5. Возможность существенно повысить безопасность полета и живучесть ЛА — это только часть комплекса серьезных проблем авиационной техники, для решения которых целесообразно привлекать концепцию адаптивности систем. Данный комплекс в целом связан, в первую очередь, с созданием высокоавтоматизированных и роботизированных ЛА, как пилотируемых, так и беспилотных. Для пилотируемых ЛА данная проблема выражается в необходимости создания высокоэффективных систем-помощников для членов экипажа. Однако наиболее серьезно эта проблема стоит для роботизированных беспилотных летательных аппаратов (БПЛА), обладающих высокой степенью самостоятельности (автономности) при решении ими различных целевых задач военного и гражданского характера в самых разнообразных условиях [12–24]. В числе этих задач можно указать следующие:

- разведка и целеуказание;
- ударные задачи (наземные, надводные и подводные цели);
- перехват воздушных целей;
- завоевание превосходства в воздухе;
- транспортировка грузов и снабжение войск;
- спасательные операции на суше и на море;
- работы по ликвидации последствий природных и техногенных катастроф;
- патрулирование, выявление угроз, охрана объектов;
- мониторинг состояния линий электропередач, трубопроводов, лесных массивов;
- аэрофотосъемка, ледовая разведка и разведка рыбных ресурсов;
- выполнение монтажных операций различного рода.

Задачи этих классов решаются, в основном, с помощью пилотируемых летательных аппаратов (самолетов и вертолетов). Применение в данной сфере БПЛА совместно с пилотируемыми ЛА, а в ряде случаев и вместо пилотируемых ЛА, привлекательно в силу следующих причин:

- существенно более высокая относительная доля полезной нагрузки для БПЛА по сравнению с пилотируемыми ЛА за счет отсутствия на борту экипажа, а также средств обеспечения его жизнедеятельности и функционирования;

- возможность реализации существенно более энергичного маневрирования БПЛА в сравнении с пилотируемыми ЛА за счет повышения допустимого уровня располагаемых перегрузок;
- возможность создания малоразмерных летательных аппаратов, существенно менее дорогих в производстве и эксплуатации;
- возможность решать требуемые целевые задачи в условиях, когда непосредственное участие человека нежелательно или недопустимо (радиационная обстановка, неприемлемый уровень риска и т. п.).

6. Применение БПЛА будет по-настоящему эффективным, только если они будут в состоянии решать поставленные перед ними задачи в максимально автономном режиме, т. е. с минимальным участием человека, которое должно в этом случае сводиться, в основном, к постановке задач, контролю их исполнения, а также, при необходимости, к корректировке целей летной операции уже в ходе полета. Данная ситуация обусловлена потенциальной уязвимостью любых радиоканалов связи, требуемых для БПЛА в дистанционно пилотируемых вариантах. Кроме того, в ряде случаев реагирование на усложненную и быстро меняющуюся обстановку может оказаться за пределами психофизиологических возможностей человека-оператора (объем внимания, скорость реагирования).

Таким образом, важнейшее требование к БПЛА состоит в том, что они должны обладать высоким уровнем самостоятельности при решении поставленных перед ними задач. Чтобы соответствовать этим требованиям, роботизированные БПЛА должны уметь:

- достигать поставленных целей в высокодинамичной среде со значительным числом разнородных неопределенностей в ней, с учетом возможного противодействия;
- корректировать поставленные цели, а также формировать новые цели и комплексы целей, исходя из заложенных в систему управления поведением БПЛА ценностных и нормативных установок (мотивации);
- уметь оценивать текущую ситуацию на основе многостороннего восприятия внешней и внутренней среды, уметь формировать прогноз развития ситуации;
- добывать новые знания, накапливать опыт решения разнообразных задач, обучаться на этом опыте, модифицировать свое поведение на основе полученных знаний и накопленного опыта;

- адаптироваться к виду задач, в решении которых возникает необходимость, в том числе обучаться решению задач, не предусмотренных первоначальным проектом системы;
- образовывать группировки, нацеленные на взаимодействие их членов при решении некоторой общей задачи.

7. Решение перечисленных выше задач с помощью роботизированного БПЛА на уровне, не уступающем, по крайней мере, тому уровню, который можно обеспечить с помощью пилотируемых ЛА, требует радикального пересмотра подходов к созданию алгоритмов управления всеми элементами процесса функционирования БПЛА. В робототехнике совокупность всех видов процессов функционирования робота принято именовать поведением робота. Соответственно, имея в виду всё усиливающиеся тенденции роботизации БПЛА, принято говорить о задаче управлением поведением БПЛА как о реализации всех видов его функционирования, необходимых для выполнения перечисленных выше целевых задач. Управление поведением БПЛА включает такие элементы, как:

- планирование летной операции, управление ее выполнением, корректировка плана при изменении ситуации;
- управление движением БПЛА, включая его траекторное движение (в том числе наведение и навигацию) и угловое движение;
- управление решением целевых задач (управление действием наблюдательной и разведывательной аппаратуры, управление применением оружия, управление действием средств выполнения монтажных операций и т.п.);
- управление взаимодействием с другими ЛА, как беспилотными, так и пилотируемыми, при выполнении поставленной задачи группировкой ЛА, в которую входит данный БПЛА.

Алгоритмы управления (формирования управляющих воздействий, принятия решений на управление) в качестве исходных данных должны использовать информацию о целях операции, а также об обстановке, характеризуемой оценками текущей и прогнозируемой ситуации, в которой выполняет задачу БПЛА. Эта ситуация складывается как из внешних составляющих (состояние окружающей среды, состояния и действия партнеров и противников), так и из внутренних (данные по диагностике состояния и оценкам работоспособности конструкции и систем ЛА). Средства получения этой базовой информации также должны включаться в состав комплекса алгоритмов, реализующих требуемое поведение роботизированного БПЛА.

8. Удовлетворение перечисленных выше требований возможно лишь, если наделить системы управления поведением БПЛА развитыми механизмами, позволяющими им приспосабливаться (адаптироваться) к существенно меняющейся ситуации, а также обучаться, извлекать опыт и знания из выполняемой деятельности для использования их в дальнейшем. Подобные механизмы должны обеспечивать решение двух важнейших задач:

- восприятие обстановки с оценкой текущей ситуации и прогнозом ее развития;
- формирование поведения БПЛА как совокупности целенаправленных реакций на текущую и/или прогнозируемую ситуацию.

Реализация этих механизмов обеспечивает возможность создавать адаптивные и интеллектуальные системы управления поведением БПЛА. Использование подобных систем позволит разрабатывать высокоавтономные роботизированные БПЛА, предназначенные для эффективного решения сложных целевых задач, успешно действующие в условиях неопределенности как одиночно, так и в составе группировки. Еще одно важное следствие применения средств адаптивного и интеллектуального управления поведением БПЛА состоит в том, что появляется возможность резко повысить выживаемость ЛА при возникновении серьезных повреждений его конструкции и отказов бортового оборудования.

9. При реализации перечисленных выше функций, как в процессе создания, так и при последующей эксплуатации ЛА различных видов значительное место занимает решение таких классов задач, как анализ поведения динамических систем (ДС), синтез алгоритмов управления для них, идентификация их неизвестных или неточно известных характеристик. Критически важная роль при решении задач этих трех классов принадлежит математическим и компьютерным моделям динамических систем.

Традиционными классами математических моделей для технических систем являются обыкновенные дифференциальные уравнения (для систем с сосредоточенными параметрами) и дифференциальные уравнения в частных производных (для систем с распределенными параметрами). Применительно к управляемым динамическим системам в качестве аппарата моделирования наибольшее распространение получили обыкновенные дифференциальные уравнения. Эти модели, в сочетании с соответствующими численными методами, широко используются при решении задач синтеза и анализа управляемого движения летательных аппаратов различных классов. Аналогичные средства используются также и для моделирования движения динамических систем других видов, включая надводные и подводные аппараты, наземные движущиеся средства и т. п.

Методы формирования и использования моделей традиционного типа к настоящему времени достаточно детально разработаны и с успехом применяются для решения широкого круга задач. Однако применительно к современным и перспективным техническим системам возникает ряд проблем, решение которых не может быть обеспечено традиционными методами. Эти проблемы вызваны наличием разнообразных и многочисленных неопределенностей в свойствах соответствующей системы и в условиях ее функционирования, которые можно парировать, только если рассматриваемая система обладает свойством адаптивности, т. е. имеются средства оперативной подстройки системы и ее модели к меняющейся текущей ситуации. Кроме того, требования по точности моделей, предъявляемые исходя из специфики решаемой прикладной задачи, в ряде случаев превышают возможности традиционных методов.

10. Как показывает имеющийся опыт, аппаратом моделирования, наиболее адекватным указанной ситуации, являются методы и средства, основанные на концепции искусственной нейронной сети. Такой подход можно рассматривать как альтернативу традиционным методам моделирования ДС, обеспечивающую, в том числе, возможность получения адаптивных моделей. При этом традиционные нейросетевые модели ДС, в частности, модели классов NARX и NARMAX, наиболее часто используемые для моделирования управляемых динамических систем, являются чисто эмпирическими (модели типа «черный ящик»), т. е. основываются исключительно на экспериментальных данных о поведении объекта. Однако в задачах того уровня сложности, что типичен для авиационной и ракетно-космической техники, очень часто для такого рода эмпирических моделей не удается добиться требуемого уровня точности, обеспечивающего, например, решение задач управления движением ЛА. Кроме того, в силу особенностей структурной организации таких моделей, они не позволяют решать задачу идентификации характеристик ДС (например, аэродинамических характеристик ЛА), что является серьезным недостатком данного класса моделей.

Один из важнейших источников невысокой эффективности НС-моделей традиционного типа в задачах, связанных со сложными техническими системами, состоит в том, что формируется чисто эмпирическая модель («черный ящик»), которая должна охватывать все нюансы поведения ДС. Для этого приходится строить НС-модель достаточно высокой размерности (т. е. с большим числом настраиваемых параметров в ней). В то же время из опыта нейросетевого моделирования известно, что чем больше размерность НС-модели, тем больший объем обучающих данных требуется для ее настройки. В итоге, при тех объемах эксперименталь-

ных данных, которые реально можно получить для сложных технических систем, не удастся осуществить обучение таких моделей, обеспечивающее заданный уровень их точности.

Чтобы преодолеть указанные затруднения, связанные с традиционными моделями как в виде дифференциальных уравнений, так и в виде НС-моделей, в диссертации вводится комбинированный подход, основу которого составляет НС-моделирование в силу того, что только в этом варианте удастся получить адаптивные модели. Теоретическое знание об объекте моделирования, существующее в виде обыкновенных дифференциальных уравнений (это, например, традиционные модели движения ЛА), вносятся специальным образом в НС-модель комбинированного типа (полуэмпирическую НС-модель). При этом часть НС-модели формируется на основе имеющегося теоретического знания и не требует дальнейшей настройки (обучения). Настройке и/или структурной корректировке в процессе обучения формируемой НС-модели подлежат только те элементы, которые содержат неопределенности, например, аэродинамические характеристики ЛА.

Результатом такого подхода являются полуэмпирические НС-модели, позволяющие решить следующие важные задачи, недоступные традиционным НС-методам: резко снизить размерность НС-модели, что позволяет добиться от нее требуемой точности, используя обучающие наборы, недостаточные по объему для традиционных НС-моделей; обеспечить возможность идентификации характеристик ДС, описываемых нелинейными функциями многих переменных (например, коэффициентов аэродинамических сил и моментов).

11. В последующих разделах рассматривается реализация описанного выше подхода, а также примеры его применения для моделирования движения ЛА и идентификации аэродинамических характеристик ЛА. В рамках этой реализации объект и предмет исследования, а также цель диссертационной работы могут быть охарактеризованы следующим образом:

- *объект исследования* — адаптивные динамические системы, действующие в условиях разнообразных неопределенностей, в частности, ЛА различных классов;
- *предмет исследования* — адаптивные законы управления многомерных нелинейных управляемых ДС, а также адаптивные модели таких ДС, обладающие высокой точностью и быстродействием;
- *цель диссертационной работы* — формирование нового подхода к математическому и компьютерному моделированию адаптивных ДС, свободного от недостатков традиционных подходов.

Формируемый подход должен обеспечивать решение задач анализа поведения, синтеза управления и идентификации неизвестных характеристик ДС, возникающих при создании и эксплуатации сложных технических систем, в частности, роботизированных пилотируемых и беспилотных ЛА.

В процессе выполнения диссертационной работы для достижения сформулированной выше цели *были решены следующие задачи:*

- 1) С позиций общей теории систем построена классификация ДС по их существенным признакам (типология ДС), а также выявлены варианты среды, с которой взаимодействуют ДС, что обеспечивает единый контекст для решения задач анализа поведения, синтеза управления и идентификации характеристик при создании перспективных технических систем.
- 2) Разработан нейросетевой подход к обеспечению адаптивности ДС (в таких элементах как закон управления ДС и модель ДС) за счет ансамблевой архитектуры используемых нейронных сетей (НС), введения в НС вставочных подсетей и использования инкрементного обучения НС.
- 3) Разработан класс математических и компьютерных моделей, представляющих собой НС-модели гибридного типа, обладающие адаптивностью, объединяющие теоретические знания о ДС и экспериментальные данные о ее поведении.
- 4) Разработано унифицированное структурное (модульное) описание НС-моделей, обеспечивающее единообразное представление всех видов статических и динамических сетей, позволяющее автоматизировать процесс синтеза НС-моделей.
- 5) Разработан композиционный подход к синтезу (формированию) НС-моделей, основанный на интерпретации НС-модели как разложения по обобщенному функциональному базису (ФБ специального вида).
- 6) Систематизированы имеющиеся результаты в области моделирования ДС с использованием НС традиционного типа (модели типа «черный ящик»), выявлены ограничения и область возможного использования этих средств.
- 7) Систематизированы и обобщены имеющиеся результаты в области моделирования ДС с использованием моделей типа «серый ящик», выявлены перспективные направления развития данной области.
- 8) Разработаны алгоритмы формирования полуэмпирических НС-моделей как одного из классов моделей типа «серый ящик» на основе теоретического знания (в виде систем

обыкновенных дифференциальных или дифференциально-алгебраических уравнений) об объекте моделирования, а также экспериментальных данных о его поведении.

- 9) Разработаны методы и алгоритмы получения обучающих данных для НС-моделей динамических систем.
- 10) Разработаны алгоритмы параметрической настройки (обучения) полуэмпирических НС-моделей.
- 11) В серии вычислительных экспериментов проведена верификация разработанных полуэмпирических НС-моделей и методов их формирования.
- 12) В серии вычислительных экспериментов проведено формирование эмпирических и полуэмпирических НС-моделей для объектов различных классов, проведено сопоставление свойств этих двух классов моделей.
- 13) Разработан подход к решению задачи идентификации характеристик ДС, описываемых нелинейными функциями многих переменных. Работоспособность этого подхода продемонстрирована на примере задачи идентификации аэродинамических коэффициентов ЛА.
- 14) В рамках предложенного подхода решено значительное число задач адаптивного управления для ЛА различных классов, в которых модель объекта и закон управления были реализованы в виде динамических НС-структур.

При решении этих задач были получены следующие *новые научные результаты*:

- сформирована типология ДС, обеспечивающая единый контекст для решения задач анализа поведения, синтеза управления и идентификации характеристик при создании перспективных технических систем различных классов;
- разработано унифицированное структурное описание НС-моделей, обеспечивающее единообразное представление всех видов статических и динамических сетей, позволяющее автоматизировать процесс синтеза НС-моделей;
- разработан композиционный подход к синтезу статических и динамических НС-моделей, основанный на интерпретации НС-модели как разложения по обобщенному функциональному базису;
- разработан класс гибридных нейросетевых моделей полуэмпирического типа, объединяющий теоретические знания об объекте моделирования и экспериментальные данные о его поведении;

- разработаны алгоритмы формирования гибридных нейросетевых моделей полуэмпирического типа, их структурной корректировки и параметрической настройки;
- разработаны методы и алгоритмы получения обучающих данных для НС-моделей динамических систем;
- разработан подход к решению задачи идентификации характеристик ДС как нелинейных функций многих переменных.

Эти результаты освещены в серии публикаций, в числе которых: монография [170], учебные пособия [171,172], статьи в рецензируемых журналах из списка ВАК РФ [173–195], статьи в рецензируемых сборниках трудов конференций [196–211]. Из числа статей 9 индексированы в международной базе данных Scopus.

12. Получаемые в ходе выполнения диссертационной работы результаты неоднократно докладывались на конференциях и семинарах международного, всероссийского и отраслевого уровня. В их число входят следующие: Экспертные системы реального времени (1995 г.); Динамические интеллектуальные системы (1996 г.); Национальная конференция по искусственному интеллекту «КИИ-96»; Seminar on Recent Research and Design Progress in Aeronautical Engineering and its Influence on Education (RRDPAE, 1996, 1998 гг.); Всероссийская научно-техническая конференция «Проблемы совершенствования робототехнических и интеллектуальных систем летательных аппаратов» (1999 г.); Национальная конференция по искусственному интеллекту (КИИ'2002); International Conference on Scientific Aspects of Unmanned Aerial Vehicle (2004, 2006, 2008, 2010 гг.); IFAC Workshop on Generalized Solutions in Control Problems (GSCP-04, 2004 г.); European Micro Air Vehicle Conference and Flight Competition (2007, 2010, 2011 гг.); International Congress of the Aeronautical Sciences (ICAS, 2010, 2014 гг.); Российская мультиконференция по проблемам управления (2012 г.); Всероссийская научно-техническая конференция «Нейроинформатика» (1999, 2001, 2002, 2010, 2012–2016 гг.); Международная конференция «Авиация и космонавтика» (2006, 2010, 2011–2015 гг.); Всероссийская научная конференция «Нейрокомпьютеры и их применение» (2012, 2014–2016 гг.); Междисциплинарный семинар по сложным системам «Экобионика» (2015 г.); Семинар по теоретическим проблемам нейроинформатики (2015 г.).

Теоретическая и практическая значимость диссертационной работы состоит в том, что предложенный класс моделей ДС, а также методы синтеза и параметрической настройки таких моделей открывают новые возможности для решения задач управления поведением сложных технических систем, включая их роботизированные варианты. Эти методы и средства

позволяют существенно повысить эффективность решения задач анализа поведения, синтеза алгоритмов управления и идентификации характеристик перспективных технических систем.

Практическая значимость полученных результатов состоит в том, что появляется возможность создания перспективных пилотируемых ЛА, а также роботизированных беспилотных ЛА различного назначения, предназначенных для решения сложных целевых задач и комплексов таких задач в условиях неопределенности на основе использования адаптивных и интеллектуальных систем управления поведением БПЛА. Роботизация БПЛА, осуществляемая на этой основе, позволит резко повысить их эффективность при решении целевых задач, а также выживаемость в неблагоприятных условиях.

Полученные результаты могут быть использованы разработчиками перспективных технических систем, включая роботизированные ЛА, при формировании алгоритмов адаптивного и интеллектуального управления поведением этих систем, а также анализа их поведения и решения задач идентификации характеристик ЛА.

1 Моделирование управляемого движения динамических систем и проблема адаптивности

1.1 Система и среда ее существования

1.1.1 Общее понятие динамической системы

1. Динамическая система \mathbb{S} — это система, состояние которой изменяется во времени под воздействием некоторых внешних и/или внутренних факторов [25–34]. Источником внешних факторов является среда \mathcal{E} , в которой действует динамическая система, источник внутренних факторов — особенности устройства данной системы, а также события, происходящие в системе (например, отказы и повреждения, влияющие на динамические свойства системы).

Любая система \mathbb{S} существует не изолированно, она функционирует в некоторой среде \mathcal{E} и во взаимодействии с этой средой². При этом реакция системы на воздействие среды может быть как пассивной³, так и активной⁴.

Динамические системы, активно взаимодействующие со средой, представляют собой управляемые ДС. При этом характер взаимодействия системы со средой зависит от свойств ДС, а также от ресурсов, которыми она располагает. Эти ресурсы можно подразделить на два вида — внутренние и внешние. В случае задачи управления движением ЛА примерами этих ресурсов могут быть следующие:

- внутренние ресурсы — законы управления, цифровая карта местности, средства инерциальной навигации и т. п.;
- внешние ресурсы — системы спутниковой навигации ГЛОНАСС и GPS, средства телеуправления, средства радионавигации и т. п.

²Чаще всего в прикладных задачах принимается во внимание только часть этого взаимодействия, а именно воздействие среды \mathcal{E} на систему \mathbb{S} , например воздействие гравитационного поля и/или атмосферы на летательный аппарат. Однако в ряде случаев необходимо учитывать и вторую половину этого взаимодействия, т. е. воздействие \mathbb{S} на \mathcal{E} , в частности возмущение типа спутного следа за самолетом. Учет такого рода фактора, например, становится важным при рассмотрении задач, связанных с организацией группового поведения ЛА — полет строем.

³Примеры пассивного взаимодействия со средой дают такие задачи, как полет некорректируемого артиллерийского снаряда или авиабомбы, полет неуправляемой ракеты.

⁴При активном взаимодействии со средой в ответ на воздействие среды система согласно некоторому «закону управления» формирует и реализует реакцию на данное воздействие, например отклоняет руль, парируя действие возмущения на ЛА.

Чем меньше система зависит от внешних ресурсов (информационных источников и управляющих команд), тем выше степень ее автономности. Изучение высокоавтономных управляемых систем становится все более актуальным в настоящее время в связи с быстрым развитием беспилотных средств различных классов, в частности беспилотных летательных аппаратов.

2. Будем считать, что система \mathbb{S} полностью определена, если заданы комплекс характеризующих ее величин (состояния, управления, возмущения и т. п.), время и правило для определения следующего состояния системы на основе ее предыдущих состояний. Будем представлять систему \mathbb{S} как упорядоченную тройку

$$\begin{aligned}\mathbb{S} &= \langle X^S, \Phi^S, W^S \rangle, \\ X^S &= \{X_i^S\}, i = 1, \dots, N_S; \quad \Phi^S = \{\Phi_j^S\}, j = 1, \dots, N_R; \\ W^S &= \langle T^S, E^S \rangle, \quad T^S \subseteq T.\end{aligned}\tag{1.1}$$

Здесь $X^S = \{X_i^S\}, i = 1, \dots, N_S$ — набор *структур* системы \mathbb{S} (термин «структура» трактуется здесь в общематематическом смысле, как набор множеств с определенными на них наборами отношений); $\Phi^S = \{\Phi_j^S\}, j = 1, \dots, N_R$ — набор *рескриптов* системы \mathbb{S} (термин «рескрипт» используется здесь как общее наименование для преобразований всех видов — отображений, алгоритмов, процедур вывода и т. п.); $W^S = \langle T^S, E^S \rangle$ — *часы* системы \mathbb{S} , $T^S \subseteq T$ — множество моментов времени функционирования системы («системное время»), T — множество всех возможных моментов времени («мировое время»), наделенное структурой линейного порядка (т. е. упорядоченное отношением \leq), E^S — механизм активности («тактовый генератор») системы \mathbb{S} .

Система-объект \mathbb{S} существует не изолированно, а во взаимодействии с системой-средой \mathcal{E} , представляемой упорядоченной тройкой вида

$$\begin{aligned}\mathcal{E} &= \langle Q^\mathcal{E}, \Psi^\mathcal{E}, W^\mathcal{E} \rangle, \\ Q^\mathcal{E} &= \{Q_i^\mathcal{E}\}, i = 1, \dots, M_\mathcal{E}; \quad \Psi^\mathcal{E} = \{\Psi_j^\mathcal{E}\}, j = 1, \dots, M_R; \\ W^\mathcal{E} &= \langle T^\mathcal{E}, E^\mathcal{E} \rangle, \quad T^\mathcal{E} \subseteq T.\end{aligned}\tag{1.2}$$

Здесь $Q^\mathcal{E} = \{Q_i^\mathcal{E}\}, i = 1, \dots, M_\mathcal{E}$ — набор *структур* среды \mathcal{E} ; $\Psi^\mathcal{E} = \{\Psi_j^\mathcal{E}\}, j = 1, \dots, M_R$ — набор *рескриптов* среды \mathcal{E} ; $W^\mathcal{E} = \langle T^\mathcal{E}, E^\mathcal{E} \rangle$ — часы среды \mathcal{E} , $T^\mathcal{E} \subseteq T$ — множество моментов времени функционирования среды, T — множество всех возможных моментов времени, $E^\mathcal{E}$ — механизм активности среды \mathcal{E} . В дальнейшем вместо термина «система-объект» для краткости будет использоваться термин «объект», а вместо термина «система-среда» — термин «среда» если это не ведет к неправильному толкованию текста.

1.1.2 Факторы неопределенности при моделировании систем

1. Задачу моделирования и управления движением ДС приходится решать в условиях многочисленных разнородных неопределенностей как в модели движения системы \mathbb{S} , так и в условиях окружающей среды \mathcal{E} , в которой ДС функционирует.

Конкретизируем возможный состав факторов неопределенности применительно к летательным аппаратам как одному из важнейших видов ДС. Проведенный анализ позволяет выделить следующие классы факторов неопределенности, характерных для данного случая:

- неопределенности параметрического типа, связанные с величинами, описывающими параметры ЛА (масса ЛА m ; моменты инерции ЛА J_x, J_y, J_z, J_{xy} и т. п.);
- неопределенности функционального типа, относящиеся к характеристикам ЛА (коэффициенты аэродинамических сил c_x, c_y, c_z ; коэффициенты моментов m_x, m_y, m_z ; тяга силовой установки P и т. п.);
- неопределенности, связанные с воздействиями среды (плотность воздуха ρ ; атмосферное давление p ; ветер W ; атмосферная турбулентность и т. п.).

Неопределенности *параметрического* типа для величин m, J_x, J_y, J_z, J_{xy} задаются в интервальном виде, т. е. в форме

$$\lambda \in [\lambda_{min}, \lambda_{max}],$$

где λ — один из параметров m, J_x, J_y, J_z, J_{xy} , λ_{min} и λ_{max} — минимально возможное и максимально возможное значение, соответственно, для параметра λ .

Неопределенности *функционального* типа, относящиеся к характеристикам ЛА, $c_x, c_y, c_z, m_x, m_y, m_z, P$ могут быть заданы в виде параметрических семейств кривых вида

$$\varphi(x) = \varphi(x)_{nom} + \Delta\varphi(x), \quad \Delta\varphi(x) = \Phi(w, x), \quad w_i \in [w_i^{min}, w_i^{max}], \quad i = 1, \dots, n.$$

где $\varphi(x)$ — одна из характеристик ЛА $c_x, c_y, c_z, m_x, m_y, m_z, P$; $\varphi(x)_{nom}$ — некоторое номинальное значение характеристики $\varphi(x)$; $\Delta\varphi(x)$ — отклонение реальной характеристики $\varphi(x)$ от ее номинального значения.

Неопределенности, связанные с воздействиями среды, т. е. плотность воздуха ρ , атмосферное давление p , ветер W , атмосферная турбулентность задаются согласно общепринятым моделям вероятностного характера (см., например, [42, 43]).

2. В процессе полета ЛА могут возникать различного рода нештатные ситуации, вызванные отказами оборудования и систем ЛА, повреждениями его планера и силовой установки.

Часть таких отказов и повреждений оказывает непосредственное влияние на свойства ЛА как объекта управления, вследствие чего возникает проблема корректировки алгоритмов управления ЛА с целью приспособиться к изменившейся динамике ЛА. При этом предусмотреть заранее все вероятные отказы, повреждения и их комбинации не представляется возможным. В связи с этим представляется целесообразным изменения динамических свойств ЛА вследствие отказов и повреждений интерпретировать как еще один класс факторов неопределенности, парирование которых возлагается на механизмы адаптации. Они должны обеспечить отказоустойчивое управление, т. е. такое управление, которое в состоянии приспособиться к изменению динамики объекта управления, порожденному отказами или повреждениями, обеспечивая приемлемое качество управления.

При таком подходе задача разделяется на две части. Первая из них связана с реконфигурацией алгоритмов управления движением ЛА при возникновении нештатной ситуации. Здесь конкретное знание о характере нештатной ситуации как правило не требуется, алгоритмы адаптации (реконфигурации) реагируют непосредственно на отклонение поведения объекта управления от некоторого нормативного, например, задаваемого эталонной моделью. Однако выявить характер и источник возникновения нештатной ситуации все же необходимо, поскольку может потребоваться, например, изолировать отказавшее устройство, чтобы оно не оказывало вредного воздействия на дальнейшее поведение объекта управления (например, выявить привод органа управления, некорректно реагирующий на входной сигнал или датчик, выдающий неверный сигнал).

Обе эти задачи можно решать параллельно, начиная сразу же после выявления факта отказа. Такое выявление можно рассматривать как разновидность задачи наблюдения: контролируется поведение объекта управления, которое сравнивается с его нормативным поведением в данных текущих условиях, задаваемым соответствующей моделью. Если наблюдаемое отклонение достигает некоторого порогового значения, выдается сигнал на запуск процессов диагностики, локализирующих место возникновения отказа и решающих задачу его распознавания (классификации). Одновременно ведут свою работу алгоритмы реконфигурации, корректирующие алгоритмы управления движением для восстановления их работоспособности.

1.2 Классы динамических систем

В данном разделе формируется иерархия систем-объектов как последовательность определений их классов, упорядоченных по уровню их возможностей.

Системы-объекты, рассматриваемые ниже, отличаются друг от друга своими свойствами и, как следствие, уровнем потенциальных возможностей.

В качестве признаков, отличающих системы разных классов, используются следующие:

- наличие/отсутствие в системе факторов неопределенности, влияющих на свойства системы;
- наличие/отсутствие в системе возможностей управления ее поведением как способа активного реагирования на изменения в текущей и/или прогнозируемой ситуации;
- наличие/отсутствие в системе возможностей адаптации к изменениям в свойствах объекта и/или среды;
- наличие/отсутствие в системе возможностей целеполагания.

1.2.1 Детерминированные системы

Низший уровень в иерархии систем \mathbb{S} — это детерминированные динамические системы \mathbb{DS} ⁵, т. е. такие, которые на одни и те же воздействия реагируют (откликаются) одним и тем же образом. Свойства таких систем либо фиксированы, либо меняются по строго заданному закону (примеры — неуправляемая ракета со сбрасываемым стартовым ускорителем; изменение массы неуправляемого ЛА в процессе выработки топлива).

Будем считать, что система \mathbb{DS} есть тройка

$$\mathbb{DS} = \langle X^{DS}, \Phi^{DS}, T^{DS} \rangle, \quad (1.3)$$

$$X^{DS} \subseteq X, T^{DS} \subseteq T, \Phi^{DS} = \Phi^{DS}(\mathbf{x}, t), \mathbf{x} \in X^{DS}, t \in T^{DS}.$$

Здесь X — *фазовое пространство* (пространство состояний) системы⁶ (1.3), элементы («точки») $\mathbf{x} \in X$ которого представляют собой возможные *фазовые состояния* (фазовый вектор) данной системы, $\mathbf{x} = (x_1, \dots, x_n)^T$, $x_1 \in X_1, \dots, x_n \in X_n$, $X = X_1 \times \dots \times X_n$; x_1, \dots, x_n — фазовые переменные (фазовые координаты) системы \mathbb{DS} ; $X^{DS} \subseteq X$ — область допустимых значений фазовых состояний системы (1.3); T — множество всех возможных моментов времени («мировое время»), наделенное структурой линейного порядка (т. е. упорядоченное отношением \leq), $T^{DS} \subseteq T$ — множество моментов времени функционирования системы

⁵DS — сокращение от Deterministic System, т. е. детерминированная система.

⁶Принятая здесь терминология восходит к традициям механики, теории динамических систем, а также теории управления и управляемых систем. Это представляется вполне логичным, учитывая ту основополагающую роль, которую играет понятие динамической системы для интересующего нас класса проблем.

(«системное время»); $\Phi^{DS} = \Phi^{DS}(\mathbf{x}, t)$, $\mathbf{x} \in X^{DS}$, $t \in T^{DS}$ — правило, которое позволяет определить состояние системы (1.3) в каждый момент времени $t \in T^{DS}$, зная ее состояния во все предыдущие моменты времени⁷. Системы класса \mathbb{DS} — предмет изучения современной теории динамических систем (см., например, [27,28]).

1.2.2 Системы с неопределенностями

Системы класса \mathbb{DS} , свойства которых являются детерминированными, представляют собой довольно редкий частный случай динамических систем. Гораздо чаще ДС содержат в себе неопределенности какого-либо вида из числа перечисленных в разд. 1.1.2. Эти неопределенности могут быть обусловлены, в частности, неполным и неточным знанием некоторых свойств объекта. Например, для ЛА весьма частой является ситуация, когда его аэродинамические характеристики известны не точно и не полностью (отсутствуют данные по ним для части режимов полета).

Система \mathbb{VS} ⁸, как и система \mathbb{DS} , остается неуправляемой и реагирующей на регулярные воздействия среды $\mathbb{VS} \rightleftharpoons \mathcal{E}$, но теперь содержит в себе неопределенности параметрического и/или функционального вида, связанные с устройством объекта (см. разд. 1.1.2). В соответствии с этим, систему \mathbb{VS} можно определить следующим образом:

$$\begin{aligned} \mathbb{VS} &= \langle X^{VS}, T^{VS}, \Phi^{VS} \rangle, \\ X^{VS} &\subseteq X, T^{VS} \subseteq T, \Phi^{VS} = \Phi^{VS}(\mathbf{x}, \boldsymbol{\xi}, t), \\ \mathbf{x} &\in X^{VS}, \boldsymbol{\xi} \in \Xi, t \in T^{VS} \subseteq T. \end{aligned} \tag{1.4}$$

Обозначения для системы (1.4) аналогичны введенным выше для системы (1.3). Отличие заключается в форме записи правила Φ^{VS} , в которое входят теперь факторы неопределенности $\boldsymbol{\xi} \in \Xi$. Как показано далее в разд. 1.4.2, системы класса \mathbb{VS} обычно не только содержат в себе факторы неопределенности, обусловленные особенностями устройства ДС и доступной информацией о них, но и взаимодействуют со средой \mathcal{E} , которая также содержит факторы

⁷Это значит, вообще говоря, что правило Φ^S должно обладать бесконечной памятью, чтобы хранить все пройденные системой \mathbb{S} состояния. Чаще всего, исходя из специфики решаемой задачи, можно утверждать, что такого рода требование является чрезмерным и достаточно использовать предысторию конечной длины, в ряде же случаев можно считать, что все будущие состояния системы \mathbb{S} для $t > t_i$; $t, t_i \in T^S$ определяются лишь ее состоянием $\mathbf{x}(t_i)$ в данный текущий момент времени $t_i \in T^S$ (и, разумеется, правилом Φ^S).

⁸ \mathbb{VS} — сокращение от Vague System, т. е. система, содержащая в себе неопределенности какого-либо вида.

неопределенности (пример такой среды — турбулентная атмосфера, в которой совершает полет ЛА). Соответственно, в таком случае состояние ДС будет теперь зависеть не только от текущего состояния $x(t_i)$ рассматриваемой системы и времени, но и от значения внешнего неконтролируемого возмущения, описываемого частью компонент вектора ξ , принимающего значения из некоторой области Ξ . «Неконтролируемость» внешнего возмущающего воздействия означает, что для системы $\forall S$ нет полной априорной информации о характеристиках данного воздействия со стороны среды \mathcal{E} , при этом соответствующие компоненты вектора ξ могут быть случайными или нечеткими величинами.

1.2.3 Управляемые системы

Как отмечалось выше, динамическая система S взаимодействует со средой \mathcal{E} , т.е. воспринимает воздействия окружающей среды и соответствующим образом реагирует на них. Реакция системы на воздействия среды может быть как пассивной, так и активной.

Пассивное взаимодействие — это, например, движение камня, полет артиллерийского снаряда или неуправляемой ракеты под воздействием гравитационных и аэродинамических сил. Именно такого рода взаимодействие реализуют системы DS вида (1.3) и $\forall S$ вида (1.4).

При активном взаимодействии система, получив воздействие среды, по тем или иным правилам формирует и реализует «ответ» (реакцию) системы на данное воздействие, например, отклоняет руль, парируя возмущение. Это значит, что динамическая система, способная активно взаимодействовать со средой, есть система *управляемая*.

Управляемую динамическую систему CS ⁹, которая активно откликается на воздействия среды и способна в определенных пределах парировать возмущения, возникающие при взаимодействии $CS \rightleftharpoons \mathcal{E}$, опишем следующим образом:

$$\begin{aligned} CS &= \langle X^{CS}, T^{CS}, \Phi^{CS}, \rangle, \\ X^{CS} &\subseteq X, T^{CS} \subseteq T, \Phi^{CS} = \Phi^{CS}(\mathbf{x}, \mathbf{u}, \xi, t), \\ \mathbf{x} &\in X^{CS}, \mathbf{u} \in U^{CS}, \xi \in \Xi, t \in T^{CS} \subseteq T. \end{aligned} \quad (1.5)$$

Как и в случае системы $\forall S$ вида (1.4), обозначения для системы (1.5) аналогичны введенным выше для системы (1.3). Отличие опять заключается в форме записи правила Φ^{CS} , которое для системы CS зависит не только от текущего состояния $x(t_i)$ рассматриваемой системы, внешнего неконтролируемого возмущения ξ и времени, но еще и от так называемых

⁹CS — сокращение от Controlled System, т.е. управляемая система.

управляющих величин \mathbf{u} , принимающих (мгновенные) значения из некоторой области U^{CS} .

Итак, система класса \mathbb{CS} — это динамическая управляемая целенаправленная система, регулярным образом реагирующая на воздействия среды \mathcal{E} . В более общем случае система класса \mathbb{CS} может находиться под воздействием различного рода неопределенных факторов $\xi \in \Xi$. Для системы данного класса характерно то, что, в отличие от систем следующих двух иерархических уровней (адаптивных и интеллектуальных), цель управления является внешней по отношению к системе и учитывается только на этапе синтеза закона управления для нее.

1.2.4 Адаптивные системы

Важнейшее отличие систем класса \mathbb{CS} от систем классов \mathbb{VS} и \mathbb{DS} состоит в том, что у систем \mathbb{CS} имеются возможности активно реагировать на воздействия среды \mathcal{E} . Эти возможности, однако, существенно ограничиваются тем обстоятельством, что правило Φ^{CS} (закон управления системы \mathbb{CS}) в (1.5) предполагается фиксированным, никак не изменяемым в процессе функционирования системы (1.5). Но если область неопределенности Ξ «достаточно велика», может получиться, что невозможно будет сформировать закон управления такой, чтобы он обеспечивал требуемый уровень качества управления при любых значениях $\xi \in \Xi$. Вследствие этого возникает потребность воздействовать на вид закона управления системы (правила Φ) непосредственно в процессе функционирования системы. Системы \mathbb{AS} ¹⁰, обладающие такого рода свойством, именуют *адаптивными* и в общем виде их можно представить следующим образом:

$$\begin{aligned} \mathbb{AS} &= \langle X^{AS}, T^{AS}, \Phi^{AS}, \Psi^{AS}, \Gamma^{AS} \rangle, \\ X^{AS} &\subseteq X, T^{AS} \subseteq T, \Phi^{AS} = \Phi^{AS}(\mathbf{x}, \mathbf{u}, \xi, t), \Psi^{AS} = \Psi^{AS}(\gamma, t), \\ \Gamma^{AS} &\subseteq \Gamma, \mathbf{x} \in X^{AS}, \mathbf{u} \in U^{AS}, \xi \in \Xi, \gamma \in \Gamma, t \in T^{AS} \subseteq T. \end{aligned} \quad (1.6)$$

В определении адаптивной системы \mathbb{AS} по сравнению с управляемой системой класса \mathbb{CS} появились два новых элемента. Первый из них — это правило Ψ^{AS} , которое описывает способы модификации правила Φ^{AS} , определяющего текущее поведение рассматриваемой системы. Второй новый элемент — множество целей Γ^{AS} , согласно которым протекает работа правила Ψ^{AS} .

¹⁰AS — сокращение от Adaptive System, т. е. адаптивная система.

Итак, система класса \mathbb{AS} — это динамическая управляемая целенаправленная система, регулярным образом взаимодействующая со средой и дополнительно к этому находящаяся под воздействием различного рода неопределенных факторов. Комплекс целей, организующих поведение, в системах класса \mathbb{AS} неизменен.

1.2.5 Интеллектуальные системы

Адаптивные системы, т. е. системы класса \mathbb{AS} , в сравнении с системами класса \mathbb{CS} получили возможность модифицировать свое поведение непосредственно в процессе функционирования системы за счет воздействия (через правило Ψ^{AS}) на правило Φ^{AS} , которое определяет текущее поведение системы \mathbb{AS} .

Однако характерная черта систем \mathbb{AS} состоит в том, что правило модификации характера поведения у них фиксировано при создании системы. Если учесть, что выбор этого правила осуществлялся, исходя из некоторых целей, на достижение которых ориентировалась создаваемая система, то это означает, что поведение системы \mathbb{AS} будет адекватным до тех пор, пока цели, поставленные перед ней (и жестко зафиксированные) остаются адекватными меняющейся ситуации. В среде с высокими уровнями разнородных неопределенностей изменение ситуации может быть таким, что повлечет за собой необходимость привлечения высшего уровня адаптации — адаптации целей (см. разд. 1.7.1). Но системы класса \mathbb{AS} этого делать не умеют — у них отсутствует механизм целеполагания, т. е. механизм формирования новых целей, если в этом возникнет необходимость, а также механизм воздействия на правило Ψ^{AS} , т. е. механизм, корректирующий правило модификации поведения.

Добавим \mathbb{AS} новое свойство — целеполагание¹¹. Системы \mathbb{IS} ¹², обладающие такого рода свойством, будем именовать *интеллектуальными* и в общем виде представлять их следующим образом:

$$\begin{aligned} \mathbb{IS} &= \langle X^{IS}, T, \Phi^{IS}, \Psi^{IS}, \Gamma^{IS}, \Omega^{IS}, \Sigma^{IS} \rangle, \\ X^{IS} \subseteq X, T^{IS} \subseteq T, \Phi^{IS} &= \Phi^{IS}(\mathbf{x}, \mathbf{u}, \boldsymbol{\xi}, t), \Psi^{AS} = \Psi^{AS}(\boldsymbol{\gamma}, t), \\ \Gamma^{AS} \subseteq \Gamma, \Omega^{IS} &= \Omega^{IS}(\mathbf{x}, \mathbf{u}, \boldsymbol{\xi}, \boldsymbol{\gamma}, t), \\ \mathbf{x} \in X^{IS}, \mathbf{u} \in U^{IS}, \boldsymbol{\xi} \in \Xi, \boldsymbol{\gamma} \in \Gamma, \Sigma^{IS} &\subseteq \Sigma, t \in T^{IS} \subseteq T. \end{aligned} \tag{1.7}$$

¹¹ Моделирование целеполагания — одна из важнейших тем современной робототехники, не получившая пока своего удовлетворительного решения. Результаты, имеющиеся в данной области, освещены в [49]

¹² IS — сокращение от Intelligent System, т. е. интеллектуальная система.

По сравнению с адаптивной системой \mathbb{AS} в определении интеллектуальной системы \mathbb{IS} добавилось правило Ω^{IS} , описывающее способ порождения целей $\gamma \in \Gamma$, т. е. способ изменения множества целей Γ , а также еще одна структура, Σ^{IS} , задающая такие элементы, как ценности, мотивы и т. п., руководствуясь которыми осуществляется процесс целеполагания.

Итак, система класса \mathbb{IS} — это динамическая управляемая целенаправленная система, располагающая средствами (механизмами) целеполагания. Система данного класса, как и системы классов \mathbb{CS} , \mathbb{AS} взаимодействуют со средой как регулярным образом, так и под воздействием различного рода неопределенных факторов.

1.3 Классы внешних сред

Аналогично классам систем \mathbb{S} , можно ввести и классы среды \mathcal{E} . Будем последовательно вводить усложняющиеся классы внешних сред, при этом будем считать, что они взаимодействуют с системой общего вида \mathbb{S} .

1.3.1 Регулярные среды

Регулярная среда \mathcal{SE}^{13} , представляет собой низший уровень в иерархии сред. Она реализует регулярное (т. е. не содержащее никаких неопределенностей) воздействие на систему \mathbb{S} . Примером среды класса \mathcal{SE} может служить центральное гравитационное поле (поле тяготения).

1.3.2 Среды с неопределенностями

Среда \mathcal{UE}^{14} (*среда с неопределенностями*) — это следующий уровень иерархии сред. Она характерна тем, что воздействие, которое она оказывает на систему \mathbb{S} , содержит ряд неопределенных факторов, т. е. факторов, априори неизвестных, в отличие от предыдущего случая, неконтролируемых и не поддающихся измерению средствами системы \mathbb{S} .

Примером среды класса \mathcal{UE} может служить турбулентная атмосфера.

¹³SE — сокращение от Stereotyped Environment, т. е. регулярная среда, воздействующая на систему \mathbb{S} некоторым шаблонным, стандартным, трафаретным образом, не меняющимся в течение срока жизни T^S системы \mathbb{S} .

¹⁴UE — сокращение от Uncertain Environment, т. е. среда, содержащая в себе некоторые неопределенности, т. е. факторы, которые система \mathbb{S} не может ни контролировать, ни, в ряде случаев, даже и измерять.

1.3.3 Реагирующие среды

Среды классов \mathcal{SE} и \mathcal{UE} — *пассивные*, они воздействуют некоторым образом на систему \mathcal{S} , но сами в ответ на действия системы \mathcal{S} реагируют очень ограниченно и нецеленаправленно. Например, турбулентная атмосфера влияет на траекторию движения тела и испытывает некоторое ответное воздействие, результаты которого проявляются в виде спутного следа, возникающего за телом.

Среды всех следующих классов (иерархических уровней) — *активные*, реагирующие тем или иным образом на действия системы \mathcal{S} . Существенный момент, который следует подчеркнуть — активное взаимодействие среды \mathcal{E} с системой \mathcal{S} может быть направлено как на содействие целям системы (пример — система управления воздушным движением как среда, в которой осуществляет полет транспортный самолет), так и на противодействие этим целям (пример — система противовоздушной обороны, в среде которой выполняет задачу наряд средств, предназначенный для поражения цели, прикрываемой данной системой ПВО). Еще один возможный вариант взаимодействия среды с системой — среда «не мешает» (т. е. и не способствует, и не противодействует) деятельности системы в ней. Первый из классов активных сред — это *реагирующие* среды \mathcal{RE} ¹⁵. Среды класса \mathcal{RE} откликаются изменениями в них на действия системы \mathcal{S} . Однако эти изменения нецеленаправленны.

Примером подобного рода сред могут служить фрагменты природной среды (биосферы), в пределах которых осуществляет свою деятельность та или иная техническая система.

1.3.4 Адаптивные среды

Следующий класс активных сред — это *адаптивные среды* \mathcal{AE} ¹⁶. Для сред данного класса характерен активный и *целенаправленный* отклик на действия системы \mathcal{S} . При этом, однако, цели (комплекс целей), которые направляют характер процессов в среде \mathcal{AE} , являются фиксированными, выбранными и заданными до начала процессов взаимодействия системы \mathcal{S} и среды \mathcal{AE} . Новые цели формироваться в средах класса \mathcal{AE} не могут.

Примером адаптивной среды может служить максимально автоматизированная система противовоздушной обороны (ПВО), особенно в ее территориальном варианте. Данная система

¹⁵RE — сокращение от Reacting Environment, т. е. среда, реагирующая (откликающаяся) некоторым образом на деятельность системы \mathcal{S} .

¹⁶AE — сокращение от Adaptive Environment, т. е. среда, активно и *целенаправленно* реагирующая определенным образом на деятельность системы \mathcal{S} .

ПВО должна рассматриваться как внешняя среда, если решается задача преодоления ПВО с помощью некоторого набора средств — самолетов, вертолетов, крылатых ракет и т. д.

1.3.5 Интеллектуальные среды

Третий, высший уровень активных внешних сред (или пятый по общему счету) — это *интеллектуальные внешние среды* IE ¹⁷. Среды данного класса обладают механизмами целеполагания и собственными интересами («свободой воли», в известном смысле).

В качестве примера здесь можно указать, как и в случае адаптивных сред, территориальную систему ПВО, но более высокого иерархического уровня — воздушную армию ПВО, включающую истребительную авиацию, зенитно-ракетные комплексы, средства обнаружения и сопровождения целей и многое другое, в том числе, что очень существенно, «человеческий фактор», входящий в различные звенья управления данной системой и обеспечивающий вариативность реакций системы-среды. Такого рода система ПВО будет внешней средой для авиационных средств нападения, решающих задачу поражения целей, прикрываемых данной системой.

1.4 Взаимодействие систем и среды

1.4.1 Иерархии систем и внешних сред

Выше были введены следующие классы динамических систем:

- детерминированная система (DS);
- система с неопределенностями (VS);
- управляемая система (CS);
- адаптивная система (AS);
- интеллектуальная система (IS).

По уровню потенциальных возможностей (в порядке нарастания этих возможностей) эти классы систем упорядочены следующим образом:

$$DS \subset VS \subset CS \subset AS \subset IS .$$

¹⁷IE — сокращение от Intelligent Environment, т. е. среда, не только активно и целенаправленно реагирующая определенным образом на деятельность системы S , но и располагающая аппаратом *порождения целей*.

Аналогично этому, иерархия классов среды может быть построена так:

$$\mathcal{SE} \subset \mathcal{UE} \subset \mathcal{RE} \subset \mathcal{AE} \subset \mathcal{IE} .$$

Здесь \mathcal{SE} — регулярная среда, \mathcal{UE} — среда с неопределенностями, \mathcal{RE} — реагирующая среда, \mathcal{AE} — адаптивная среда, \mathcal{IE} — интеллектуальная среда.

1.4.2 Системы-комплексы

1. Как отмечалось выше, систему \mathbb{S} следует рассматривать во взаимодействии со средой \mathcal{E} . Символически это положение будем обозначать следующим образом:

$$\begin{aligned} \text{Система-Комплекс } \mathbb{K} &= \text{Система-Объект } \mathbb{S} + \text{Система-Среда } \mathcal{E}, \\ \mathbb{K} &= (\mathbb{S} \rightleftharpoons \mathcal{E}), \end{aligned}$$

т. е. на наиболее общем уровне предметом рассмотрения является система-комплекс \mathbb{K} , которая состоит из двух взаимодействующих между собой систем — системы-объекта \mathbb{S} и системы-среды \mathcal{E} , в которой система-объект функционирует¹⁸:

$$\mathbb{K} = \langle \mathbb{S}, \mathcal{E}, T, \Theta \rangle, \quad \mathbb{S} \rightleftharpoons \mathcal{E}. \quad (1.8)$$

Здесь Θ — закон взаимодействия \mathbb{S} и \mathcal{E} во времени T .

2. Конкретный вид комплекса (1.8) определяется тем, каким образом определены его составные части $\mathbb{S}, \mathcal{E}, T, \Theta$.

Возможны, в частности, следующие варианты:

- комплекс $\mathbb{K}_{\mathbb{DS}}^{\mathcal{SE}} = \langle \mathbb{DS}, \mathcal{SE} \rangle$, представляющий собой неуправляемую детерминированную динамическую систему \mathbb{DS} , которая регулярным образом взаимодействует с детерминированной средой \mathcal{SE} (пример — объект, движущийся в гравитационном поле небесного тела, не обладающего атмосферой);
- комплекс $\mathbb{K}_{\mathbb{DS}}^{\mathcal{UE}} = \langle \mathbb{DS}, \mathcal{UE} \rangle$, представляющий собой неуправляемую детерминированную динамическую систему \mathbb{DS} , взаимодействующую со средой \mathcal{UE} , которая содержит факторы неопределенности (пример — неуправляемая ракета, движущаяся в беспокойной атмосфере);

¹⁸Далее для краткости Система-Комплекс будет именоваться просто «комплекс», Система-Объект — просто «система» («динамическая система») или «объект» (иногда с уточнением «объект моделирования» или «объект управления»), Система-Среда — просто «среда».

- комплекс $\mathbb{K}_{\mathbb{V}\mathbb{S}}^{\mathcal{U}\mathcal{E}} = \langle \mathbb{V}\mathbb{S}, \mathcal{U}\mathcal{E} \rangle$, представляющий собой неуправляемую динамическую систему $\mathbb{V}\mathbb{S}$ с неопределенностями, взаимодействующую со средой $\mathcal{U}\mathcal{E}$, которая содержит факторы неопределенности (пример — неуправляемая ракета с неточно известными АДХ, движущаяся в неспокойной атмосфере);
- комплекс $\mathbb{K}_{\mathbb{C}\mathbb{S}}^{\mathcal{S}\mathcal{E}} = \langle \mathbb{C}\mathbb{S}, \mathcal{S}\mathcal{E} \rangle$, представляющий собой управляемую детерминированную динамическую систему $\mathbb{C}\mathbb{S}$, которая регулярным образом взаимодействует с детерминированной средой $\mathcal{S}\mathcal{E}$ (пример — летательный аппарат, осуществляющий управляемое движение в спокойной атмосфере);
- комплекс $\mathbb{K}_{\mathbb{A}\mathbb{S}}^{\mathcal{U}\mathcal{E}} = \langle \mathbb{A}\mathbb{S}, \mathcal{U}\mathcal{E} \rangle$, представляющий собой адаптивную динамическую систему $\mathbb{A}\mathbb{S}$, взаимодействующую со средой $\mathcal{U}\mathcal{E}$, которая содержит факторы неопределенности (пример — летательный аппарат, действующий в среде с неопределенностями¹⁹ и обладающий способностью оперативно приспосабливаться к ним).

1.5 Поведение и деятельность систем

1. Текущее состояние системы \mathbb{S} описывается набором (1.14) величин $x_i \in X_i$, характеризующих ее в решаемой задаче. Этот набор принято трактовать как кортеж длины n

$$x = \langle x_1, x_2, \dots, x_n \rangle, \quad x_i \in X_i, \quad i = 1, 2, \dots, n$$

или как вектор-столбец $x = [x_1 x_2 \dots x_n]^T$. При этом

$$x \in X, \quad X \subseteq X_1 \times X_2 \times \dots \times X_n.$$

Областями значений X_i непрерывных величин x_i являются обычно подмножества множества вещественных чисел \mathcal{R} .

2. Какие именно величины x_i войдут в набор (вектор) x , описывающий состояние системы \mathbb{S} , зависит от природы данной системы и от решаемой задачи. В разных задачах одну и ту же систему \mathbb{S} будут характеризовать, как правило, различные (по составу величин x_i) наборы x . Иллюстрацией этого может служить Пример 1.1 на с. 39, а также Пример 1.2 на с. 40. Видно, что один и тот же объект моделирования в разных задачах, решаемых для него (продольное движение самолета при наборе высоты и криволинейный полет самолета в горизонтальной

¹⁹Неопределенности, встречающиеся в проблемах моделирования поведения ДС, многообразны по своей природе (см., например, [35–37]).

плоскости, соответственно) требуют несовпадающих наборов величин, характеризующих состояние объекта.

3. Точка $x \in X \subseteq \mathcal{R}^n$ в пространстве состояний представляет собой состояние системы \mathbb{S} в некоторый момент времени $t \in T = [t_0, t_N]$. Для случая непрерывного времени $t \in T$ и конечномерного вектора состояния $x \in X \subseteq \mathcal{R}^n$ задать состояния во все моменты времени — значит задать векторную функцию

$$x(t) = (x_1(t), x_2(t), \dots, x_n(t)) = [x_1(t)x_2(t) \dots x_n(t)]^T. \quad (1.9)$$

Современные и перспективные системы управления создаются и реализуются на борту ЛА в цифровой среде, что предопределяет необходимость перехода от непрерывного времени $t \in [t_0, t_N]$ к дискретному времени $t \in \{t_i\}$, $i = 1, \dots, N$. Соответственно, вместо непрерывных фазовых траекторий (1.9) в данном случае рассматриваются их дискретные представления в виде наборов последовательностей вида:

$$x(t_i) = \{(x_1(t_i), x_2(t_i), \dots, x_n(t_i))\}, \quad i = 0, 1, \dots, N. \quad (1.10)$$

4. Поведение системы \mathbb{S} — это последовательность ее фазовых состояний $x(t_i) \in X$, связанных к соответствующим моментам времени $t_i \in T$, т. е.

$$\{\langle x(t_i), t_i \rangle\}, \quad t_i \in [t_0, t_f] \subseteq T, \quad i = 0, 1, \dots, N. \quad (1.11)$$

Деятельность системы \mathbb{S} — это последовательность ее целенаправленных действий, каждое из которых представляет собой реакцию вида

$$\langle \text{ситуация, цель} \rangle \Rightarrow \text{действие} \Rightarrow \text{результат},$$

т. е.

$$\{\langle \lambda(t_i), \gamma(t_i) \rangle\} \xrightarrow{\Phi^S} \lambda(t_{i+1}), \quad i = 0, 1, 2, \dots, N, \quad (1.12)$$

или, что то же самое,

$$\lambda(t_{i+1}) = \Phi^S(\langle \lambda(t_i), \gamma(t_i) \rangle), \quad i = 0, 1, 2, \dots, N.$$

Здесь $\lambda(t_i) \in \Lambda$ — текущая ситуация, $\gamma(t_i) \in \Gamma$ — текущая цель, Φ^S — закон эволюции системы \mathbb{S} .

Поведение (1.11) есть у систем \mathbb{S} всех видов, включая неуправляемые и управляемые, с неопределенностями и без них. В отличие от этого, деятельность (1.12) имеется только у

систем, в состав которых в том или ином виде входит формулировка целей управления: неизменяемые цели (адаптивные системы) или самостоятельно корректируемые системой цели (интеллектуальные системы).

Управление поведением системы \mathbb{S} представляет собой более общее понятие, чем управление ее движением. Какие именно элементы входят в управление поведением, можно видеть из перечня, представленного во введении на с. 14.

Понятие деятельности робототехникой заимствовано из наук о живых системах (биология, психология, этология). Все расширяющиеся работы по роботизации летательных аппаратов, особенно беспилотных ЛА, приводят к необходимости расширения класса задач управления ЛА от традиционных задач управления движением до задач управления поведением и деятельностью ЛА. Особенно это актуально для высокоавтономных роботизированных БПЛА.

Действие, исполняемое системой \mathbb{S} , не обязательно зависит только от текущих (т. е. рассматриваемых в данный момент времени $t_i \in T$) значений ситуации $\lambda(t_i)$ и цели $\gamma(t_i)$. В более общем случае оно зависит от набора ситуаций $\Lambda(t_i)$ и набора целей $\Gamma(t_i)$ в данный момент времени t_i , т. е.

$$\begin{aligned} \langle \Lambda(t_i), \Gamma(t_i) \rangle &\xrightarrow{\Phi^S} \lambda(t_{i+1}), \quad i = 0, 1, 2, \dots, N; \\ \Lambda(t_i) &\subset \Lambda^S, \quad \Gamma(t_i) \subset \Gamma^S. \end{aligned} \tag{1.13}$$

Здесь переход к ситуации $\lambda(t_{i+1})$ осуществляется не только с учетом текущей ситуации $\lambda(t_i)$ и текущей цели $\gamma(t_i)$ в данный момент времени t_i , но и с учетом прошлых (предыстория) и будущих (прогноз) состояний и целей, что отражается соответствующими множествами: набором ситуаций $\Lambda(t_i)$ и набором целей $\Gamma(t_i)$ для данного момента времени t_i .

1.6 Общий подход к решению проблемы моделирования динамических систем

1.6.1 Динамическая система как объект исследования

1. Введем формализованное понятие системы \mathbb{S} в том виде, в котором оно будет использоваться далее. В общем случае в таком описании необходимо показать следующие элементы, связанные с \mathbb{S} :

- 1) совокупность величин (с областями их допустимых значений), характеризующих \mathbb{S} и условия, в которых она функционирует;
- 2) совокупность величин (с областями их допустимых значений), характеризующих факторы, влияющие на состояния системы \mathbb{S} :

3) время, в котором \mathbb{S} функционирует;

4) закон функционирования \mathbb{S} , т. е. набор правил²⁰, согласно которым совокупность величин, характеризующих \mathbb{S} , меняется во времени.

Совокупность величин, характеризующих систему \mathbb{S} , включает в себя:

- величины x_1, \dots, x_n , характеризующие состояние \mathbb{S} ,
- величины y_1, \dots, y_m , характеризующие результаты наблюдения состояния \mathbb{S} .

Величины x_1, \dots, x_n , характеризующие состояние системы \mathbb{S} , объединяются в совокупность (вектор) x , именуемый состоянием (вектором состояния) данной системы:

$$\begin{aligned} x &= (x_1, \dots, x_i, \dots, x_n); \\ x_i &\in X_i \subset \mathcal{R}; \quad x \in R_X \subset X = X_1 \times \dots \times X_n. \end{aligned} \tag{1.14}$$

Здесь X — область всех возможных состояний системы \mathbb{S} (пространство состояний²¹); R_X — область всех допустимых состояний системы \mathbb{S} ; \mathcal{R} — множество вещественных чисел.

Величины y_1, \dots, y_m , характеризующие результаты наблюдения состояния системы \mathbb{S} , объединяются в вектор наблюдения:

$$\begin{aligned} y &= (y_1, \dots, y_j, \dots, y_p); \\ y_j &\in Y_j \subset \mathcal{R}; \quad y \in R_Y \subset Y = Y_1 \times \dots \times Y_p. \end{aligned} \tag{1.15}$$

Перечень величин, которые характеризуют факторы, влияющие на состояние системы \mathbb{S} , включает в себя следующие элементы:

- величины, характеризующие воздействия на \mathbb{S} , как контролируемые, так и неконтролируемые, в том числе:
 - управления u_1, \dots, u_m как контролируемые воздействия на \mathbb{S} , т. е. такие, которые можно варьировать для удовлетворения целей управления данной системой;
 - возмущения ξ_1, \dots, ξ_q как неконтролируемые воздействия на \mathbb{S} (они могут обладать как известными, так и неизвестными свойствами, могут быть как измеримыми, например, температура уличного воздуха в задаче терморегулирования в доме, так и неизмеримыми, например, атмосферная турбулентность);

²⁰В теории динамических систем [27, 29] этот набор правил часто именуют также законом эволюции системы \mathbb{S} .

²¹Пространство состояний в теории динамических систем часто именуют также фазовым пространством, а состояния системы \mathbb{S} — фазовыми состояниями.

- шумы ζ_1, \dots, ζ_r , характеризующие погрешности, вносимые измерительной аппаратурой;
- величины w_1, \dots, w_s (постоянные или переменные), характеризующие свойства системы \mathbb{S} (их значения прямо или косвенно определяются решениями, принимаемыми на этапе создания \mathbb{S}) и влияющие на поведение системы через закон ее эволюции (они входят как постоянные или переменные параметры в этот закон); примеры постоянных параметров системы — размах крыла, площадь крыла, длина самолета и т. п.; примеры переменных параметров системы — коэффициенты аэродинамических сил и моментов, представляющие собой нелинейные функции нескольких переменных (переменных состояния объекта и среды, а также управляющих переменных).

Величины u_1, \dots, u_m , характеризующие управляющие воздействия на состояние системы \mathbb{S} , объединяются в совокупность u , именуемую управлением (вектором управления) данной системы:

$$u = (u_1, \dots, u_k, \dots, u_m); \quad (1.16)$$

$$u_k \in U_k \subset \mathcal{R}; \quad u \in R_U \subset U = U_1 \times \dots \times U_m.$$

Здесь U — область всех возможных состояний системы \mathbb{S} ; R_U — область всех допустимых состояний системы \mathbb{S} .

ПРИМЕР 1.1. Продольное движение самолета при наборе высоты.

Система уравнений движения самолета при наборе высоты для неустановившегося ($\dot{V}_H \neq 0$) и непрямолинейного ($\dot{\theta}_H \neq 0$) полета может быть записана в виде [39, 41]:

$$m \frac{dV_H}{dt} = P - X_a - G \sin \theta_H, \quad (1.17)$$

$$m V_H \frac{d\theta_H}{dt} = Y_a - G \cos \theta_H.$$

Здесь X_a — аэродинамическая сила лобового сопротивления; Y_a — подъемная сила; P — тяга силовой установки; θ_H — угол наклона траектории; V_H — скорость полета; $G = mg$ — вес самолета.

В задаче набора высоты состояние самолета как динамической системы описывается двумя величинами: скоростью полета V_H и углом наклона траектории θ_H . Соответственно, вектор состояния $x \in X$ в данном случае имеет вид:

$$x = (x_1, x_2) = (V_H, \theta_H).$$

ПРИМЕР 1.2. Криволинейный полет самолета в горизонтальной плоскости.

Система уравнений движения самолета при выполнении им виража с креном и скольжением может быть записана в виде [39, 41]:

$$\begin{aligned} m \frac{dV}{dt} &= P - X_a, \\ mV \frac{d\Psi}{dt} &= -Z_a \cos \gamma_a - Y_a \sin \gamma_a + P \sin \beta \cos \gamma_a, \\ 0 &= Y_a \cos \gamma_a - G. \end{aligned} \quad (1.18)$$

Здесь X_a — аэродинамическая сила лобового сопротивления; Y_a — подъемная сила; Z_a — боковая сила; P — тяга силовой установки; Ψ — угол рыскания; V — скорость полета; β — угол скольжения; γ_a — угол крена; $G = mg$ — вес самолета.

В задаче выполнения виража с креном и скольжением состояние самолета как динамической системы описывается такими величинами, как скорость полета V и угол рыскания Ψ . Соответственно, вектор состояния $x \in X$ в данном случае имеет вид:

$$x = (x_1, x_2) = (V, \Psi).$$

ПРИМЕР 1.3. Продольное угловое движение самолета.

Система уравнений, описывающая его продольное короткопериодическое движение самолета, может быть записана в виде [39]:

$$\begin{aligned} \dot{\alpha} &= \omega_z - \frac{qS}{mV} C_{y_a}(\alpha, \omega_z, \varphi) + \frac{g}{V} \cos \vartheta, \\ \dot{\omega}_z &= \frac{qS b_A}{J_{zz}} m_z(\alpha, \omega_z, \varphi), \\ T^2 \ddot{\varphi} &= -2T \kappa \dot{\varphi} - \varphi + \varphi_{act}, \end{aligned} \quad (1.19)$$

где α — угол атаки, град; ϑ — угол тангажа, град; ω_z — угловая скорость тангажа, град/с; φ — угол отклонения управляемого стабилизатора, град; C_{y_a} — коэффициент подъемной силы; m_z — коэффициент момента тангажа; m — масса самолета, кг; V — воздушная скорость, м/с; $q = \rho V^2 / 2$ — скоростной напор; ρ — плотность воздуха, кг/м³; g — ускорение силы тяжести, м/с²; S — площадь крыла, м²; b_A — средняя аэродинамическая хорда крыла, м; J_{zz} — момент инерции самолета относительно боковой оси, кг·м²; безразмерные коэффициенты C_{y_a} и m_z являются нелинейными функциями своих аргументов; T, ζ — постоянная времени и коэффициент относительного демпфирования привода, φ_{act} — командный сигнал на привод цельноповоротного управляемого стабилизатора (ограничивается $\pm 25^\circ$). В модели (1.19) величины $\alpha, \omega_z, \varphi$ и $\dot{\varphi}$ — это состояния объекта управления, величина φ_{act} — управление. Здесь же $g(H)$

и $\rho(H)$ — величины, характеризующие состояние среды (гравитационного поля и атмосферы, соответственно), где H — высота полета; $m, S, b_A, J_{zz}, T, \varkappa$ — постоянные параметры объекта моделирования, C_{y_a} и m_z — переменные параметры объекта моделирования.

2. Помимо ограничений R_X на допустимые комбинации значений состояний системы \mathbb{S} , а также ограничений R_U на допустимые комбинации значений управляющих воздействий на ее состояние, имеются также, как правило, ограничения на допустимые значения комбинаций компонент вектора состояния x и управляющего вектора u :

$$\langle x, u \rangle \in R_{XU} \subset X \times U = X_1 \times \dots \times X_n \times U_1 \times \dots \times U_p. \quad (1.20)$$

3. Учитывая сказанное выше, в общем виде систему \mathbb{S} можно представить в следующем виде:

$$\mathbb{S} = \langle \{U, \Xi, Z\}, \{F, G\}, \{X, Y\}, T \rangle, \quad (1.21)$$

где U — контролируемые (управляющие) воздействия на \mathbb{S} ; Ξ и Z — неконтролируемые воздействия соответственно на состояния и выходы системы \mathbb{S} ; F и G — правила изменения соответственно состояния и выхода системы \mathbb{S} во времени; X и Y — соответственно множество состояний и выходов системы \mathbb{S} . Элементы этих множеств будем обозначать как $u \in U$, $\xi \in \Xi$, $\zeta \in Z$, $x \in X$, $y \in Y$, $t \in T$.

4. Через T в (1.21) обозначается интервал времени, на котором рассматривается система \mathbb{S} . Время внутри этого интервала может быть как непрерывным, т. е. $T \subset \mathcal{R}$, так и дискретным. В случае дискретного времени последовательность моментов времени задается следующим правилом:

$$T = \{t_0, t_1, \dots, t_{i-1}, t_i, t_{i+1}, \dots, t_{N-1}, t_N\}, \quad t_N = t_f, \quad (1.22)$$

$$t_{i+1} = t_i + \Delta t, \quad i = 0, 1, \dots, N.$$

В дальнейшем, если не оговорено обратное, будет использоваться дискретное время (1.22). Это представляется вполне естественным применительно к рассматриваемому кругу задач. А именно, при создании систем \mathbb{S} рассматриваемого класса активно используется аппарат обыкновенных дифференциальных и дифференциально-алгебраических уравнений. Решение задач анализа и идентификации таких систем, а также синтеза управления для них в настоящее время осуществляется, как правило, с привлечением численного интегрирования ОДУ и ДАУ, реализуемого в цифровой среде и основанного на переходе от ОДУ или ДАУ к их конечно-разностной аппроксимации, задаваемой соответствующими рекуррентными соотношениями

(рекуррентные схемы Рунге-Кутты, Адамса, прогноза и коррекции и т. п.). Таким образом, в процессе решения этих задач происходит обязательный переход от непрерывного времени к дискретному.

Аналогично, бортовая реализация систем управления современных и, тем более, перспективных ЛА осуществляется в настоящее время также в цифровой среде, т. е. на борту эти системы будут работать в дискретном времени.

Запись $u(t)$ здесь и далее обозначает векторную переменную величину u как функцию времени $t \in T$. Запись $u(t_i)$ обозначает мгновенное значение этой величины в момент дискретного времени $t_i \in [t_0, t_f]$. Будет также использоваться сокращенная запись u_i вместо $u(t_i)$. Запись $u^{(j)}(t)$ обозначает j -ю компоненту векторной переменной величины $u(t)$. Аналогично, запись $u^{(j)}(t_i)$ или, в сокращенном варианте, $u_i^{(j)}$, обозначает мгновенное значение j -й компоненты векторной переменной величины $u(t)$ в момент дискретного времени $t_i \in [t_0, t_f]$. Аналогичным образом вводятся соответствующие обозначения для величин $x \in X$, $y \in Y$, $\xi \in \Xi$, $\zeta \in Z$.

5. Как уже отмечалось выше, объектом исследования является управляемая динамическая система, действующая в условиях разнообразных неопределенностей. Эти неопределенности можно разделить на следующие основные виды:

- неопределенности, порожденные неконтролируемыми возмущениями, действующими на объект (например, атмосферная турбулентность, ветровые порывы);
- недостаточный уровень знаний об объекте моделирования и среде, в которой он функционирует (например, недостаточно точно известные или вообще неизвестные аэродинамические характеристики летательного аппарата);
- неопределенности, порожденные изменением свойств объекта из-за отказов его оборудования и повреждений в конструкции (например, боевые и/или эксплуатационные повреждения конструкции и отказы оборудования летательного аппарата, меняющие его свойства).

6. Для того, чтобы характеризовать текущее (мгновенное) состояние комплекса \mathbb{K} , введем понятие ситуации, которое включает в себя компоненты, характеризующие состояние как системы \mathbb{S} , так и среды \mathcal{E} . Компоненты, характеризующие систему \mathbb{S} , будем именовать внутренними, а среду \mathcal{E} — внешними, тогда

Ситуация = Внешняя-Ситуация + Внутренняя-Ситуация

$$\lambda(t_i) = \langle \lambda_{int}(t_i), \lambda_{ext}(t_i) \rangle, \quad \lambda(t_i) \in \Lambda, \quad \lambda_{int}(t_i) \in \Lambda_{int}, \quad \lambda_{ext}(t_i) \in \Lambda_{ext}$$

Наряду с понятием ситуации важную роль играет понятие ситуационной осведомленности. Если понятие ситуации описывает объективную реальность (объект + среда), то понятие ситуационной осведомленности характеризует степень информированности системы \mathbb{S} об этой реальности, т. е. о текущей ситуации. Она показывает состав данных, получаемых наблюдением и доступных в системе \mathbb{S} для выработки управляющих решений. Ситуационная осведомленность относительно части компонент ситуации обычно неполная (их значения известны неточно) или нулевая (их значения неизвестны). Она обеспечивается для одной части компонент прямым наблюдением (измерением), а для другой части — алгоритмическим путем, т. е. вычислением их значений по известным значениям других компонент.

Таким образом, когда говорится о системе \mathbb{S} с неопределенностями, речь идет о неполной ситуационной осведомленности для нее, т. е. о том, что значения части внутренних и внешних компонент ситуации для \mathbb{S} неизвестны, либо известны неточно.

1.6.2 Схема процесса моделирования динамической системы

1. По отношению к системе \mathbb{S} требуется найти ответ на следующие три основных вопроса:

- 1) Как \mathbb{S} будет реагировать («откликаться») на различного рода воздействия на нее?
- 2) Как добиться от \mathbb{S} таких реакций (откликов), которые требуется от нее получить?
- 3) Если имеются данные о реакциях системы \mathbb{S} в ответ на определенные воздействия, то каким может быть²² устройство системы \mathbb{S} , согласующееся с этими данными?

Чтобы получить ответы количественного характера на поставленные вопросы, необходимо принять ряд соглашений относительно системы \mathbb{S} и условий, в которых она работает. При этом требуется уточнить, что представляет собой объект исследования, а также какие задачи будут решаться для данного объекта.

2. Что касается ответа на первый из этих двух вопросов, то объектом исследования будут ДС с сосредоточенными параметрами, т. е. системы, рассматриваемые как твердое тело или взаимосвязанная совокупность твердых тел (набор твердых тел, между которыми имеются

²²«Может быть», а не «должно быть» — потому что существует, как правило, не единственное решение данной задачи.

кинематические и/или динамические связи). Такой подход охватывает чрезвычайно широкий круг прикладных задач из различных научно-технических областей. Традиционными математическими моделями таких ДС являются системы обыкновенных дифференциальных или дифференциально-алгебраических уравнений.

Не всегда рассмотрение ДС как твердого тела позволяет получить адекватную модель системы, в частности это не удастся, когда существенную роль играет упругость конструкции рассматриваемого объекта моделирования. В этом случае ДС следует рассматривать как систему с распределенными параметрами, для которой традиционным аппаратом моделирования являются дифференциальные уравнения в частных производных. Развиваемый далее подход к моделированию ДС может быть распространен и на такой случай. Этот класс систем, однако, представляет собой предмет отдельного исследования и в данной работе не рассматривается.

3. Для ответа на вопрос о том, какие классы проблем должны рассматриваться применительно к ДС как объекту моделирования и/или управления, введем определение системы \mathcal{S} как упорядоченной тройки следующего вида:

$$\mathcal{S} = \langle U, P, Y \rangle, \quad (1.23)$$

где U — входные воздействия на объект моделирования/управления; P — объект моделирования/управления; Y — реакции объекта на входные воздействия.

Входные воздействия U здесь — это начальные условия, управления, а также неконтролируемые внешние воздействия на объект P . В качестве объекта моделирования P могут рассматриваться, в частности, летательные аппараты различных видов. Выходы Y динамической системы \mathcal{S} представляют собой наблюдаемые реакции объекта P на входные воздействия U .

Имея в виду эти определения, можно следующим образом сформулировать три основных класса задач, связанных с ДС:

1. $\langle U, P, Y \rangle$ — анализ поведения ДС (по U и P найти Y).
2. $\langle U, P, Y \rangle$ — синтез управления ДС (по P и Y найти U).
3. $\langle U, P, Y \rangle$ — идентификация ДС (по U и Y найти P).

Здесь задача 1 относится к классу прямых, а задачи 2 и 3 — к классу обратных задач динамики систем. При этом задачу 3 можно охарактеризовать как задачу создания модели

ДС, а задачи 1 и 2 — как задачи, в которых осуществляется использование ранее созданных моделей. Предметом исследования в данной работе являются все эти классы задач, но с акцентом на задачу 3 вследствие как ее собственной значимости, так и критической важности для решения задач 1 и 2.

4. Еще один важный вопрос, на который необходимо ответить применительно к рассматриваемой ДС, — это вопрос о том, что делать с неопределенностями в ДС и среде ее функционирования.

Что касается характера учитываемых неопределенностей, то, как уже отмечалось в разд. 1.1.2, основные их виды будут следующими:

- неопределенности, порожденные неконтролируемыми возмущениями, действующими на объект;
- недостаточный уровень знаний об объекте моделирования и среде, в которой он функционирует;
- неопределенности, порожденные изменением свойств объекта из-за отказов его оборудования и повреждений в конструкции.

Анализ проблем, связанных с созданием сложных систем, действующих в условиях неопределенности, приводит к выводу о необходимости привлечения идей адаптивности. В реализации адаптивных систем, как было показано, ключевую роль играют модели объектов. Эти модели предназначаются, в первую очередь, для использования в составе бортовых комплексов создаваемых систем (например, летательных аппаратов) в реальном или даже опережающем времени, что определяет ряд требований к ним, включая такие, как повышенная точность, высокое быстродействие, адаптивность.

Традиционные модели (в виде ОДУ или ДАУ) не в полной мере удовлетворяют требованиям по точности и быстродействию. Требованию адаптивности они не удовлетворяют вообще. Пути реализации перечисленных требований могут быть следующими. Точность и быстродействие модели можно обеспечить за счет максимально полного использования знаний и данных об объекте моделирования. Адаптивность модели можно трактовать как способность к оперативному восстановлению ее адекватности объекту моделирования через процедуры структурной и/или параметрической корректировки для нее.

5. Формирование модели ДС сводится к решению четырех основных задач. А именно, требуется найти:

- 1) набор величин, характеризующих моделируемый объект;
- 2) класс (семейство) моделей, включающий искомую модель;
- 3) средства выбора конкретной модели в заданном классе (критерий адекватности модели и алгоритм ее поиска);
- 4) представительный (информативный) набор экспериментальных данных для формирования и тестирования модели.

Задача 1, т. е. формирование набора величин, характеризующих моделируемый объект, является предметом отдельного исследования, нацеленного на то, чтобы понять, каким образом содержательная трактовка задачи формализуется в математической модели (см. также замечания относительно примеров 1.1 и 1.2 на с. 39). Предметом дальнейшего рассмотрения являются задачи 2, 3 и 4.

6. Пусть имеется некоторая динамическая система \mathbb{S} , являющаяся объектом моделирования (рис. 1.1).

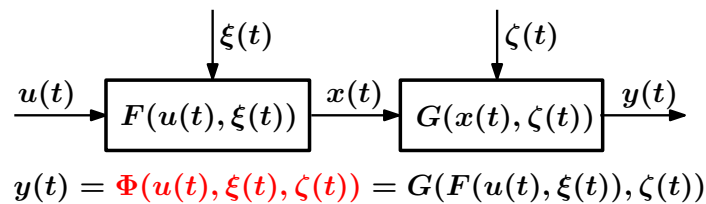


Рис. 1.1. Общая структура моделируемой динамической системы

Система \mathbb{S} воспринимает контролируемое $u(t)$ и неконтролируемое $\xi(t)$ воздействия. Под этими воздействиями \mathbb{S} изменяет свое состояние $x(t)$ в соответствии с реализуемым ею преобразованием (отображением) $F(u(t), \xi(t))$. В начальный момент времени $t = t_0$ состояние системы \mathbb{S} принимает значение $x(t_0) = x_0$.

Состояние $x(t)$ воспринимается измерителем (наблюдателем), реализующим преобразование $G(x(t), \zeta(t))$, и выдается в виде выхода системы \mathbb{S} , т. е. результатов наблюдения $y(t)$ за ее состоянием $x(t)$. Неидеальность измерителей состояния системы \mathbb{S} учитывается введением дополнительного неконтролируемого воздействия $\zeta(t)$ («измерительного шума»). Композиция отображений $F(\cdot)$ и $G(\cdot)$ описывает связь контролируемого входа $u(t) \in U$ системы \mathbb{S} с ее выходом $y(t) \in Y$, с учетом влияния неконтролируемых воздействий $\xi(t)$ и $\zeta(t)$ на рассматриваемую систему:

$$y = \Phi(u(t), \xi(t), \zeta(t)) = G(F(u(t), \xi(t)), \zeta(t)).$$

7. Пусть над системой \mathbb{S} выполнено N_P наблюдений

$$\{y_i\} = \Phi(u_i, \xi, \zeta), \quad i = 1, \dots, N_P, \quad (1.24)$$

в каждом из которых фиксировалось текущее значение контролируемого входного воздействия $u_i = u(t_i)$ и отвечающего ему выхода $y_i = y(t_i)$. Результаты $y(t_i)$, $t_i \in [t_0, t_f]$ этих наблюдений в совокупности с соответствующими значениями контролируемых входов u_i образуют набор из N_P упорядоченных пар:

$$\{(u_i, y_i)\}, \quad u_i \in U, \quad y_i \in Y, \quad i = 1, \dots, N_P. \quad (1.25)$$

Требуется найти, используя данные (1.25), такое приближение $\widehat{\Phi}(\cdot)$ для отображения $\Phi(\cdot)$, реализуемого системой \mathbb{S} , чтобы выполнялось условие

$$\begin{aligned} \|\widehat{\Phi}(u(t), \xi(t), \zeta(t)) - \Phi(u(t), \xi(t), \zeta(t))\| &\leq \varepsilon, \\ \forall u(t_i) \in U, \quad \forall \xi(t_i) \in \Xi, \quad \forall \zeta(t_i) \in Z, \quad t \in [t_0, t_f]. \end{aligned} \quad (1.26)$$

Таким образом, как это следует из (1.26), необходимо, чтобы искомое приближенное отображение $\widehat{\Phi}(\cdot)$ обладало требуемой точностью не только при воспроизведении наблюдений (1.25), но и для всех допустимых значений $u_i \in U$. Будем называть данное свойство отображения $\widehat{\Phi}(\cdot)$ *обобщающим*. Записи $\forall \xi(t_i) \in \Xi$ и $\forall \zeta(t_i) \in Z$ в (1.26) означают, что приближение $\widehat{\Phi}(\cdot)$ будет обладать требуемой точностью при условии, что в любой момент времени $t \in [t_0, t_f]$ неконтролируемые воздействия $\xi(t)$ на систему \mathbb{S} и измерительные шумы $\zeta(t)$ не выходят за допустимые пределы.

Отображение $\Phi(\cdot)$ соответствует рассматриваемому объекту моделирования (динамической системе \mathbb{S}), а отображение $\widehat{\Phi}(\cdot)$ будем далее именовать моделью данного объекта. Будем также далее считать, что относительно системы \mathbb{S} мы располагаем данными вида (1.25), а также, возможно, некоторыми знаниями об «устройстве» отображения $\Phi(\cdot)$, реализуемого рассматриваемой системой. При этом наличие данных указанного вида обязательно (как минимум, они необходимы для тестирования создаваемой модели $\widehat{\Phi}(\cdot)$), знаний же об отображении $\Phi(\cdot)$ может и не быть в наличии, либо они могут быть, но не будут использоваться при формировании модели $\widehat{\Phi}(\cdot)$.

8. Требуется уточнить, что понимается под нормой $\|\cdot\|$ в выражении (1.26), т.е. каким образом интерпретировать величину расхождения между результатами, которые дают отображения $\widehat{\Phi}(\cdot)$ и $\Phi(\cdot)$.

Можно понимать (1.26) как максимальное отклонение $\widehat{\Phi}(\cdot)$ от $\Phi(\cdot)$:

$$\|\widehat{\Phi}(u(t), \xi, \zeta) - \Phi(u, \xi, \zeta)\| = \max_{t_0 \leq t \leq t_n} |\widehat{\Phi}(u(t), \xi, \zeta) - \Phi(u(t), \xi, \zeta)|. \quad (1.27)$$

Второй, более употребительный, вариант использует в качестве оценки величины расхождения между $\widehat{\Phi}(\cdot)$ и $\Phi(\cdot)$ норму вида

$$\|\widehat{\Phi}(u, \xi, \zeta) - \Phi(u, \xi, \zeta)\| = \int_{t_0}^{t_n} [\widehat{\Phi}(u(t), \xi, \zeta) - \Phi(u(t), \xi, \zeta)]^2 dt. \quad (1.28)$$

9. Доступное число экспериментов, порождающих множество (1.25), является конечным. В силу этого, вместо (1.28) следует использовать один из возможных конечномерных вариантов данного выражения, например среднеквадратическое отклонение вида

$$\|\widehat{\Phi}(u, \xi, \zeta) - \Phi(u, \xi, \zeta)\| = \frac{1}{N_P} \sum_{i=0}^{N_P} [\widehat{\Phi}(u_i, \xi, \zeta) - \Phi(u_i, \xi, \zeta)]^2 \quad (1.29)$$

или вида

$$\|\widehat{\Phi}(u, \xi, \zeta) - \Phi(u, \xi, \zeta)\| = \sqrt{\frac{1}{N_P} \sum_{i=0}^{N_P} [\widehat{\Phi}(u_i, \xi, \zeta) - \Phi(u_i, \xi, \zeta)]^2}. \quad (1.30)$$

Тестирование отображения $\widehat{\Phi}(\cdot)$ для оценки его обобщающих свойств осуществляется на множестве упорядоченных пар, аналогичных (1.25)

$$\{\langle \tilde{u}_j, \tilde{y}_j \rangle\}, \quad \tilde{u} \in U, \quad \tilde{y} \in Y, \quad i = 1, \dots, N_T, \quad (1.31)$$

при этом необходимо, чтобы выполнялось условие $u_i \neq \tilde{u}_i, \forall i \in \{1, \dots, N_P\}, \forall j \in \{1, \dots, N_T\}$, т. е. все пары в множествах

$$\{\langle u_i, y_i \rangle\}_{i=1}^{N_P}, \quad \{\langle \tilde{u}_j, \tilde{y}_j \rangle\}_{j=1}^{N_T}$$

должны быть несовпадающими.

Ошибка на тестовом наборе (1.31) вычисляется аналогично тому, как это делается для обучающего набора (1.25)

$$\|\widehat{\Phi}(\tilde{u}, \xi, \zeta) - \Phi(\tilde{u}, \xi, \zeta)\| = \frac{1}{N_T} \sum_{j=0}^{N_T} [\widehat{\Phi}(\tilde{u}_j, \xi, \zeta) - \Phi(\tilde{u}_j, \xi, \zeta)]^2, \quad (1.32)$$

она может быть также представлена в виде

$$\|\widehat{\Phi}(\tilde{u}, \xi, \zeta) - \Phi(\tilde{u}, \xi, \zeta)\| = \sqrt{\frac{1}{N_T} \sum_{j=0}^{N_T} [\widehat{\Phi}(\tilde{u}_j, \xi, \zeta) - \Phi(\tilde{u}_j, \xi, \zeta)]^2}. \quad (1.33)$$

10. Теперь можно сформулировать проблему формирования модели динамической системы \mathbb{S} . Требуется построить модель $\widehat{\Phi}(\cdot)$, которая с требуемым уровнем точности будет воспроизводить отображение $\Phi(\cdot)$, реализуемое системой \mathbb{S} , т. е. модель $\widehat{\Phi}(\cdot)$, для которой величина погрешности моделирования (1.32) или (1.33) на тестовом наборе (1.31) не будет превышать заданного допустимого значения ε в (1.26). Это формирование должно основываться на данных (1.25), используемых для настройки (обучения) модели, а также (1.31), используемых для тестирования модели, кроме того, возможно, на знаниях относительно системы \mathbb{S} .

Предполагается, что при решении данной проблемы мы располагаем некоторым конечным или бесконечным семейством (набором) вариантов $\widehat{\Phi}_j(\cdot)$, $j = 1, 2, \dots$, среди которых необходимо выбрать наилучший в некотором смысле вариант $\widehat{\Phi}^*(\cdot)$. В связи с этим возникают следующие два вопроса:

- что представляет собой данное семейство вариантов $\widehat{\Phi}^{(F)} = \{\widehat{\Phi}_j(\cdot)\}$, $j = 1, 2, \dots$;
- каким образом выбрать из семейства $\widehat{\Phi}^{(F)}$ тот вариант $\widehat{\Phi}^*(\cdot)$, который удовлетворяет условию (1.26).

11. При формировании семейства вариантов искомой модели надо удовлетворить двум требованиям, которые, в общем случае, противоречат друг другу:

- семейство моделей $\{\widehat{\Phi}_j(\cdot)\}$, $j = 1, 2, \dots$ должно быть возможно более богатым, чтобы «было из чего выбирать»;
- это семейство должно быть устроено таким образом, чтобы максимально упростить процесс выбора модели $\widehat{\Phi}^*(\cdot)$ из него.

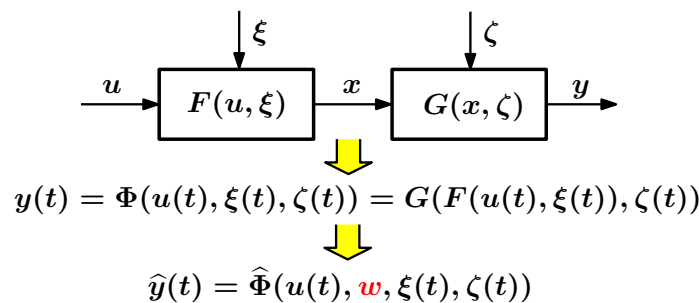


Рис. 1.2. Параметризация моделируемой динамической системы

В качестве основы для поиска решений, удовлетворяющих этим требованиям, в следующих разделах реализуется подход, ориентированный на эффективной структуризации и параметризации искомой модели $\widehat{\Phi}(\cdot)$, подразумевающий соответствующий выбор ее структуры и введение в нее необходимого числа варьируемых параметров $w \in W$ (рис. 1.2).

1.6.3 Основные проблемы, требующие решения при формировании модели динамической системы

При формировании модели ДС, независимо от того, в рамках какого подхода оно осуществляется, возникает ряд проблем, требующих решения. Как уже отмечалось выше, требуется сформировать:

- набор величин, характеризующих моделируемый объект;
- класс (семейство) моделей, включающий искомую модель;
- представительный (информативный) набор экспериментальных данных для формирования и тестирования модели;
- средства выбора конкретной модели из заданного класса (критерий адекватности модели и алгоритм ее поиска).

Ниже дается краткая характеристика этих проблем. Более подробно методы их решения в рамках развиваемого подхода рассматриваются в разд. 2 и 5, а примеры применения этих методов — в разд. 3 и 6.

1.6.3.1 Формирование набора величин, характеризующих моделируемый объект

Первое, что требуется сделать при формировании модели ДС, это сформировать набор величин, характеризующих данную систему. Соображения, учитываемые при этом, приводились выше, на с. 46. Данная проблема относится к постановочной части процесса формирования модели ДС и далее не рассматривается, считается, что задача уже поставлена, т. е. уже принято решение относительно того, какие величины должны учитываться в моделировании.

1.6.3.2 Формирование семейства моделей, включающий искомую модель

Для решения проблемы моделирования ДС требуется вначале сформировать некоторый набор вариантов (семейство) $\widehat{\Phi}^{(F)} = \{\widehat{\Phi}_j(\cdot)\}$, $j = 1, 2, \dots$ среди которых необходимо затем выбрать наилучший в некотором смысле вариант $\widehat{\Phi}^*(\cdot)$. Как уже отмечалось, при решении данной части проблемы моделирования ДС необходимо ответить на следующие два вопроса:

- что представляет собой искомое семейство вариантов $\widehat{\Phi}^{(F)} = \{\widehat{\Phi}_j(\cdot)\}$, $j = 1, 2, \dots$;
- каким образом выбрать из семейства $\widehat{\Phi}^{(F)}$ тот вариант $\widehat{\Phi}^*(\cdot)$, который удовлетворяет условию $\|\widehat{\Phi}(u(t), \xi, \zeta) - \Phi(u(t), \xi, \zeta)\| \leq \varepsilon$, $t \in [t_0, t_f]$, $\forall u \in U$, $\xi \in \Xi$, $\zeta \in Z$.

Основные идеи, с привлечением которых далее даются ответы на эти вопросы, состоят в следующем:

- при формировании набора вариантов $\widehat{\Phi}^{(F)}$ ключ — эффективная структуризация и параметризация данного семейства моделей;
- при выборе варианта $\widehat{\Phi}^*(\cdot)$ из набора $\widehat{\Phi}^{(F)}$ ключ — машинное (в том числе нейросетевое) обучение.

1.6.3.3 Формирование представительного набора экспериментальных данных для формирования и тестирования модели

Одна из важнейших составных частей процесса формирования модели ДС — это получение набора данных, с требуемой полнотой характеризующего поведение рассматриваемой системы. Как будет показано далее, от того, насколько информативен имеющийся обучающий набор, в очень большой степени зависит успех решения задачи моделирования.

1.6.3.4 Формирование средства выбора конкретной модели из заданного класса

После того, как сформировано семейство моделей для рассматриваемой ДС, а также получен представительный набор данных, характеризующих ее поведение, необходимо определиться с инструментом, позволяющим «извлечь» из этого семейства конкретную модель, удовлетворяющую определенному набору требований. В качестве такого инструмента в рамках рассматриваемого подхода вполне естественно использовать средства нейросетевого обучения.

1.7 Проблема адаптивности систем

Как отмечалось во введении, адаптивность является важнейшим инструментом, повышающим безопасность полета и живучесть ЛА, обеспечивающим реконфигурацию его системы управления. Понятие адаптивности ДС является многоуровневым. Рассмотрим содержание этих уровней, т. е. видов адаптации.

1.7.1 Виды адаптации

В зависимости от вида используемых механизмов (т. е. от конкретного вида правила Ψ^{AS}), будем выделять, основываясь главным образом на [52], следующие виды (иерархические

уровни) адаптации:

- параметрическая адаптация;
- структурная адаптация;
- адаптация объекта;
- адаптация целей управления.

Ниже дается краткая характеристика принципиальных особенностей этих видов адаптации.

1.7.1.1 Параметрическая адаптация

Параметрическая адаптация обеспечивается изменением значений компонент вектора настроечных параметров $\vartheta(t_i) \in \Theta$ системы \mathbb{S} (такими параметрами могут быть, например, коэффициенты усиления регулятора).

В данном случае считается, что правило Φ^S зависит не только от x, u, ξ, t , как было указано ранее, но еще и от $\vartheta(t_i) \in \Theta$, т. е. $\Phi^S = \Phi^S(\vartheta)$ представляет собой параметрическое семейство функций, фиксируя значение вектора $\vartheta(t_i) \in \Theta$, мы тем самым выбираем определенную функцию $\Phi^S = \Phi^S(x, u, \xi, t)$ из этого семейства. Правило $\Psi^S = \Psi^S(\lambda, \gamma, t)$ определяет значения $\vartheta(t_i)$, передаваемые в $\Phi^S = \Phi^S(\vartheta(t_i))$, что приводит к изменению характера реагирования системы \mathbb{S} на воздействия среды \mathcal{E} , т. е. к изменению ее поведения.

Возможные механизмы изменения значений вектора $\vartheta(t_i) \in \Theta$ параметров системы \mathbb{S} здесь пока не затрагиваются, они будут рассмотрены в последующих разделах.

Значения вектора $\vartheta(t_i) \in \Theta$ параметров системы \mathbb{S} могут быть кусочно-постоянными, т. е. их значения будут оставаться неизменными не для единственной «точки» $\langle \lambda(t_i), \gamma(t_i) \rangle \in \Lambda \times \Gamma$, а для целой подобласти $\Lambda_i \times \Gamma_i \subset \Lambda \times \Gamma$ такой области. Такого рода подход довольно распространен в системах управления²³.

Подстройка может быть также и непрерывной, когда каждой паре $\langle \lambda(t_i), \gamma(t_i) \rangle \in \Lambda \times \Gamma$ в общем случае будет соответствовать свое значение $\vartheta(t_i) \in \Theta$.

В биологии параметрической адаптации соответствует понятие *аккомодации*.

²³В англоязычной литературе этот подход именуется Gain Scheduling (GS), в отечественной — «программирование коэффициентов усиления регулятора».

1.7.1.2 Структурная адаптация

Далеко не всегда требуемой пластичности поведения системы \mathbb{S} можно добиться, варьируя только значения параметров системы $\vartheta(t_i) \in \Theta$. Следующий иерархический уровень адаптивных систем — это системы, способные к структурной адаптации, т. е. к изменению структуры (набора элементов системы \mathbb{S} , а также связей между этими элементами) применительно к меняющейся ситуации $\lambda(t_i) \in \Lambda^S$ и цели $\gamma(t_i) \in \Gamma^S$.

Простейший вариант — система \mathbb{S} с набором альтернативных по структуре вариантов $\Phi^S = \{\Phi_p\}^S$, $p = 1, \dots, N^P$, из них в данный момент времени t_i работает правило с некоторым номером p , значение которого определяется правилом $\Psi^S = \Psi^S(\lambda, \gamma, t)$. Более сложный, но и более интересный вариант связан с эволюционным изменением структуры системы \mathbb{S} под воздействием среды \mathcal{E} (и, возможно, некоторых других факторов).

В биологии именно механизм такого рода и называется адаптацией (под которой понимается эволюционное необратимое изменение генотипа системы). Обратимые варианты («подстройка параметров») в биологии именуют аккомодацией (приспособлением к изменившимся условиям).

1.7.1.3 Адаптация объекта

Вполне возможен случай, когда никакое варьирование структуры системы \mathbb{S} и, тем более, ее параметров $\vartheta(t_i) \in \Theta$, не позволяет добиться удовлетворения целей существования системы. Это вполне естественно, ведь потенциальные возможности любой системы не безграничны, пределы этих возможностей достаточно жестко заданы «конструкцией» системы. Если возник случай такого рода, то может быть вовлечен следующий уровень адаптации — адаптация объекта.

В разделе о системах был сформулирован тезис о том, что есть некая система — предмет нашего изучения, и есть все остальное, не входящее в эту систему — внешняя среда. Адаптация объекта заключается в пересмотре границы между объектом и средой.

Основная идея этого уровня адаптации состоит в том, что решение требуемой целевой задачи можно осуществлять не одной системой \mathbb{S} , как это было в предыдущих двух случаях, а совокупностью (группировкой) таких систем $\{\mathbb{S}_\mu\}$, $\mu = 1, \dots, N^\mu$, взаимодействующих между собой при решении поставленной задачи. Соответственно, вместо единичного правила $\Phi^S = \Phi^S(\mathbf{x}, \mathbf{u}, \boldsymbol{\xi}, t)$ получаем набор взаимодействующих правил $\Phi^S = \{\Phi_\mu\}^S$, принадлежащих

различным системам \mathbb{S}_μ , $\mu = 1, \dots, N^\mu$.

ПРИМЕР 1.4. Пусть решаемая задача — перехват воздушных целей, в том числе и групповых. Если область воздушного пространства и число целей в группе сравнительно невелики, то задача в ряде случаев может быть решена единственным истребителем-перехватчиком, обладающим ракетным вооружением и многоканальной системой обнаружения и сопровождения целей. Если указанные условия не выполняются, возможностей одного самолета уже недостаточно.

Выход из этой ситуации — образование группировки систем, нацеленной на совместное решение общей задачи. Пример такого подхода — комплекс перехвата МиГ-31 [48], в котором группа из четырех взаимодействующих самолетов данного типа контролирует воздушное пространство протяженностью по фронту 800–900 км. В варианте МиГ-31Б оборудование комплекса перехвата получило дополнительную возможность передавать в автоматическом режиме данные об обнаруженных истребителем целях наземным средствам ПВО для обеспечения стрельбы зенитных ракетных комплексов, т.е. здесь осуществлено дополнительное расширение группировки систем (комплекса перехвата).

ПРИМЕР 1.5. Создание первого советского комплекса перехвата Су-9 [48]. Создание этого комплекса, принятого на вооружение в 1960 г., явилось реакцией на то, что «изолированный» истребитель-перехватчик даже с ракетным вооружением не обеспечивал эффективного перехвата воздушных целей на средних и больших дальностях. Адаптационная реакция состояла в том, чтобы в комплекс средств, решающих поставленную задачу, включить помимо самолета Су-9 с бортовой РЛС ЦД-30 и управляемыми ракетами «воздух-воздух» РС-2-УС также и систему автоматизированного наземного наведения «Воздух».

1.7.1.4 Адаптация целей управления

Если адаптация системы \mathbb{S} не позволяет решить поставленную задачу, т.е. обеспечить достижение поставленных целей, то, вполне возможно, перед системой поставлены цели $\gamma \in \Gamma^S$, которые для нее являются недостижимыми. В этом случае остается возможность изменить цели управления так, чтобы они стали достижимыми. Выполнение этой операции осуществляется с помощью правила $\Omega^S = \Omega^S(\lambda, \gamma, t)$, на основе мотивационно-нормативного набора элементов $\Sigma^{IS} \subseteq \Sigma$. Адаптация целей — это, по-существу, адаптация потребностей субъекта управления.

ПРИМЕР 1.6. Пояснить суть адаптации целей можно следующим образом. Пусть перед

самоходным аппаратом, доставленным на некоторое небесное тело, поставлена задача обследовать конкретный объект. Может обнаружиться, что решение поставленной задачи требует слишком большого расхода ресурсов, что ставит под угрозу выполнение других задач экспедиции. В этом случае, исходя из общих установок (например, получить максимум возможных знаний об исследуемом небесном теле), система \mathcal{S} может заменить одну цель другой и найти для изучения объект, «похожий» на тот, который не удалось исследовать, или же отказаться вообще от данного пункта программы, переключившись на другие.

1.7.2 Общая характеристика проблемы адаптивности систем

1. Как известно [51], традиционная теория управления требует знания математической модели объекта, входящих в эту модель параметров и характеристик объекта, а также параметров и характеристик среды, в которой данный объект функционирует.

На практике эти требования далеко не всегда могут быть удовлетворены. Кроме того, в процессе функционирования параметры и характеристики объекта и среды могут значительно изменяться. В этих случаях традиционные методы часто дают неудовлетворительные результаты.

В связи с этим возникает потребность в построении управляющих систем, не требующих полного априорного знания объекта управления и условий его функционирования. Такая система должна быть в состоянии приспосабливаться к меняющимся свойствам и условиям функционирования объекта. Этим требованиям отвечают адаптивные системы [51, 54–62, 64–66], в которых текущая доступная информация используется не только для выработки управляющего воздействия (как в обычных неадаптивных системах), но также и для изменения (корректировки) алгоритма управления.

Принято выделять два основных класса адаптивных систем [51, 58, 61]:

- самонастраивающиеся системы, в которых в ходе работы структура алгоритма управления не меняется, а изменяются только его параметры;
- самоорганизующиеся системы, в которых в ходе работы изменяется не только параметры, но и структура алгоритма управления.

2. Традиционно при рассмотрении адаптивных систем [51, 54–56] неполное знание параметров и характеристик объекта управления, а также условий его функционирования трактуется как их принадлежность к некоторому классу Ξ , который определяется через факторы

неопределенности, существенные для рассматриваемой ДС.

Например, факторы неопределенности, связанные с задачей управления ЛА, можно определить через следующие три множества параметров:

$$\begin{aligned} W &= W_1 \times W_2 \times \dots \times W_p, \\ V &= V_1 \times V_2 \times \dots \times V_q, \\ \Xi &= \Xi_1 \times \Xi_2 \times \dots \times \Xi_r, \end{aligned} \quad (1.34)$$

где W_i , V_j , Ξ_k — области изменения значений величин w_i (определяют возможные значения параметров ЛА), v_j (определяют возможные значения характеристик ЛА), ξ_k (определяют возможные значения параметров атмосферы и атмосферных воздействий), соответственно, т. е.

$$\begin{aligned} w_i &\in W_i, \quad W_i = [w_i^{min}, w_i^{max}], \quad i = 1, 2, \dots, p, \\ v_j &\in V_j, \quad V_j = [v_j^{min}, v_j^{max}], \quad j = 1, 2, \dots, q, \\ \xi &\in \Omega_k, \quad \Xi_k = [\xi_k^{min}, \xi_k^{max}], \quad k = 1, 2, \dots, r. \end{aligned} \quad (1.35)$$

Конкретное сочетание параметров w_i , v_j , ξ_k образует кортеж ω_s длины $p + q + r$

$$\omega_s = \langle w_1, \dots, w_p, v_1, \dots, v_q, \xi_1, \dots, \xi_r \rangle, \quad s = 1, 2, \dots, p \cdot q \cdot r. \quad (1.36)$$

Все возможные сочетания значений факторов неопределенности, характеризующих задачу управления полетом ЛА, образуют совокупность Ω кортежей ω_s как декартово произведение множеств W , V , Ξ

$$\Omega = W \times V \times \Xi, \quad \omega_s \in \Omega \quad (1.37)$$

или же как подмножество $\tilde{\Omega} \subset \Omega$, если не все кортежи ω_s являются допустимыми. Для других видов ДС факторы неопределенности и их возможные сочетания определяются аналогичным образом.

Имея в виду это определение условий функционирования рассматриваемой ДС, задача адаптивного управления в традиционной постановке формулируется следующим образом: регулятор будет адаптивным в классе Ξ , если он спустя конечное время T_a , называемое временем адаптации, обеспечит выполнение цели управления.

1.7.3 Варианты схем адаптивного управления

1. Как уже отмечалось, система управления считается адаптивной, если в ней текущая информация о состоянии системы используется не только для выработки управляющего воз-

действия (как это имеет место в обычных, неадаптивных, системах), но также и для изменения (корректировки) алгоритма управления. В общем виде структуру адаптивной системы можно представить так, как это показано на рис. 1.3. Из него видно, что корректирующее воздействие $\xi(t)$ для регулятора вырабатывается с помощью механизма адаптации, использующего в качестве входной информации значения задающего воздействия $r(t)$, управления $u(t)$, выхода объекта $y(t)$, а также дополнительную информацию «внешнего» характера $\psi(\lambda)$, $\lambda \in \Lambda$, которую следует учитывать при выработке корректирующего воздействия (например, скорость и высота полета летательного аппарата в задаче управления его угловым движением).

2. Возможны различные частные варианты данной схемы, отличающиеся друг от друга составом входной информации, используемой при выработке корректирующего воздействия $\xi(t)$. Один из таких вариантов состоит в том, что корректировка осуществляется только на основе «внешней» информации $\psi(\lambda)$, $\lambda \in \Lambda$; он именуется программированием коэффициентов регулятора (GS-подходом). Принципиальное отличие данного подхода (рис. 1.4) от полного варианта адаптационной схемы (рис. 1.3) состоит в том, что в GS-подходе значения корректирующих воздействий ξ как функция от $\psi(\lambda)$ должны быть вычислены заранее (off-line), после чего данная функция используется в неизменном виде в процессе управления объектом. В полном варианте адаптационной схемы алгоритм корректировки перестает быть неизменным, он уточняется непосредственно (on-line) в ходе работы системы.

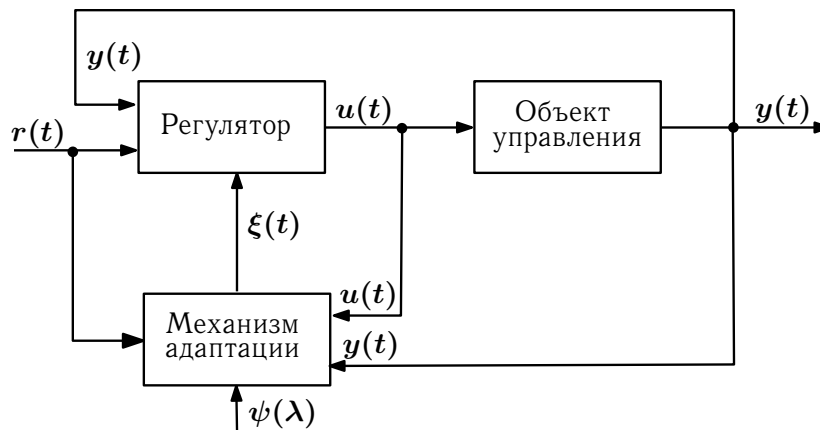


Рис. 1.3. Схема управляемой системы с подстраиваемым законом управления, реализуемым регулятором: $r(t)$ — задающее воздействие; $u(t)$ — управление; $y(t)$ — выход объекта управления; $\xi(t)$ — корректирующее воздействие для регулятора; $\psi(\lambda)$, $\lambda \in \Lambda$ — дополнительная информация, которую следует учитывать при выработке корректирующего воздействия (например, скорость и высота полета летательного аппарата в задаче управления его угловым движением)

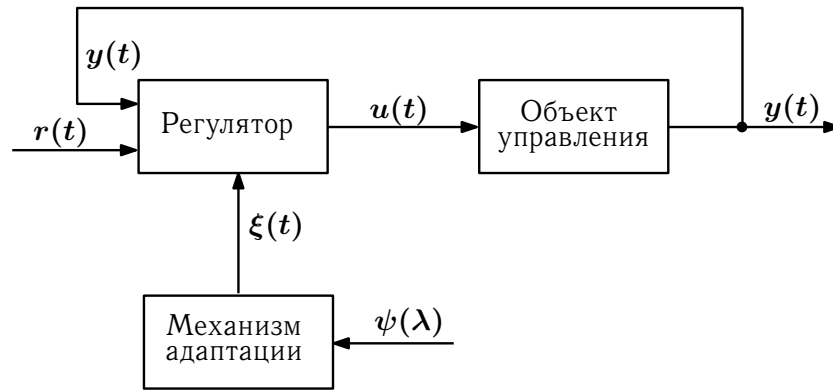


Рис. 1.4. Схема корректировки параметров закона управления, реализуемого регулятором по схеме с программированием его коэффициентов (GS — Gain Scheduling): $r(t)$ — задающее воздействие; $u(t)$ — управление; $y(t)$ — выход объекта управления; $\xi(t)$ — корректирующее воздействие для регулятора; $\psi(\lambda)$, $\lambda \in \Lambda$ — информация, на основании которой осуществляется выработка корректирующего воздействия

Несмотря на ограниченные адаптационные возможности GS-подхода, он довольно часто используется на практике. Например, именно по такой схеме было организовано управление испытательными полетами экспериментального гиперзвукового ЛА X-43 [45].

3. Схемы адаптивного управления принято разделять на два основных вида: прямое адаптивное управление и не прямое адаптивное управление [51, 57–62, 66, 70].

Схемы прямого адаптивного управления часто основываются на использовании некоторой эталонной модели (ЭМ), задающей требуемый характер поведения рассматриваемой системы²⁴. Структура таких систем показана на рис. 1.5. В системах прямого адаптивного управления параметры регулятора $\theta_c(t)$ корректируются по алгоритму, реализуемому законом адаптации, который вычисляет значения производной $\dot{\theta}_c(t)$ или разности $\theta_c(t+1) - \theta_c(t)$. Это вычисление основывается непосредственно на значении ошибки слежения $\varepsilon(t) = y(t) - y_m(t)$.

В системах непрямого адаптивного управления, структура которых показана на рис. 1.6, параметры регулятора $\theta_c(t)$ вычисляются с помощью уравнения связи (отображения вида $\hat{\theta}_p(t) \rightarrow \theta_c(t)$) на основании оценок $\hat{\theta}_p(t)$ для параметров объекта $\theta_p(t)$. Оценки $\hat{\theta}_p(t)$ вырабатываются оперативно, в процессе функционирования объекта, путем вычисления значения производной $\dot{\hat{\theta}}_p(t)$ или разности $\hat{\theta}_p(t+1) - \hat{\theta}_p(t)$.

В обеих схемах адаптивного управления, прямой и не прямой, основная идея состоит в

²⁴Цель управления при этом состоит в том, чтобы поведение ДС максимально приблизить к поведению, определяемому эталонной моделью. Корректировка цели управления, упоминавшаяся в разд. 1.7.1.4, может быть осуществлена в данном случае заменой одной ЭМ на другую, имеющейся в составе ДС.

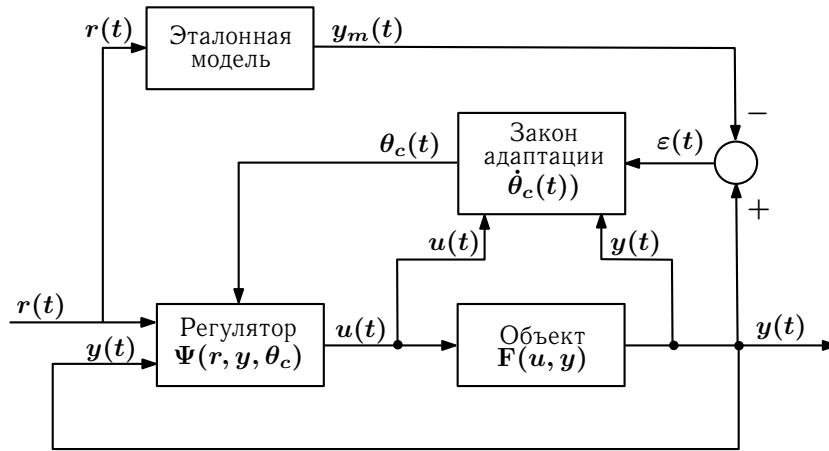


Рис. 1.5. Прямая схема адаптивного управления: $r(t)$ — задающее воздействие; $u(t)$ — управление; $y(t)$ — выход объекта управления; $y_m(t)$ — выход эталонной модели; $\theta_c(t)$ — корректируемые параметры регулятора; $\varepsilon(t) = y(t) - y_m(t)$ — расхождение между выходами объекта и эталонной модели

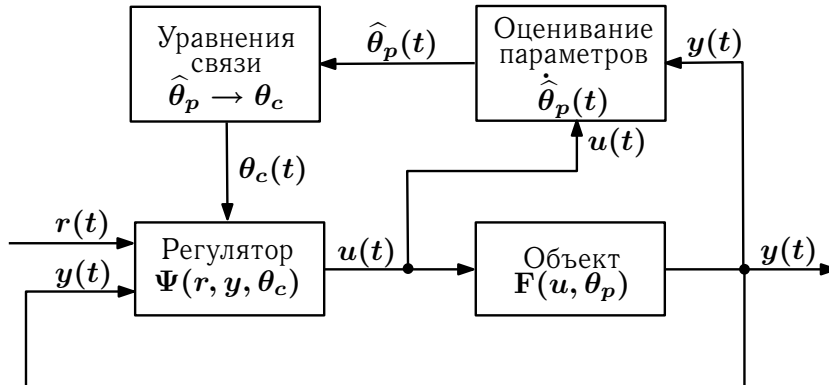


Рис. 1.6. Непрямая схема адаптивного управления: $r(t)$ — задающее воздействие; $u(t)$ — управление; $y(t)$ — выход объекта управления; $\theta_p(t)$ — оцениваемые параметры объекта; $\hat{\theta}_p(t)$ — оценка параметров объекта; $\theta_c(t)$ — корректируемые параметры регулятора

том, что идеальные значения параметров регулятора (при прямом адаптивном управлении) или объекта (при непрямом адаптивном управлении) используются так, как если бы это были параметры реального регулятора или объекта, соответственно. Вследствие того, что реальные значения указанных параметров неизбежно отличаются от идеальных, возникает ошибка, ухудшающая качество управления. Один из подходов к компенсации этой ошибки излагается ниже. Он состоит в том, чтобы данную ошибку трактовать как возмущающее воздействие на систему и парировать это воздействие, вводя в систему компенсирующий контур.

1.7.4 Роль моделей в проблеме адаптивного управления

1. Критически важную роль моделей при решении задач, связанных с адаптивными ДС, можно проиллюстрировать на двух примерах, приводимых ниже. Подробное рассмотрение этого вопроса будет проведено в разд. 4.

На рис. 1.7 показана общая схема нейросетевого адаптивного управления с эталонной моделью (MRAC — Model Reference Adaptive Control). В данном случае роль НС-модели объекта состоит в том чтобы обеспечить преобразование измеряемой ошибки на выходе системы в ошибку на выходе нейроконтроллера, значение которой необходимо для корректировки параметров этого контроллера.

2. Второй пример роли модели в адаптивном управлении ДС показан на рис. 1.8, где представлена общая схема нейросетевого адаптивного прогнозирующего управления (MPC — Model Predictive Control). В данном случае точная и быстродействующая модель объекта необходима для того, чтобы прогнозировать поведение объекта при различных управляющих воздействиях с тем, чтобы выбрать управление, наилучшее для данных конкретных условий.

3. Итак, при решении задач, связанных с адаптивными системами, ключевую роль играют модели объектов. На использовании этих моделей основано решение ряда важнейших подза-

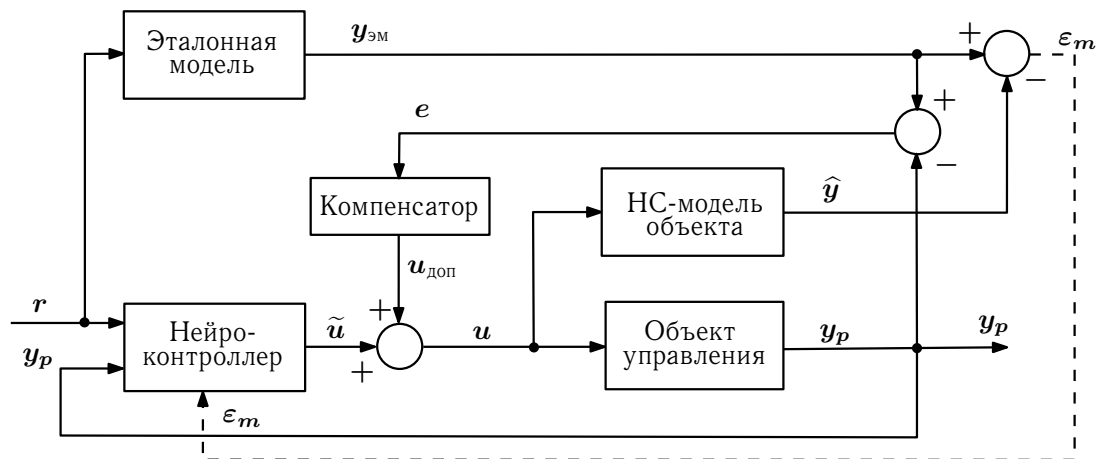


Рис. 1.7. Общая схема нейросетевого адаптивного управления с эталонной моделью: \tilde{u} — управление на выходе нейроконтроллера, $u_{\text{доп}}$ — добавочное управление от компенсатора, u — результирующее управление, y_p — выход объекта управления, \hat{y} — выход нейросетевой модели объекта управления; $y_{\text{ЭМ}}$ — выход эталонной модели; ε — расхождение между выходами объекта управления и эталонной модели; ε_m — расхождение между выходами объекта управления и НС-модели; r — задающее воздействие

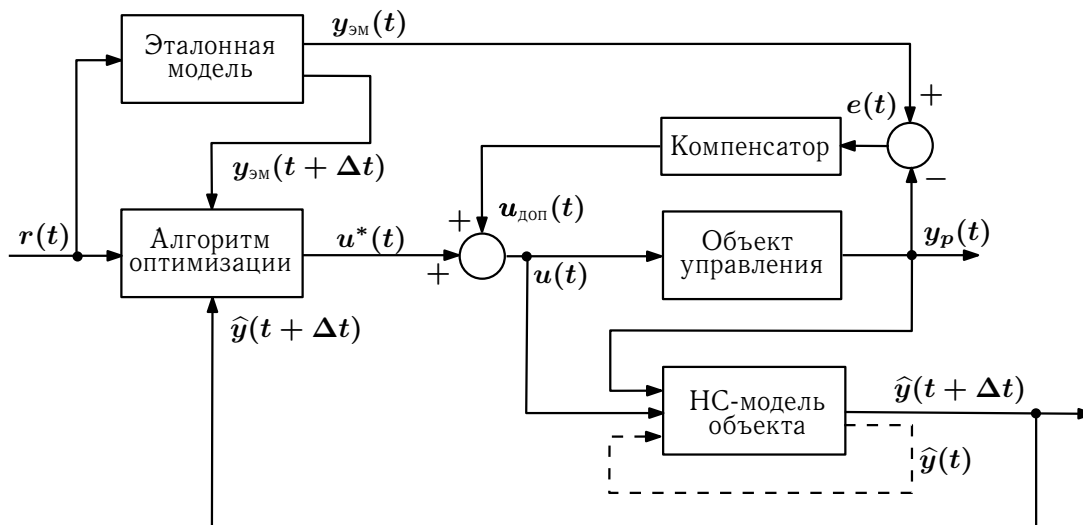


Рис. 1.8. Общая схема нейросетевого адаптивного управления с прогнозирующей моделью: u^* — управление на выходе алгоритма оптимизации, $u_{доп}$ — добавочное управление от компенсатора, u — результирующее управление, y_p — выход объекта управления, \hat{y} — выход нейросетевой модели объекта управления; $y_{эм}$ — выход эталонной модели; ε — расхождение между выходами объекта управления и эталонной модели; r — задающее воздействие

дач, например:

1. Задача анализа поведения ДС как подзадача в схеме с прогнозирующей моделью (прогнозирование поведения ДС для выработки управляющего воздействия на объект).
2. Адаптивное управление по схеме с эталонной моделью — модель объекта как «технологическая среда» для передачи и преобразования ошибки на выходе ДС в ошибку на выходе нейроконтроллера.
3. Задача реконфигурации системы управления — использование модели нормативного поведения объекта для выявления факта возникновения нештатной ситуации.

1.7.5 Нейросетевая реализация алгоритмов адаптивного моделирования и управления

1. Искусственные нейронные сети с успехом применяются для решения задач идентификации динамических систем, а также для управления такими системами [81, 138–141].

Привлекательным инструментом решения указанных выше задач являются многослойные сети персептронного типа в силу того, что они обладают свойствами универсального аппроксиматора [54, 81, 82].

Аппроксимирующие способности НС-моделей с динамическими алгоритмами обучения позволяют моделировать сложные нелинейные динамические объекты управления в виде прямых и инверсных моделей, основываясь на измерениях входов и выходов рассматриваемого объекта.

2. Существующий в настоящее время арсенал нейросетевых средств позволяет решать задачи идентификации и управления как в ходе проектирования систем управления (с сохранением затем неизменными полученных алгоритмов управления), так и непосредственно в процессе функционирования системы управления, подстраивая алгоритм управления под меняющуюся ситуацию, в том числе и при возникновении нештатных ситуаций (отказы оборудования, повреждения конструкции и т. п.). Другими словами, имеется возможность создания как неадаптивных, так и адаптивных вариантов нейросетевых систем управления. Применение динамических схем обучения сетей рассматриваемого класса дает реальную возможность создавать адаптивные системы управления, позволяющие обеспечить эффективную эксплуатацию сложных систем в условиях разнообразных неопределенностей.

3. Существует значительное число схем адаптивного управления, в том числе и в нейросетевом исполнении [53,54,82]. К числу наиболее популярных из них можно отнести адаптивное управление с эталонной моделью, адаптивное управление с прогнозирующей моделью, адаптивное управление с инверсной моделью, а также адаптивное управление на основе обратной задачи динамики. В разд. 3, 4 и 6 рассматривается использование этих схем для решения задачи управления движением летательных аппаратов различных классов.

Среди нейронных сетей традиционного типа, т. е. имеющих структуру типа «черный ящик», применительно к моделированию управляемых динамических систем значительной популярностью пользуются две сети, а именно, NARX и NARMAX, реализующие нелинейную авто-регрессию с внешними входами, в качестве которых выступают управляющие сигналы ДС.

В разд. 4 показано, что эти НС-структуры могут быть с успехом использованы для решения задач формирования адаптивных ДС, как в части их законов управления, так и в части моделей объектов управления. Однако исследование возможностей традиционных НС-структур показывает, что у них имеется ряд ограничений, препятствующих эффективному решению задач адаптивного управления и моделирования. Разд. 5 и 6 посвящены способам устранения этих ограничений за счет введения нового класса НС-моделей, имеющих структуру типа «серый ящик».

В качестве одной из прикладных задач, на которых демонстрируются возможности НС-

подхода к реализации адаптивной ДС, используется задача управления продольным угловым движением ЛА. Эта задача, как будет показано далее, представляет значительный интерес с точки зрения оценки возможностей НС-структур традиционного типа («черный ящик»). Как будет показано далее, такие НС-структуры имеют серьезные ограничения по уровню сложности решаемых прикладных задач. Важность задачи управления продольным угловым движением состоит в том, что она как раз и является такого рода «граничной» задачей. Если задача, требующая решения, превосходит ее по сложности, то методы, основанные на традиционных НС-структурах, непригодны для ее решения. Соответствующие результаты, иллюстрирующие это положение, будут представлены в разд. 4 и 6.

2 Нейросетевой подход к задачам моделирования и управления систем

2.1 Порождающий подход к формированию НС-моделей

2.1.1 Структура порождающего подхода

1. В качестве гибкого инструмента формирования моделей ДС перспективным представляется порождающий подход, широко используемый в прикладной и вычислительной математике, развитый с привлечением идей НС-моделирования.

Порождающий подход далее трактуется следующим образом. Класс моделей, который содержит искомую (формируемую) модель ДС, можно трактовать как совокупность средств, порождающих модели ДС, удовлетворяющие заданным требованиям. Основные требования к данному комплексу средств состоят, во-первых, в том, что он должен порождать потенциально богатый класс моделей (т. е. он должен обеспечивать «богатство выбора») и, во-вторых, должен иметь возможно более простое «устройство», чтобы реализация данного класса моделей не была «неподъемной» проблемой. Два этих требования, вообще говоря, являются взаимоисключающими, каким образом и какими средствами обеспечить приемлемый баланс между ними, рассматривается далее в этом разделе.

2. Чтобы «порождать» какие-либо модели, требуется иметь в своем распоряжении:

- базис, т. е. набор элементов, из которых формируются модели;
- правила, используемые для формирования моделей путем соответствующего комбинирования элементов базиса:
 - правила структуризации формируемых моделей;
 - правила параметрической настройки формируемых моделей;

Один из вариантов порождающего подхода²⁵ состоит в том, что искомая зависимость $y(x)$ представляется в виде линейной комбинации базисных функций $\varphi_i(x)$, $i = 1, \dots, n$:

$$y(x) = \varphi_0(x) + \sum_{i=1}^n \lambda_i \varphi_i(x), \quad \lambda_i \in \mathcal{R}. \quad (2.1)$$

Набор функций $\{\varphi_i(x)\}$, $i = 1, \dots, n$ будем именовать функциональным базисом (ФБ), а конструкцию вида (2.1) — разложением функции $y(x)$ по функциональному базису $\{\varphi_i(x)\}_{i=1}^n$.

²⁵Примеры других вариантов — порождающие грамматики из теории формальных грамматик и языков [71, 72], синтаксический подход к описанию паттернов в теории распознавания образов [73].

Формирование разложения по ФБ путем варьирования настраиваемых параметров (коэффициенты λ_i в разложении (2.1)) будет рассматриваться далее как средство порождения решений (каждой конкретной комбинации значений параметров λ_i соответствует свое решение). Правило комбинирования элементов ФБ в случае (2.1) представляет собой взвешенное суммирование этих элементов.

3. Приведем несколько примеров функциональных разложений, часто используемых в тематическом моделировании.

ПРИМЕР 2.1. Ряд Тейлора:

$$F(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)^2 + \dots + a_n(x - x_0)^n + \dots \quad (2.2)$$

Базис данного разложения — это $\{(x - x_0)^i\}_{i=0}^{\infty}$, правило комбинирования элементов ФБ — взвешенное суммирование.

ПРИМЕР 2.2. Ряд Фурье:

$$F(x) = \sum_{i=0}^{\infty} (a_i \cos(ix) + b_i \sin(ix)). \quad (2.3)$$

Базис данного разложения — это $\{\cos(ix), \sin(ix)\}_{i=0}^{\infty}$, правило комбинирования элементов ФБ — взвешенное суммирование.

ПРИМЕР 2.3. Разложение Галеркина:

$$y(x) = u_0(x) + \sum_{i=1}^n c_i u_i(x). \quad (2.4)$$

Базис данного разложения — это $\{u_i(x)\}_{i=0}^n$, правило комбинирования элементов ФБ — взвешенное суммирование.

Во всех этих примерах порождаемые решения (линейные комбинации элементов базиса) — параметризованные, параметрами являются числовые величины, на которые умножаются элементы ФБ.

2.1.2 Сетевое представление функциональных разложений

1. Функциональным разложениям можно дать сетевую трактовку, позволяющую выявить общие черты и различия между их отдельными вариантами. Такая трактовка обеспечивает в дальнейшем простой переход к НС-моделям, а также позволяет установить взаимосвязи между моделями традиционного типа и НС-моделями.

Структурное представление функциональной зависимости от одной переменной как линейной и нелинейной комбинации элементов базиса $f_i(x)$, $i = 1, \dots, n$ показано на рис. 2.1a и рис. 2.1b, соответственно.

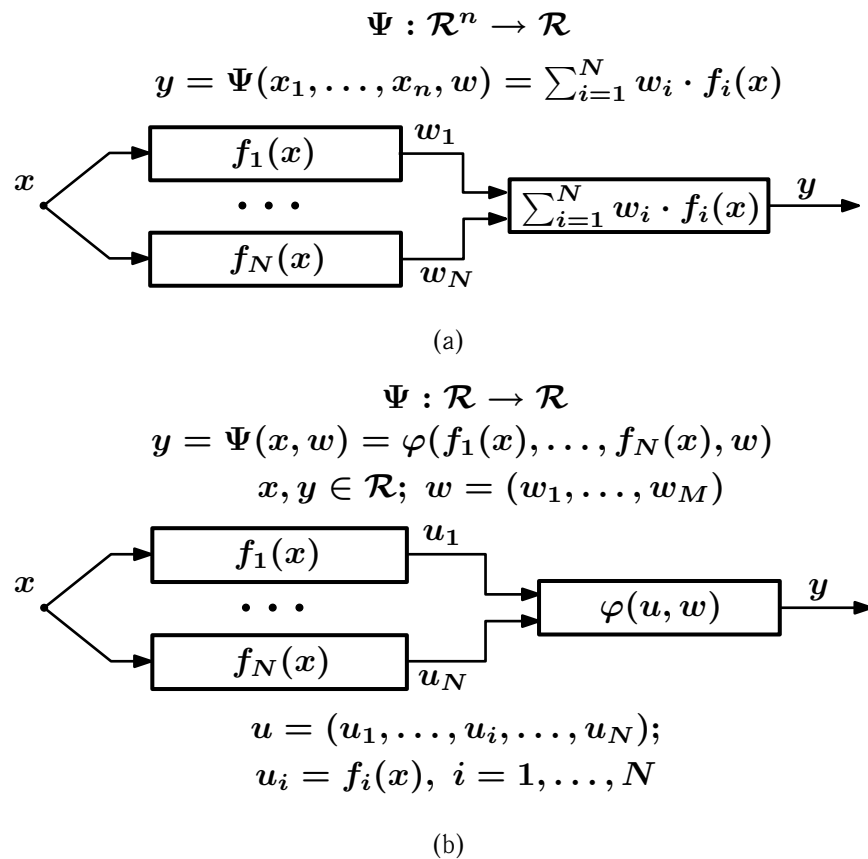
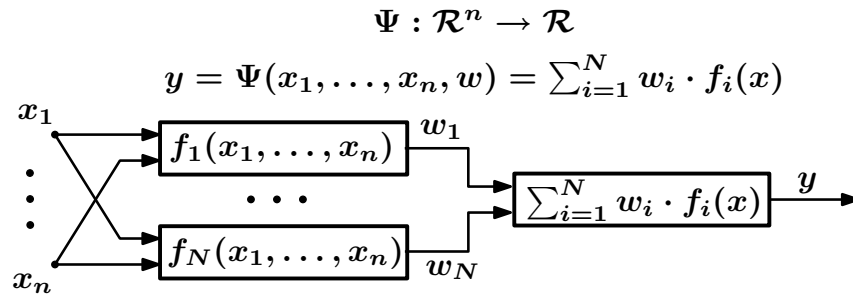


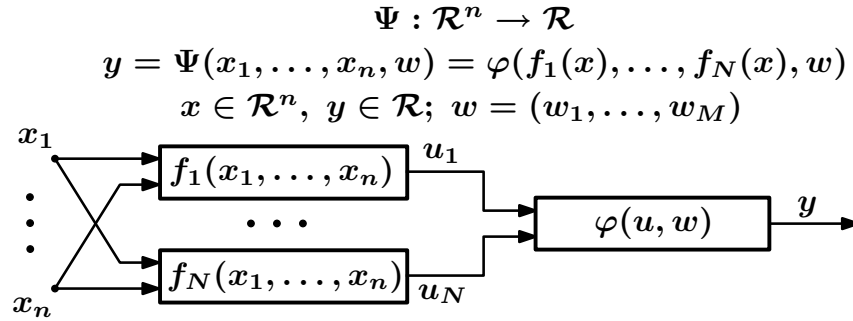
Рис. 2.1. Функциональная зависимость от одной переменной как линейная (a) и нелинейная (b) комбинация элементов базиса $f_i(x)$, $i = 1, \dots, n$

Аналогично, для скалярнозначной функциональной зависимости от нескольких переменных как линейной и нелинейной комбинации элементов базиса $f_i(x_1, \dots, x_n)$, $i = 1, \dots, N$ структурное представление приведено, соответственно, на рис. 2.2a и рис. 2.2b.

Векторнозначная функциональная зависимость от нескольких переменных как линейная комбинация элементов базиса $f_i(x_1, \dots, x_n)$, $i = 1, \dots, N$ в сетевом представлении показана на рис. 2.3. Нелинейная комбинация представляется аналогичным образом: вместо правила комбинирования $\sum_{i=1}^N (\cdot)$ в ней используются нелинейные правила $\varphi_i(f_1(x), \dots, f_m(x))$, $i = 1, \dots, m$; $x = x_1, \dots, x_m$.



(a)



$$u = (u_1, \dots, u_i, \dots, u_N)$$

$$u_i = f_i(x_1, \dots, x_n), i = 1, \dots, N$$

(b)

Рис. 2.2. Скалярнозначная функциональная зависимость от нескольких переменных как линейная (а) и нелинейная (б) комбинация элементов базиса $f_i(x_1, \dots, x_n)$, $i = 1, \dots, N$

2. Перечисленные выше функциональные разложения традиционного вида в общем виде записываются как

$$y(x) = F(x_1, x_2, \dots, x_n) = \sum_{i=0}^m \lambda_i \varphi_i(x_1, x_2, \dots, x_n). \quad (2.5)$$

Здесь функция $F(x_1, x_2, \dots, x_n)$ — (линейная) комбинация элементов базиса $\varphi_i(x_1, x_2, \dots, x_n)$.

Разложение вида (2.5) имеют следующие характерные черты:

- формируемое разложение является одноуровневым;
- функции $\varphi_i : \mathcal{R}^n \rightarrow \mathcal{R}$ как элементы базиса обладают ограниченной гибкостью (с вариативностью типа смещение, сжатие/растяжение), либо являются фиксированными.

Такая ограниченная гибкость традиционного ФБ в сочетании с одноуровневым характером разложения резко сужают возможности получить «достаточно хорошую» модель²⁶.

²⁶На интуитивном уровне «хорошая модель» — это модель с обобщающими свойствами, адекватными решаемой прикладной задаче; см. также разд. 1.6.2.

$$\begin{aligned} \Psi : \mathcal{R}^n &\rightarrow \mathcal{R}^m \\ y &= \Psi(x, W); \quad x \in \mathcal{R}^n, \quad y \in \mathcal{R}^m \\ W &= (w^{(1)}, \dots, w^{(m)}) \\ w^{(j)} &= (w_1^{(j)}, \dots, w_N^{(j)}); \quad j = 1, \dots, m \\ y_j &= \sum_{i=1}^N w_i^{(j)} f_i(x); \quad j = 1, \dots, m \end{aligned}$$

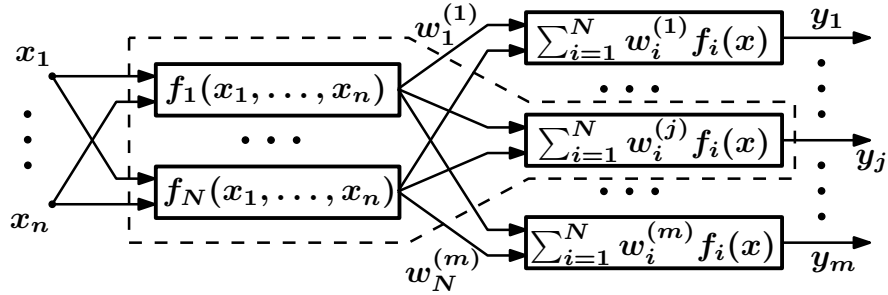


Рис. 2.3. Векторнозначная функциональная зависимость от нескольких переменных как линейная комбинация элементов базиса $f_i(x_1, \dots, x_n)$, $i = 1, \dots, N$

2.1.3 Многоуровневые настраиваемые функциональные разложения

1. Как отмечалось в предыдущем разделе, возможности получить «хорошую» модель ограничиваются одноуровневой структурой и негибким базисом традиционных разложений. По этой причине вполне естественным будет для преодоления этих недостатков построить такую модель, которая будет обладать требуемой гибкостью (и, соответственно, требуемой изменчивостью порождаемых вариантов) за счет формирования ее в виде многоуровневой сетевой структуры, а также подходящей параметризации элементов этой структуры.

На рис. 2.4 показано, как может быть построено многоуровневое настраиваемое функциональное разложение. Здесь видно, что в таком варианте настройка разложения осуществляется не только путем варьирования коэффициентов линейной комбинации, как в разложениях типа (2.5). Теперь параметризованными являются и элементы ФБ, т. е. по ходу решения задачи ФБ подстраивается так, чтобы получить приемлемую, в смысле критерия (1.26), модель ДС.

Переход от одноуровневого разложения к многоуровневому состоит, как видно из рис. 2.4, в том, что каждый элемент $\varphi_j(\mathbf{v}, \mathbf{w}^\varphi)$, $j = 1, \dots, M$ подвергается разложению по некоторому ФБ $\{\psi_k(\mathbf{x}, \mathbf{w}^\psi)\}$, $j = 1, \dots, K$. Аналогичным образом можно построить разложение элементов $\psi_k(\mathbf{x}, \mathbf{w}^\psi)$ еще по какому-либо ФБ и так далее, требуемое число раз, что дает сетевую структуру с необходимым числом уровней, а также требуемую параметризацию элементов ФБ.

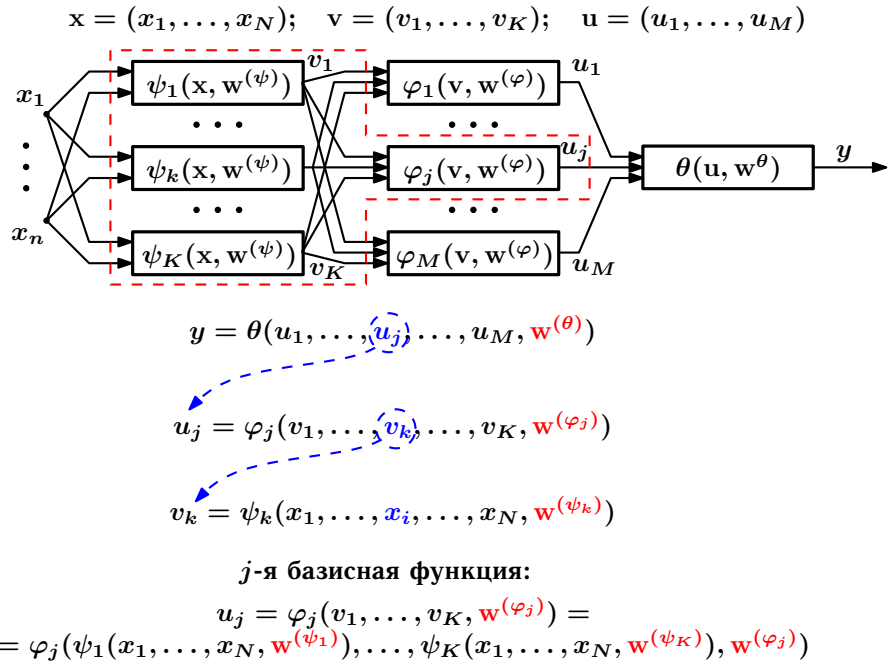


Рис. 2.4. Многоуровневое настраиваемое функциональное разложение

2.1.4 Функциональные и нейронные сети

1. Итак, модель можно интерпретировать как разложение по функциональному базису (2.5), каждый из элементов $\varphi_i(x_1, x_2, \dots, x_n)$ которого осуществляет преобразование n -мерного входа $x = (x_1, x_2, \dots, x_n)$ в скалярный выход y .

Можно выделить следующие виды элементов функционального базиса:

- элемент ФБ как целостное (одноэтапное) отображение $\varphi_i : \mathcal{R}^n \rightarrow \mathcal{R}$, непосредственно преобразующее n -мерный вход $x = (x_1, x_2, \dots, x_n)$ в скалярный выход y ;
- элемент ФБ как композиционное (двухэтапное) отображение n -мерного входа $x = (x_1, x_2, \dots, x_n)$ в скалярный выход y .

В двухэтапном (композиционном) варианте на первом этапе выполняется отображение $\mathcal{R}^n \rightarrow \mathcal{R}$, «сжимающее» векторный вход $x = (x_1, x_2, \dots, x_n)$ в промежуточный скалярный выход v , который на втором этапе дополнительно обрабатывается выходным отображением $\mathcal{R} \rightarrow \mathcal{R}$ для получения выхода y (рис. 2.5).

В зависимости от того, какие из этих элементов ФБ применяются при формировании сетевых моделей, получаются следующие основные виды этих моделей:

- Одноэтапное отображение $\mathcal{R}^n \rightarrow \mathcal{R}$ — функциональные сети.
- Двухэтапное отображение $\mathcal{R}^n \rightarrow \mathcal{R} \rightarrow \mathcal{R}$ — нейронные сети.

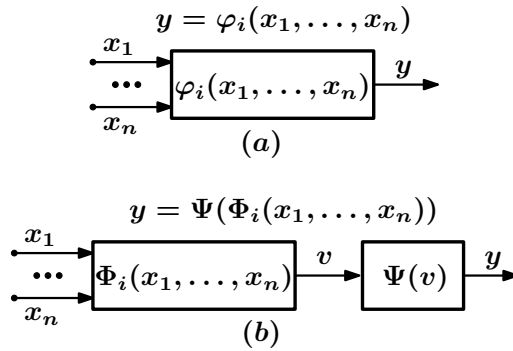


Рис. 2.5. Элемент функционального базиса, преобразующий n -мерный вход $x = (x_1, x_2, \dots, x_n)$ в скалярный выход y : (a) одноэтапное отображение $\mathcal{R}^n \rightarrow \mathcal{R}$; (b) двухэтапное (композиционное) отображение $\mathcal{R}^n \rightarrow \mathcal{R} \rightarrow \mathcal{R}$

Элемент композиционного типа, т. е. двухэтапное отображение n -мерного входа в скалярный выход — это нейрон, он характерен для функциональных разложений нейросетевого типа и является «фирменной чертой» таких разложений, другими словами, НС-моделей всех видов.

2.2 Структурная организация НС-моделей

2.2.1 Слоистая структурная организация НС-модели

1. Будем считать, что НС-модели в общем случае имеет слоистую структуру. Это означает, что вся совокупность элементов-нейронов, составляющих НС-модели, разделена на непересекающиеся подмножества, которые будем именовать слоями. Для слоев НС-модели введем обозначения $L^{(0)}, L^{(1)}, \dots, L^{(p)}, \dots, L^{(N_L)}$.

Слоистая организация НС-модели определяет логику срабатывания ее нейронов, эта логика будет различной для разных структурных вариантов сети. При этом имеет место следующая специфика в работе слоистой НС-модели²⁷ — нейроны, входящие в НС-модели, срабатывают послойно, т. е. пока не сработают все нейроны p -го слоя, нейроны $(p + 1)$ -го слоя не вступают в работу. Общий вариант, определяющий правила срабатывания нейронов в НС-модели, будет рассмотрен далее.

2. В самом простом варианте структурной организации слоистых сетей все слои $L^{(p)}$, пронумерованные числами от 0 до N_L , срабатывают в порядке следования их номеров. Это означает, что пока не сработают все нейроны, входящие в слой с номером p , нейроны из слоя

²⁷Для случая, когда слои идут в порядке следования их номеров, обратные связи между слоями отсутствуют. В таком случае слои будут срабатывать последовательно и однократно.

$(p + 1)$ находятся в состоянии ожидания. В свою очередь, p -й слой может начинать работу, только если уже сработали все нейроны $(p - 1)$ -го слоя.

Визуально такую структуру можно представить как «стопку слоев», упорядоченных по их номерам. В простейшем варианте эта «стопка» выглядит так, как это показано на рис. 2.6. Здесь слой $L^{(0)}$ является входным, элементы которого представляют собой компоненты вектора, поступающего на вход НС-модели.

Любой слой $L^{(p)}$, $1 \leq p < N_L$, связан с двумя соседними слоями: от предшествующего слоя $L^{(p-1)}$ он получает свои входы, а последующему слою $L^{(p+1)}$ передает свои выходы. Исключением является слой $L^{(N_L)}$, последний в НС (выходной слой), который не имеет слоя, следующего за ним. Выходы слоя $L^{(N_L)}$ являются выходами сети в целом. Слои $L^{(p)}$, $0 < p < N_L$ принято именовать скрытыми.

Поскольку НС, показанная на рис. 2.6а, является сетью прямого распространения, все связи между ее слоями идут строго от слоя $L^{(0)}$ к слою $L^{(N_L)}$ без «перескоков» через соседние слои и возвратов назад (обратных связей).

Для сетей рассматриваемого вида принимается также, что любая пара нейронов, между которыми установлена связь, относится к разным слоям. Другими словами, нейроны в пределах любого из обрабатывающих слоев $L^{(p)}$, $p = 1, \dots, N_L$, не имеют никаких связей между собой. Вариант, в котором такого рода связи, называемые латеральными, имеются в НС, будет рассмотрен отдельно.

3. Схема, показанная на рис. 2.6а, может быть развита в направлении усложнения структуры связей в ней.

Первый из возможных вариантов такого развития состоит во введении в структуру НС обратной связи, которая полученный выход сети (т. е. выход слоя $L^{(N_L)}$) передает «назад», на вход НС, точнее на вход ее первого обрабатывающего слоя $L^{(1)}$, как это показано на рис. 2.6б.

На рис. 2.6с показан другой вариант введения обратной связи в слоистую сеть, в котором обратная связь идет от выходного слоя $L^{(N_L)}$ к произвольному слою $L^{(p)}$, $1 < p < N_L$. Данный вариант можно трактовать также как объединение (последовательное соединение) НС прямого распространения (слои $L^{(1)}, \dots, L^{(p-1)}$) и сети с обратной связью типа показанной на рис. 2.6б (слои $L^{(p)}, \dots, L^{(N_L)}$).

Наиболее общий вариант введения обратной связи в структуру типа «стопка слоев» показан на рис. 2.6д. Здесь обратная связь идет от некоторого «внутреннего» (скрытого) слоя $L^{(q)}$, $1 < q < N_L$ к слою $L^{(p)}$, $1 \leq p < N_L$, $q > p$. Аналогично случаю, представленно-

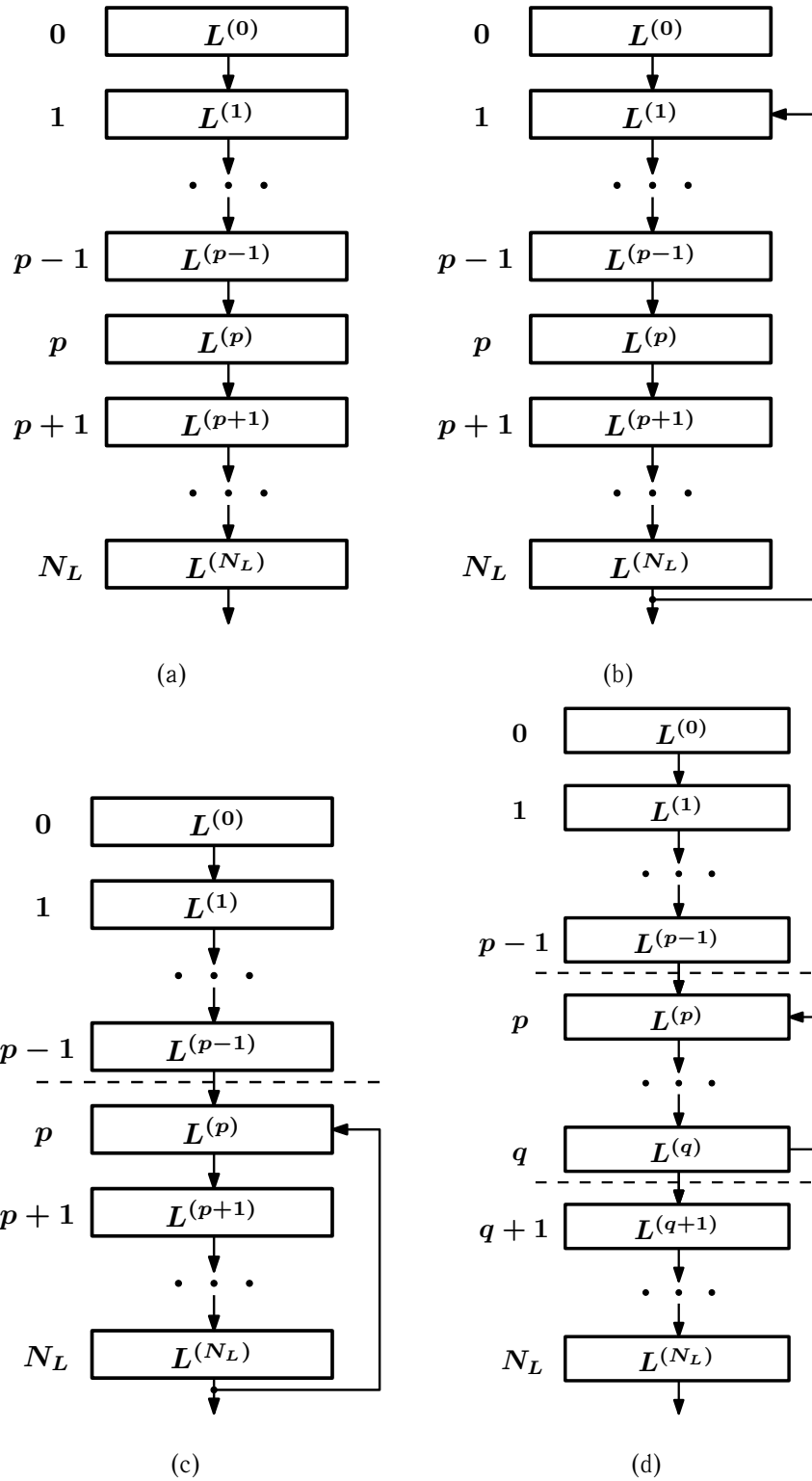


Рис. 2.6. Варианты структурной организации слоистой нейронной сети с последовательной нумерацией слоев: (а) — сеть прямого распространения; (б) — сеть с обратной связью от выходного слоя $L^{(N_L)}$ к первому обрабатывающему слою $L^{(1)}$; (с) — сеть с обратной связью от выходного слоя $L^{(N_L)}$ к произвольному слою $L^{(p)}$, $1 < p < N_L$; (д) — сеть с обратной связью от слоя $L^{(q)}$, $1 < q < N_L$ к слою $L^{(p)}$, $1 < p < N_L$

му на рис. 2.6b, данный вариант можно трактовать как последовательное соединение нейронной сети прямого распространения (слои $L^{(1)}, \dots, L^{(p-1)}$), сети с обратной связью (слои $L^{(p)}, \dots, L^{(q)}$) и еще одной сети прямого распространения (слои $L^{(q+1)}, \dots, L^{(N_L)}$). Работу такой сети можно, например, интерпретировать следующим образом: рекуррентная подсеть (слои $L^{(p)}, \dots, L^{(q)}$) — это основная часть НС в целом, а две подсети прямого распространения (слои $L^{(1)}, \dots, L^{(p-1)}$ и $L^{(q+1)}, \dots, L^{(N_L)}$) осуществляют предобработку данных, поступающих в основную подсеть (слои $L^{(1)}, \dots, L^{(p-1)}$), и постобработку данных, выдаваемых основной рекуррентной подсетью (слои $L^{(q+1)}, \dots, L^{(N_L)}$).

4. Во всех вариантах НС, показанных на рис. 2.6, сохраняется неизменным строгий порядок следования слоев, которые активируются один за другим в порядке, задаваемом прямыми и обратными связями имеющимися в рассматриваемой НС. Для сети прямого распространения это означает, что любой нейрон из слоя $L^{(p)}$ получает свои входы только от нейронов из слоя $L^{(p-1)}$ и передает свои выходы слою $L^{(p+1)}$, т. е.

$$L^{(p-1)} \rightarrow L^{(p)} \rightarrow L^{(p+1)}, \quad p \in \{0, 1, \dots, N_L\}. \quad (2.6)$$

При этом одновременно (параллельно) два или большее число слоев исполняться не могут, даже если имеется такая техническая возможность (сеть исполняется на некоторой параллельной вычислительной системе) в силу последовательной логики срабатывания слоев НС, отмеченной выше.

Использование обратной связи вводит цикличность в порядок срабатывания слоев, всех, начиная с $L^{(1)}$ и по $L^{(N_L)}$ включительно, или их части для некоторого диапазона номеров $p_1 \leq p \leq p_2$, в зависимости от того, какие слои НС охвачены обратной связью, однако строгая последовательность сохраняется — если какой-то из слоев НС начал свою работу, то пока эта работа не будет завершена, никакой другой слой запускаться на обработку не будет.

Отказ от такого рода строгой последовательности срабатывания слоев НС приводит к появлению в сети параллелизма на уровне ее слоев. В предельном случае допускается для любого нейрона из слоя $L^{(p)}$ и любого нейрона из слоя $L^{(q)}$ установление связи любого типа — прямой, обратной (для этих случаев $p \neq q$) или латеральной (в этом случае $p = q$). Здесь пока по-прежнему считается, что используется слоистая организация типа «стопка слоев».

5. Варианты структурной организации НС, показанные на рис. 2.6, используют одну и ту же схему «стопка слоев» для упорядочения слоев сети. Здесь на каждом временном отрезке работают нейроны лишь одного слоя. Остальные либо уже отработали, либо ждут своей

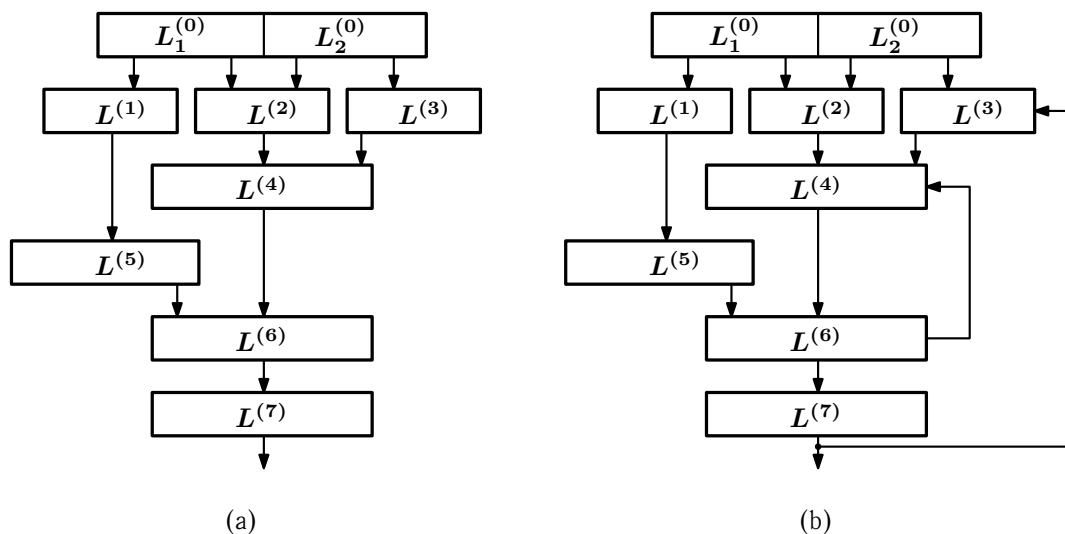


Рис. 2.7. Пример структурной организации слоистой нейронной сети с параллелизмом на уровне слоев: (a) — НС прямого распространения; (b) — НС с обратными связями

очереди. Это касается как сетей прямого распространения, так и сетей с обратными связями.

Следующий вариант позволяет уйти от схемы «стопка слоев» и перейти к более сложным структурам. В качестве примера, иллюстрирующего структуры подобного рода, на рис. 2.7 показаны два варианта структур НС с параллелизмом в них на уровне слоев²⁸.

Очевидно, что как для схем, показанных на рис. 2.6, так и для схем, изображенных на рис. 2.7, чтобы нейрон из некоторого p -го слоя сработал, он должен получить значения всех своих входов, до этого момента он «ждет». Возможности распараллеливания срабатывания нейронов определяются тем же самым условием: все нейроны, у которых к данному моменту времени сформирован полный комплект входов, могут обрабатываться («срабатывать») независимо друг от друга в произвольном порядке или параллельно, если есть такая техническая возможность.

Логика срабатывания нейронов в НС (т.е. последовательность и условия срабатывания), организованной по схеме «стопка слоев», обеспечивает отсутствие конфликтов между ними. Если же в НС вводится параллелизм на уровне слоев, для обеспечения такой бесконфликтности в логику срабатывания нейронов сети уже требуется вводить дополнительные правила синхронизации.

А именно, нейрон может сработать, как только он будет готов к срабатыванию, а он будет готов, как только для него будут получены значения всех входов. Как только нейрон будет

²⁸При отказе от схемы «стопка слоев» часть слоев в НС могут работать параллельно, т.е. одновременно друг с другом, если имеется такая техническая возможность.

готов к срабатыванию, его надо запускать сразу же, как только для этого появляется возможность, так как его выходы требуются для обеспечения готовности к срабатыванию других нейронов, следующих за данным в тех причинно-следственных цепочках, в которых задействован данный нейрон.

Для данной конкретной НС можно указать (сформировать) совокупность причинно-следственных цепочек, которые обеспечивают возможность контролировать условия срабатывания различных нейронов с целью предотвращения конфликтов между ними.

Для слоистых сетей прямого распространения со структурами, показанными на рис. 2.6, причинно-следственные цепочки будут иметь строго линейную структуру, без ветвлений и циклов. Введение структур с параллелизмом на уровне слоев сети типа показанных на рис. 2.7, в которых могут присутствовать как «перескоки» вперед, так и обратные связи, вводит в причинно-следственные цепочки нелинейность (древовидные структуры и циклы).

Причинно-следственная цепочка должна показывать, от каких нейронов получает сигналы данный анализируемый нейрон, т. е. какие нейроны-предшественники должны сработать, чтобы данный нейрон получил полный комплект значений входов (необходимое условие готовности к срабатыванию данного нейрона) — это причинная часть цепочки, кроме того, указывается, каким нейронам будет передан выход данного «текущего нейрона» — это следственная часть цепочки.

6. Во всех вариантах структурной организации НС содержались только прямые и обратные связи, т. е. такие связи между парами нейронов, в которых нейроны, входящие в эту пару, принадлежат к разным слоям.

Третий вид связей, возможных между нейронами в НС, это латеральные связи, в которых оба нейрона, между которыми установлена связь, принадлежат одному и тому же слою. Одним из примеров НС с латеральными связями является рекуррентный мультиперсептрон RMLP (Recurrent MultiLayer Perceptron) [83, 84].

2.2.2 Нейроны как элементы, из которых образована НС

1. Совокупность (множество) L_{Σ} всех элементов (нейронов), входящих в НС, разделяется на подмножества (слои):

$$L^{(0)}, L^{(1)}, \dots, L^{(p)}, \dots, L^{(N_L)}, \quad (2.7)$$

или, в более краткой записи,

$$\begin{aligned} L^{(p)}, \quad p = 0, 1, \dots, N_L, \\ L^{(p)}, L^{(q)}, L^{(r)}; \quad p, q, r \in \{0, 1, \dots, N_L\}, \end{aligned} \quad (2.8)$$

где N_L — число слоев, на которые делится совокупность элементов НС; p, q, r — индексы, используемые для нумерации произвольного («текущего») слоя НС.

В перечне (2.7) $L^{(0)}$ — это входной (нулевой) слой, назначение которого состоит в «раздаче» входных данных по элементам-нейронам, осуществляющим первичную обработку данных. Слои $L^{(1)}, \dots, L^{(N_L)}$ — обрабатывающие, совокупность их обеспечивает переработку входов НС в ее выходы.

2. Пусть в НС имеется N_L слоев $L^{(p)}, p = 0, 1, \dots, N_L$.

Слой $L^{(p)}$ имеет в своем составе $N_L^{(p)}$ элементов-нейронов $S_j^{(p)}$, т. е.

$$L^{(p)} = \{S_j^{(p)}\}, \quad j = 1, \dots, N_L^{(p)}. \quad (2.9)$$

Элемент $S_j^{(p)}$ имеет $N_j^{(p)}$ входов $x_{i,j}^{(p,q)}$ и $M_j^{(p)}$ выходов $x_{j,k}^{(p,q)}$.

Связи элемента $S_j^{(p)}$ с другими элементами сети можно представить как множество кортежей, показывающих куда передаются выходы элемента $S_j^{(p)}$.

3. Итак, отдельный нейрон, как модуль НС (рис. 2.8) — это отображение n -мерного входного вектора $x^{(in)} = (x_1^{(in)}, \dots, x_n^{(in)})$ в m -мерный выходной вектор $x^{(out)} = (x_1^{(out)}, \dots, x_m^{(out)})$, т. е. $x^{(out)} = \Theta(x^{(in)})$.

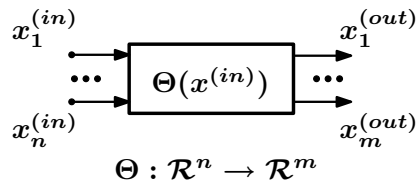


Рис. 2.8. Нейрон как модуль-преобразователь n -мерного входного вектора в m -мерный выходной вектор

Отображение Θ формируется как композиция следующих отображений-примитивов (рис. 2.9):

1) набор входных отображений $f_i(x_i^{(in)})$:

$$f_i : \mathcal{R} \rightarrow \mathcal{R}; \quad u_i = f_i(x_i^{(in)}), \quad i = 1, \dots, n; \quad (2.10)$$

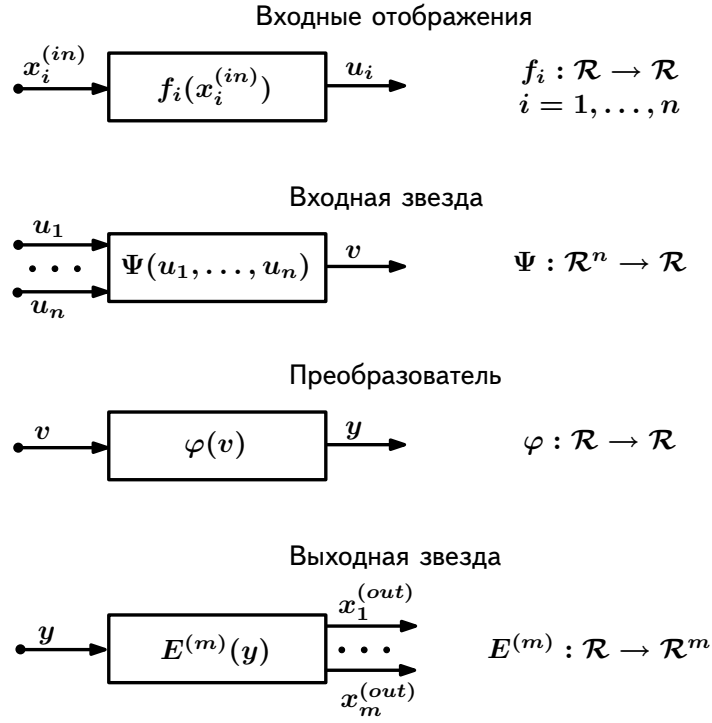


Рис. 2.9. Отображения-примитивы, из которых состоит нейрон

2) сжимающее отображение («входная звезда») $\varphi(u_1, \dots, u_n)$:

$$\varphi : \mathcal{R}^n \rightarrow \mathcal{R}; \quad v = \varphi(u_1, \dots, u_n); \quad (2.11)$$

3) преобразователь $\Psi(v)$:

$$\psi : \mathcal{R} \rightarrow \mathcal{R}; \quad y = \psi(v); \quad (2.12)$$

4) выходное отображение («выходная звезда») $E^{(m)}$:

$$E^{(m)} : \mathcal{R} \rightarrow \mathcal{R}^m; \quad E^{(m)}(y) = \{x_j^{(out)}\}, \quad j = 1, \dots, m \quad (2.13)$$

$$x_j^{(out)} = y, \quad \forall j \in \{j = 1, \dots, m\}$$

Соотношения (2.13) интерпретируются следующим образом: отображение $E^{(m)}(y)$ порождает в качестве результата m -элементное упорядоченное множество $\{x_j^{(out)}\}$, каждый из элементов которого принимает значение $x_j^{(out)} = y$.

Отображение Θ формируется как композиция отображений $\{f_i\}$, ψ , φ и $E^{(m)}$ (рис. 2.10):

$$x^{(out)} = \Theta(x^{(in)}) = E^{(m)}(\psi(\varphi(f_1^{(in)}(x_1^{(in)}), \dots, f_n^{(in)}(x_n^{(in)})))). \quad (2.14)$$

Взаимодействие отображений-примитивов, образующих нейрон, показано на рис. 2.11.

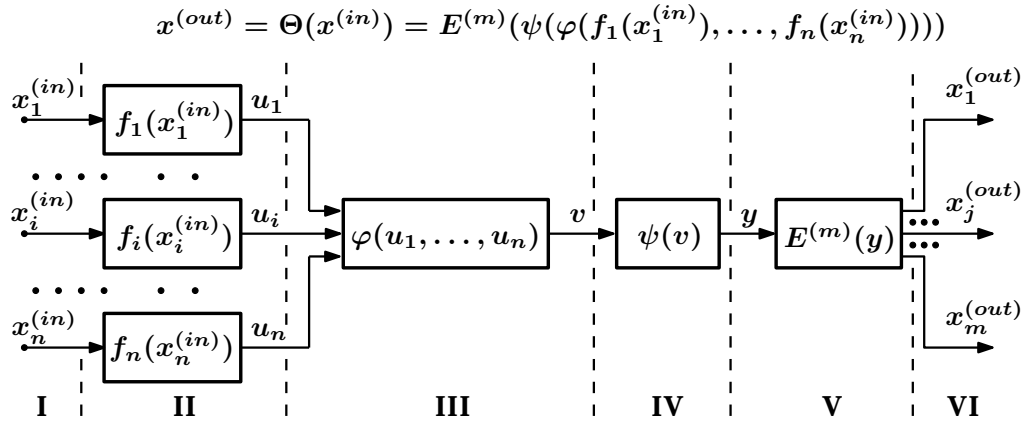


Рис. 2.10. Структура нейрона: I — входной вектор; II — входные отображения; III — агрегирующее отображение; IV — преобразователь; V — выходное отображение; VI — выходной вектор

$$x^{(out)} = \Theta(x^{(in)}) = E^{(m)}(\psi(\varphi(f_1(x_1^{(in)}), \dots, f_n(x_n^{(in)}))))$$

$$\begin{array}{l}
 \text{I} \quad x^{(in)} = (x_1^{(in)}, \dots, x_i^{(in)}, \dots, x_n^{(in)}) \\
 \Downarrow \\
 \text{II} \quad u_1 = f_1(x_1^{(in)}), \dots, u_n = f_n(x_n^{(in)}) \\
 \Downarrow \\
 \text{III} \quad v = \varphi(u_1, \dots, u_n) \\
 \Downarrow \\
 \text{IV} \quad y = \psi(v) \\
 \Downarrow \\
 \text{V} \quad x^{(out)} = E^{(m)}(y) \\
 \Downarrow \\
 \text{VI} \quad x^{(out)} = (x_1^{(out)}, \dots, x_j^{(out)}, \dots, x_m^{(out)})
 \end{array}$$

Рис. 2.11. Последовательность преобразований (отображений-примитивов), реализуемых нейроном: I — входной вектор; II — входные отображения; III — агрегирующее отображение; IV — преобразователь (активационная функция); V — выходное отображение; VI — выходной вектор

2.2.3 Структурная организация нейрона

1. Отдельный элемент-нейрон $S_j^{(p)}$ нейросетевой структуры (т. е. j -й нейрон из p -го слоя) представляет собой упорядоченную пару вида

$$S_j^{(p)} = \langle \Theta_j^{(p)}, R_j^{(p)} \rangle, \quad (2.15)$$

где $\Theta_j^{(p)}$ — преобразование входного вектора размерности $N_j^{(p)}$ в выходной вектор размерности $M_j^{(p)}$; $R_j^{(p)}$ — связи выходов элемента $S_j^{(p)}$ с другими нейронами рассматриваемой НС (с нейронами из других слоев — прямые и обратные связи; с нейронами из того же самого слоя — латеральные связи).

Преобразование $\Theta_j^{(p)}(x_{i,j}^{(r,p)})$ представляет собой композицию примитивов, из которых состоит нейрон:

$$\Theta_j^{(p)}(x_{i,j}^{(r,p)}) = \Theta(\psi(\varphi(f_{i,j}^{(r,p)}(x_{i,j}^{(r,p)}))))), \quad (2.16)$$

Связи $R_j^{(p)}$ нейрона $S_j^{(p)}$ — это множество упорядоченных пар, показывающих, куда идут выходы данного нейрона:

$$R_j^{(p)} = \{\langle q, k \rangle\}, \quad q \in \{1, \dots, N_L\}, \quad k \in \{1, \dots, N_L^q\}. \quad (2.17)$$

Входы/выходы нейронов описываются следующим образом.

В варианте с максимальной детализацией описания (расширенный уровень описания НС), который обеспечивает возможность представления любой НС-структуры, используется обозначение вида $x_{(i,l),(j,m)}^{(r,p)}$. Оно идентифицирует сигнал, передаваемый от нейрона $S_i^{(r)}$ (i -й нейрон из r -го слоя) к $S_j^{(p)}$ (j -й нейрон из p -го слоя), причем выходы i -го нейрона в r -м слое и входы j -го нейрона p -го слоя перенумерованы; согласно их нумерации, l — порядковый номер выхода элемента $S_i^{(r)}$, а m — порядковый номер входа элемента $S_j^{(p)}$. Такое подробное представление требуется в случаях, если важен порядок следования входных/выходных величин, т. е. когда набор этих величин интерпретируется как вектор. Например, такого рода представление используется в сжимающем отображении RBF-нейрона, реализующем вычисление расстояния между двумя векторами.

В варианте, когда полная детализация связей нейрона не требуется (это имеет место, когда результат «сжатия» $\varphi : \mathcal{R}^n \rightarrow \mathcal{R}$ не зависит от порядка следования входных компонент), можно использовать более простое обозначение для входных/выходных сигналов нейрона, которое имеет вид $x_{i,j}^{(r,p)}$. В таком варианте просто указывается, что связь идет от i -го нейрона r -го слоя к j -му нейрону p -го слоя, без указания порядковых номеров входных/выходных компонент.

Систему нумерации входов/выходов нейронов в НС, а также межнейронных связей иллюстрируют рис. 2.12 для базисного уровня описания НС и рис. 2.13 — для расширенного уровня.

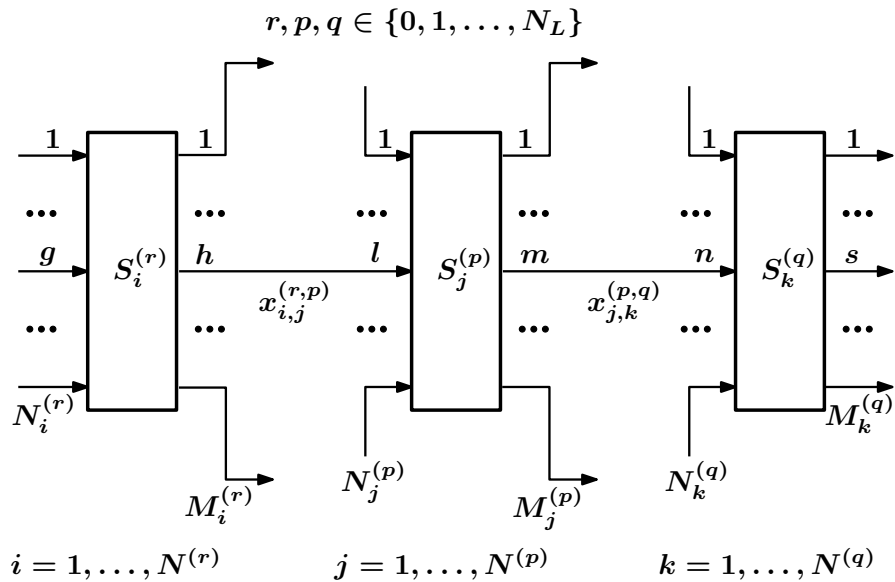


Рис. 2.12. Нумерация входов/выходов нейронов и обозначения сигналов ($x_{i,j}^{(r,p)}$ и $x_{j,k}^{(p,q)}$), передаваемых по межнейронным связям — базисный уровень описания НС: $S_i^{(r)}$, $S_j^{(p)}$ и $S_k^{(q)}$ — нейроны НС (i -й в r -м слое, j -й в p -м слое и k -й в q -м слое, соответственно); $N_i^{(r)}$, $N_j^{(p)}$, $N_k^{(q)}$ — количество входов и $M_i^{(r)}$, $M_j^{(p)}$, $M_k^{(q)}$ — количество выходов в нейронах $S_i^{(r)}$, $S_j^{(p)}$ и $S_k^{(q)}$, соответственно; $x_{i,j}^{(r,p)}$ — сигнал, передаваемый от выхода i -го нейрона из r -го слоя на вход j -го нейрона из p -го слоя; $x_{j,k}^{(p,q)}$ — сигнал, передаваемый от выхода j -го нейрона из p -го слоя на вход k -го нейрона из q -го слоя; g, h, l, m, n, s — номера входов/выходов нейронов; N_L — число слоев в НС; $N^{(r)}$, $N^{(p)}$, $N^{(q)}$ — число нейронов в слоях с номерами r, p, q , соответственно

2.2.4 Параметризация примитивов и образуемого ими нейрона

1. В НС-моделях традиционно принято в качестве параметров, варьируемых в процессе обучения сети, использовать веса межнейронных связей²⁹. В отличие от этого, в рамках развиваемого подхода вся параметризация сосредоточена внутри элементов-примитивов, композиция которых составляет нейрон, связи здесь остаются непараметризованными и лишь показывают, куда передается тот или иной сигнал. Таким образом, нейрон представляет собой композицию параметризованных отображений-примитивов. Этот вариант структурного описания НС-моделей позволяет унифицировать структурное представление всех возможных НС-моделей, что является важным фактором с точки зрения автоматизации процесса синтеза

²⁹ Данный тезис представляет собой одно из краеугольных положений коннекционизма (от англ. connection, т. е. связь), доминирующего направления в нейроинформатике, в рамках которого варьируемыми могут быть только веса связей, внутренние параметры нейронов настройке не подлежат.

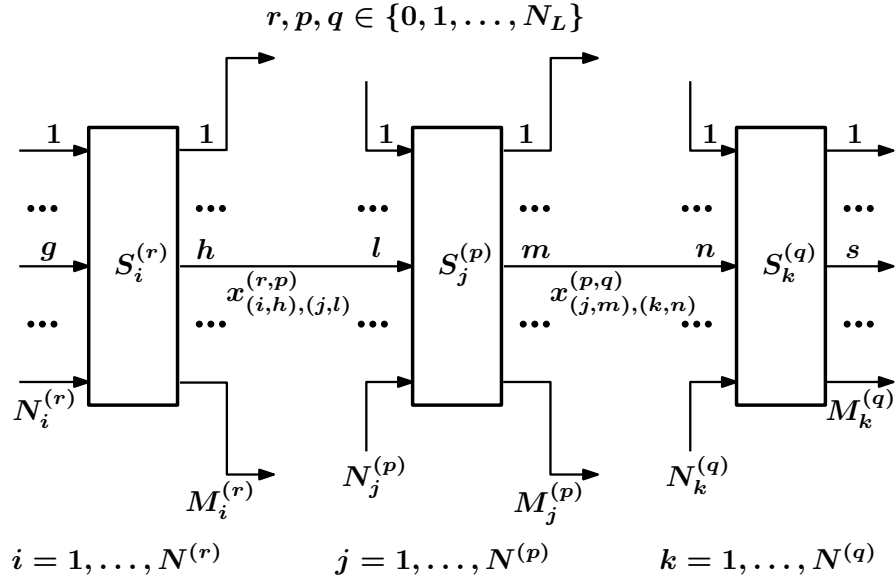


Рис. 2.13. Нумерация входов/выходов нейронов и обозначения сигналов $(x_{(i,h),(j,l)}^{(r,p)})$ и $(x_{(j,m),(k,n)}^{(p,q)})$, передаваемых по межнейронным связям — расширенный уровень описания НС: $S_i^{(r)}$, $S_j^{(p)}$ и $S_k^{(q)}$ — нейроны НС (i -й в r -м слое, j -й в p -м слое и k -й в q -м слое, соответственно); $N_i^{(r)}$, $N_j^{(p)}$, $N_k^{(q)}$ — количество входов и $M_i^{(r)}$, $M_j^{(p)}$, $M_k^{(q)}$ — количество выходов в нейронах $S_i^{(r)}$, $S_j^{(p)}$ и $S_k^{(q)}$, соответственно; $x_{(i,h),(j,l)}^{(r,p)}$ — сигнал, передаваемый от h -го выхода i -го нейрона из r -го слоя на l -й вход j -го нейрона из p -го слоя; $x_{(j,m),(k,n)}^{(p,q)}$ — сигнал, передаваемый от m -го выхода j -го нейрона из p -го слоя на n -й вход k -го нейрона из q -го слоя; g, h, l, m, n, s — номера входов/выходов нейронов; N_L — число слоев в НС; $N^{(r)}$, $N^{(p)}$, $N^{(q)}$ — число нейронов в слоях с номерами r, p, q , соответственно

НС-моделей, рассматриваемого в разд. 2.2.6.

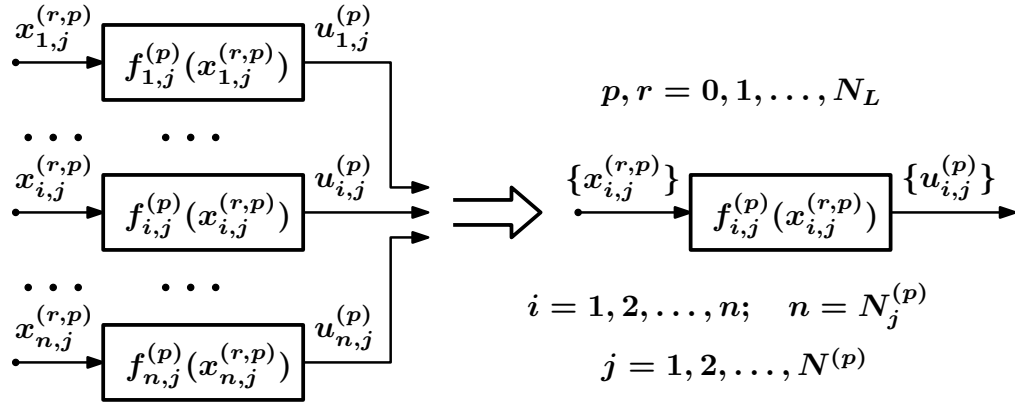
Параметризованное представление нейрона как модуля-преобразователя n -мерного входного вектора $x^{(in)} = (x_1^{(in)}, \dots, x_n^{(in)})$ в m -мерный выходной вектор $x^{(out)} = (x_1^{(out)}, \dots, x_m^{(out)})$ показано на рис. 2.17. Для отображений-примитивов, из которых состоит нейрон, такое представление дается на рис. 2.18, а для всего нейрона в целом как композиции отображений-примитивов — на рис. 2.19. Здесь приняты следующие обозначения.

Входные отображения нейрона $S_j^{(p)}$:

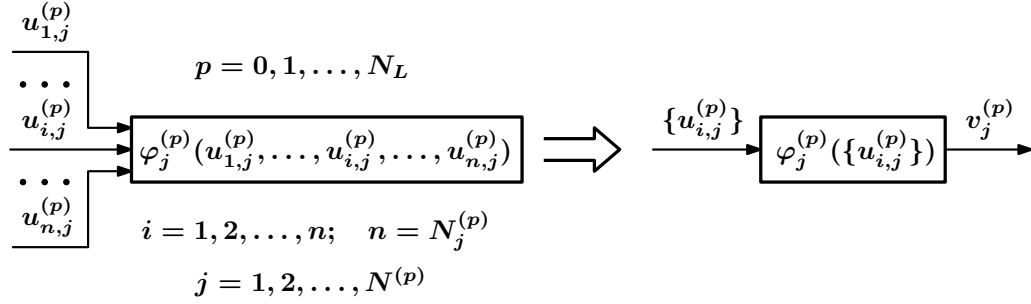
$u_{\mu,j}^{(p)} = f_{\mu,j}^{(p)}(\bar{w}_{\mu,j}^{(p)}, x_{i,j}^{(r,p)})$ — сигнал, выдаваемый входным отображением, связанным с μ -м лучом входной звезды j -го нейрона из p -го слоя НС;

$f_{\mu,j}^{(p)}$ — входное отображение, связанное с μ -м лучом входной звезды j -го нейрона из p -го слоя НС;

$\bar{w}_{\mu,j}^{(p)} = (w_{1,\mu,j}^{(p)}, w_{2,\mu,j}^{(p)}, \dots, w_{P,\mu,j}^{(p)})$, — набор параметров входного отображения $f_{\mu,j}^{(p)}$, связанно-



(a)



(b)

Рис. 2.14. Упрощенное структурное представление для элементов нейрона: (a) — совокупность входных отображений; (b) — сжимающее отображение

го с μ -м лучом входной звезды j -го нейрона из p -го слоя НС; $P = N_{\mu,j}^{(p)}$;

$\bar{w}_j^{(p)} = (\bar{w}_{1,j}^{(p)}, \bar{w}_{2,j}^{(p)}, \dots, \bar{w}_{Q,j}^{(p)})$ — набор параметров (элементарная структура, в частном случае — матрица) для всей совокупности лучей входной звезды нейрона $S_j^{(p)}$; $Q = N_j^{(p)}$;

$W^{(p)} = (\bar{w}_1^{(p)}, \bar{w}_2^{(p)}, \dots, \bar{w}_R^{(p)})$; $R = N^{(p)}$ — набор параметров (составная структура, в частном случае — блочная матрица) входных звезд всех нейронов слоя $L^{(p)}$.

Агрегирующее (сжимающее) отображение (входная звезда) нейрона $S_j^{(p)}$:

$v_j^{(p)} = \varphi_j^{(p)}(\tilde{\lambda}_j^{(p)}, \tilde{u}_j^{(p)})$ — сигнал, выдаваемый агрегирующим отображением j -го нейрона из p -го слоя НС;

$\tilde{u}_j^{(p)} = (u_{1,j}^{(p)}, u_{2,j}^{(p)}, \dots, u_{P,j}^{(p)})$ — набор сигналов, выдаваемых входными отображениями, связанными со всеми лучами входной звезды j -го нейрона из p -го слоя НС; $P = N_j^{(p)}$;

$\tilde{\lambda}_j^{(p)} = (\lambda_{1,j}^{(p)}, \lambda_{2,j}^{(p)}, \dots, \lambda_{P,j}^{(p)})$ — набор параметров агрегирующего отображения j -го нейрона из p -го слоя НС; $P = N_j^{(p)}$;

$\Lambda^{(p)} = (\tilde{\lambda}_1^{(p)}, \tilde{\lambda}_2^{(p)}, \dots, \tilde{\lambda}_Q^{(p)})$; $Q = N^{(p)}$ — набор параметров агрегирующих отображений всей совокупности нейронов слоя $L^{(p)}$ (структура, в частном случае — матрица).

$$\{x_{j,k}^{(p,q)}\} = \Theta(\{x_{i,j}^{(r,p)}\}) = E^{(m)}(\psi(\varphi(f_{1,j}^{(p)}(x_{1,j}^{(r,p)}), \dots, f_{n,j}^{(p)}(x_{n,j}^{(r,p)}))))$$

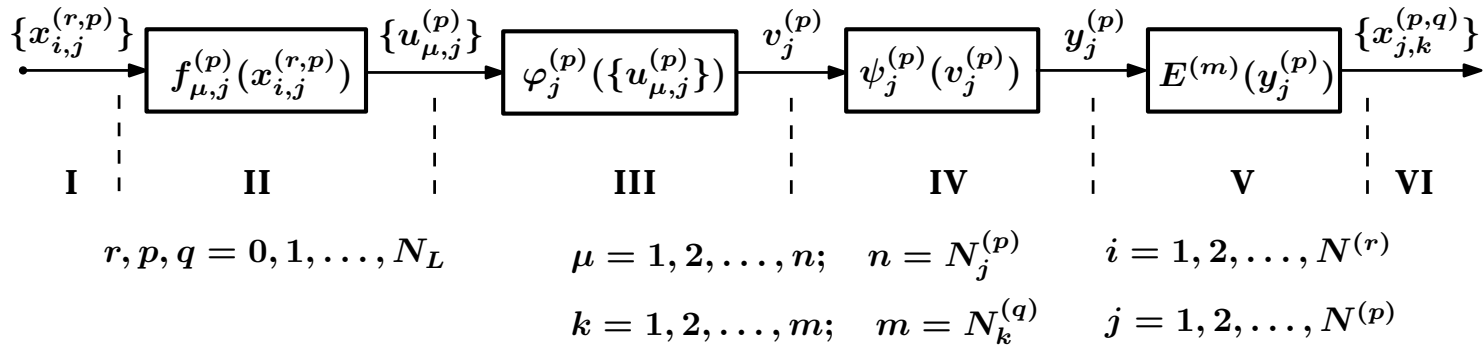


Рис. 2.15. Структура нейрона: I — входной вектор; II — входные отображения; III — сжимающее отображение; IV — преобразователь; V — выходное отображение; VI — выходной вектор

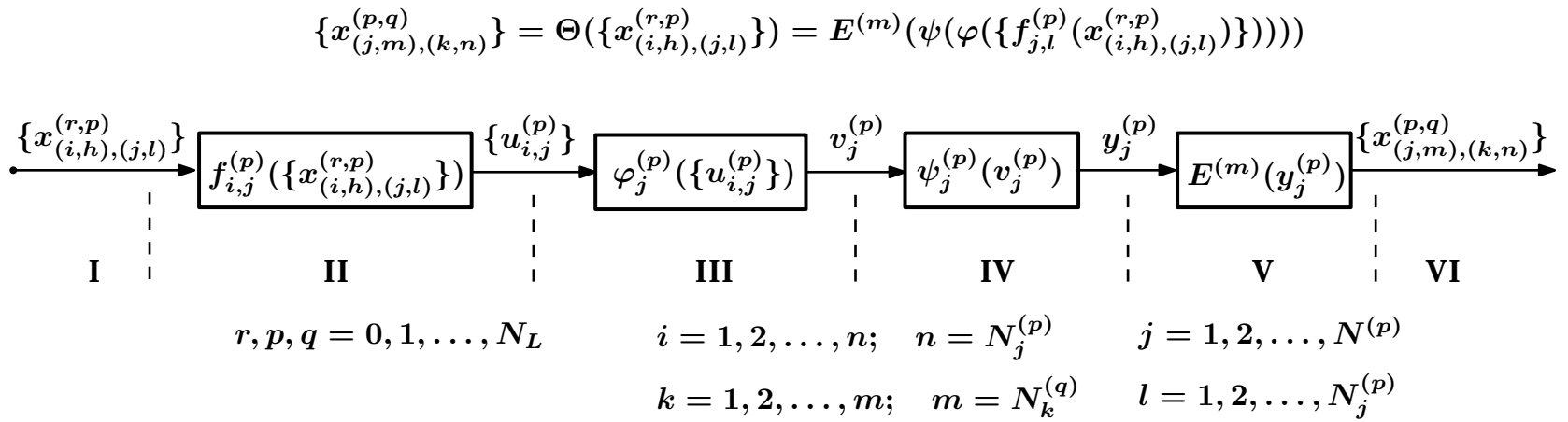


Рис. 2.16. Структура нейрона: I — входной вектор; II — входные отображения; III — сжимающее отображение; IV — преобразователь; V — выходное отображение; VI — выходной вектор

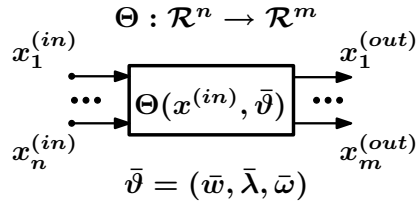


Рис. 2.17. Параметризованное представление нейрона как модуля-преобразователя n -мерного входного вектора в m -мерный выходной вектор

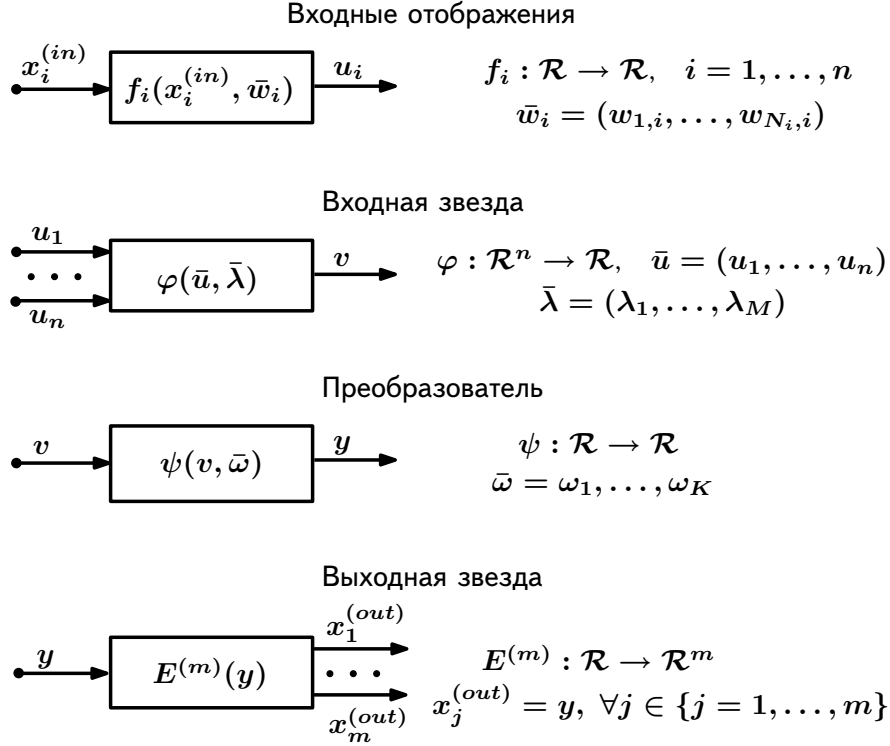


Рис. 2.18. Параметризованное представление отображений-примитивов, из которых состоит нейрон

Преобразователь (активационная функция) нейрона $S_j^{(p)}$:

$y_j^{(p)} = \psi_j^{(p)}(\tilde{\omega}_j^{(p)}, \tilde{v}_j^{(p)})$ — сигнал, выдаваемый активационной функцией j -го нейрона из p -го слоя НС;

$\tilde{\omega}_j^{(p)} = (\omega_{1,j}^{(p)}, \omega_{2,j}^{(p)}, \dots, \omega_{P,j}^{(p)})$ — набор параметров активационной функции j -го нейрона из p -го слоя НС; $P = N^{(p)}$;

$\Omega^{(p)} = (\tilde{\omega}_1^{(p)}, \tilde{\omega}_2^{(p)}, \dots, \tilde{\lambda}_P^{(p)})$ — набор параметров активационных функций всей совокупности нейронов слоя $L^{(p)}$ (структура, в частном случае — матрица); $P = N^{(p)}$.

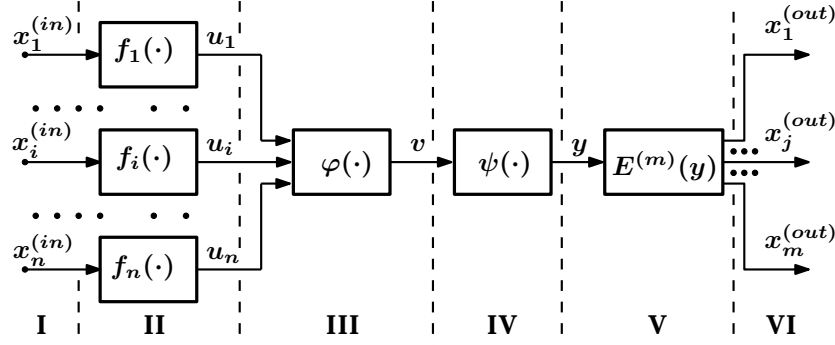
Выходное отображение (выходная звезда) нейрона $S_j^{(p)}$:

$\{x_{j,k}^{(p,q)}\} = E^{(m)}(y_j^{(p)})$ — совокупность из m выходных сигналов j -го нейрона из p -го слоя

НС, при этом все m выходов совпадают между собой по величине;

$$x^{(out)} = \Theta(x^{(in)}, \bar{\vartheta}), \quad \bar{\vartheta} = (\bar{w}, \bar{\lambda}, \bar{\omega})$$

$$x^{(out)} = \Theta(x^{(in)}, \bar{\vartheta}) = E^{(m)}(\psi(\bar{\omega}, \varphi(\bar{\lambda}, f_1(x_1^{(in)}, \bar{w}_1), \dots, f_n(x_n^{(in)}, \bar{w}_n))))$$



$$f_i(\cdot) = f_i(x_i^{(in)}, \bar{w}_i), \quad i = 1, \dots, n; \quad \bar{w}_i = (w_{1,i}, \dots, w_{N_i,i})$$

$$\varphi(\cdot) = \varphi(\bar{u}, \bar{\lambda}), \quad \bar{u} = (u_1, \dots, u_n), \quad \bar{\lambda} = (\lambda_1, \dots, \lambda_M)$$

$$\psi(\cdot) = \psi(v, \bar{\omega}); \quad \bar{\omega} = \omega_1, \dots, \omega_K$$

$$\{x_j^{(out)}\} = E^{(m)}(y); \quad j = 1, \dots, m; \quad x_j^{(out)} = y, \quad \forall j \in \{j = 1, \dots, m\}$$

Рис. 2.19. Параметризованное представление структуры нейрона: I — входной вектор; II — входные отображения; III — агрегирующее отображение; IV — преобразователь; V — выходное отображение; VI — выходной вектор

$\{x_{j,k}^{(p,q)}\}$ — сигнал, который передается от j -го нейрона из p -го слоя к k -му нейрону из q -го слоя НС.

2. После того, как указан способ формирования элементов сети (см. разд. 2.2.3), начиная с уровня элементов-примитивов (рис. 2.18), следующий шаг состоит в том, что требуется установить правила комбинирования этих элементов, т. е. объединения их в сетевые структуры.

Можно показать, что существует всего два способа комбинирования элементов сети, а именно, их последовательное соединение (рис. 2.20а) и параллельно-последовательное соединение (рис. 2.20б).

При последовательном соединении, представленном на рис. 2.20а, задействована пара отображений $f(x)$ и $g(x)$, которая порождает новое (комбинированное) отображение $z(x)$:

$$y = g(x), \quad z = f(y),$$

$$z(x) = (g \circ f)(x) = f(g(x)).$$

Таким образом, последовательное соединение — это композиция двух отображений $f : X \rightarrow \mathcal{R}$ и $g : Y \rightarrow \mathcal{R}$, $X \subset \mathcal{R}$, $Y \subset \mathcal{R}$.

При параллельно-последовательном комбинировании, которое показано на рис. 2.20б, вместо одного из исходных отображений $g(x)$ задействовано n отображений $g_1(x), \dots, g_n(x)$, а

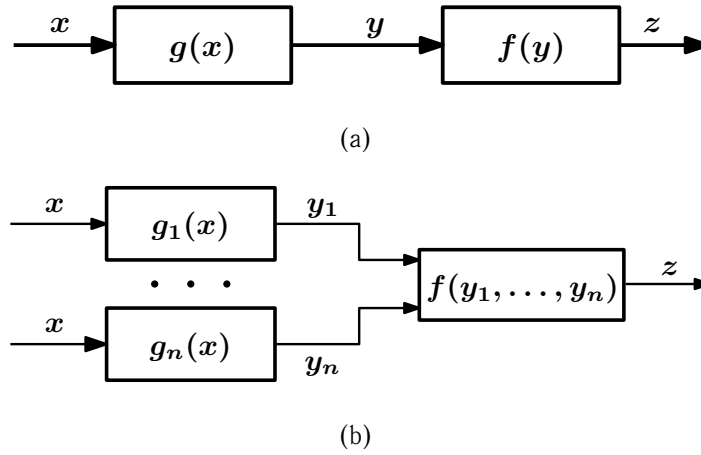


Рис. 2.20. Способы комбинирования элементов сети: (a) последовательное комбинирование; (b) параллельно-последовательное комбинирование

отображение $f(x)$ от одной независимой переменной заменяется на отображение $f(y_1, \dots, y_n)$ с n независимыми переменными:

$$y_1 = g_1(x), \dots, y_n = g_n(x), z = f(y_1, \dots, y_n);$$

$$z(x) = f(g_1(x), \dots, g_n(x)).$$

Следовательно, параллельно-последовательное соединение — это композиция набора отображений $g_1 : X \rightarrow \mathcal{R}, \dots, g_n : X \rightarrow \mathcal{R}$ и отображения $f : Y \rightarrow \mathcal{R}^n, X \subset \mathcal{R}, Y \subset \mathcal{R}^n$.

3. Таким образом, исходная база НС-моделирования, на уровне элементов-примитивов, проста и компактна: она включает в себя всего четыре класса отображений-примитивов (входные отображения, входная звезда, преобразователь и выходная звезда³⁰). Для комбинирования элементов при формировании из них сетевой модели, а также для комбинирования примитивов, образующих эти элементы, достаточно единственного правила — это композиция (суперпозиция) отображений, которая может принимать вид последовательного или параллельно-последовательного соединения отображений.

Сформулированное правило порождения решений из элементов функционального базиса применимо как к нейронным, так и к функциональным сетям.

³⁰В состав этих классов входят конкретные виды примитивов, число которых сравнительно невелико, причем, что очень важно, эти классы открыты для изменений состава примитивов в них.

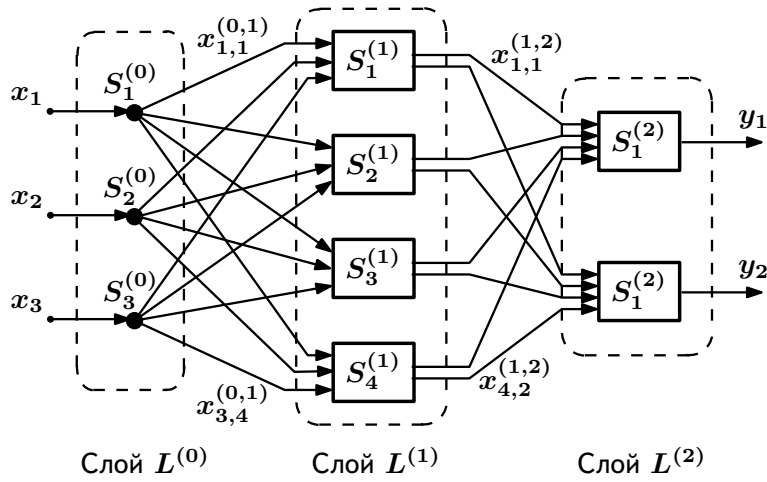


Рис. 2.21. Структура сети (пример)

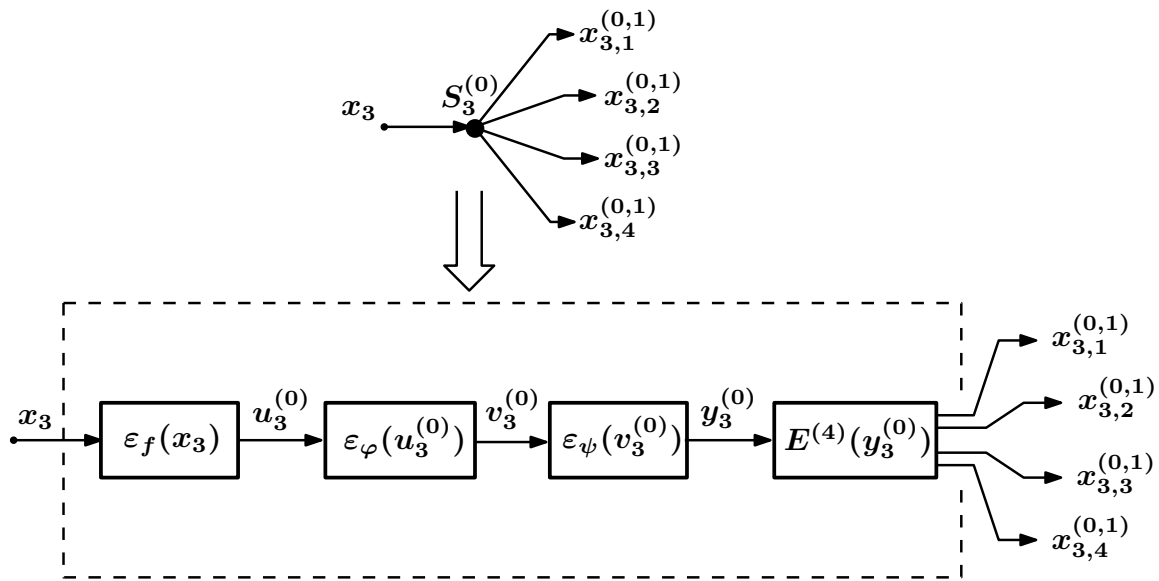


Рис. 2.22. Структура нейрона $S_3^{(0)}$

2.2.5 Пример структурного описания нейронной сети

Покажем на примере, каким образом элементы-нейроны, описанные в предыдущих разделах, объединяются в сеть. Данная сеть мультиперсептронного типа представлена на рис. 2.21. Ее структурное описание дается следующими соотношениями.

Слой $L^{(0)}$:

$$L^{(0)} = \{S_1^{(0)}, S_2^{(0)}, S_3^{(0)}\};$$

$$L^{(0)} = \{S_i^{(0)}\}, \quad i = 1, 2, 3; \quad N_L^{(0)} = 3.$$

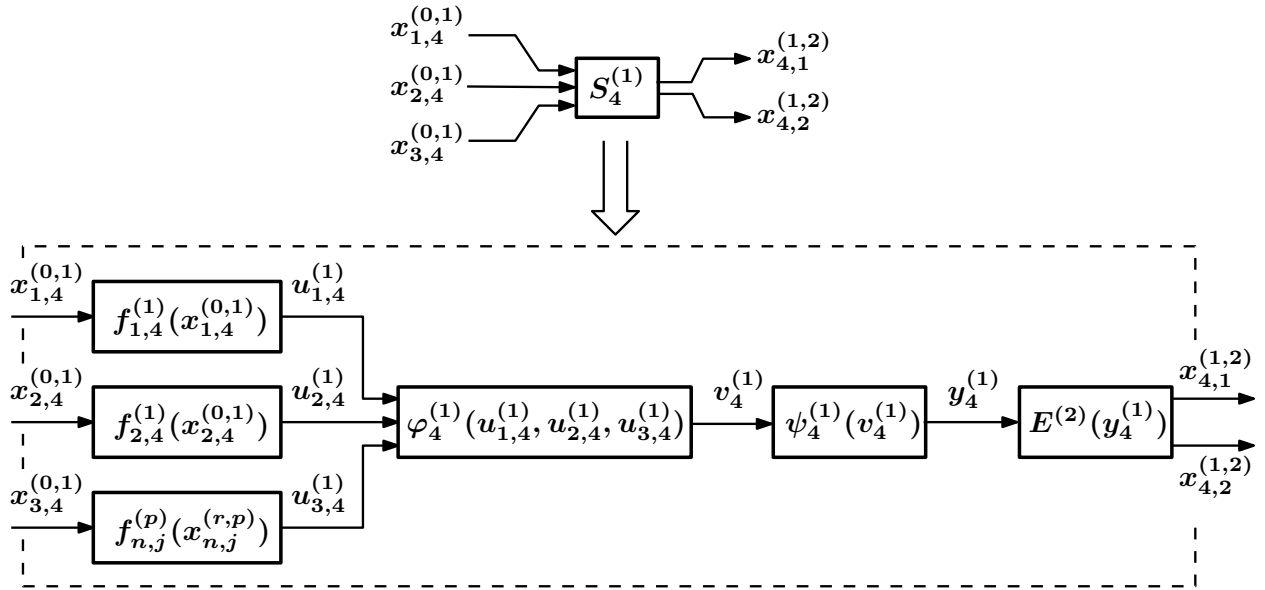


Рис. 2.23. Структура нейрона $S_4^{(1)}$

Слой $L^{(1)}$:

$$L^{(1)} = \{S_1^{(1)}, S_2^{(1)}, S_3^{(1)}, S_4^{(1)}\};$$

$$L^{(1)} = \{S_j^{(1)}\}, \quad j = 1, 2, 3, 4; \quad N_L^{(1)} = 4.$$

Слой $L^{(2)}$:

$$L^{(2)} = \{S_2^{(2)}, S_2^{(2)}\};$$

$$L^{(2)} = \{S_k^{(2)}\}, \quad k = 1, 2; \quad N_L^{(2)} = 2.$$

Сеть в целом как совокупность слоев L_Σ :

$$L_\Sigma = L^{(0)} \cup L^{(1)} \cup L^{(2)} = \{\{S_i^{(0)}\}, \{S_j^{(1)}\}, \{S_k^{(2)}\}\},$$

$$i = 1, 2, 3; \quad j = 1, 2, 3, 4; \quad k = 1, 2.$$

Далее требуется дать перечень отображений, реализуемых нейронами $S_p^{(j)}$, $p = 0, 1, \dots, N_L$, $j = 1, \dots, N_L^{(p)}$. В качестве примера приведем совокупность отображений, реализуемых элементом $S_3^{(0)}$ (рис. 2.22):

$$u_3^{(0)} = \varepsilon_f(x_3) \Rightarrow u_3^{(0)} \equiv x_3,$$

$$v_3^{(0)} = \varepsilon_\varphi(u_3^{(0)}) \Rightarrow u_3^{(0)} \equiv v_3^{(0)} \equiv x_3,$$

$$y_3^{(0)} = \varepsilon_\psi(v_3^{(0)}) \Rightarrow y_3^{(0)} \equiv u_3^{(0)} \equiv v_3^{(0)} \equiv x_3,$$

$$x_{3,1}^{(0,1)} = x_{3,2}^{(0,1)} = x_{3,3}^{(0,1)} = x_{3,4}^{(0,1)} = x_3.$$

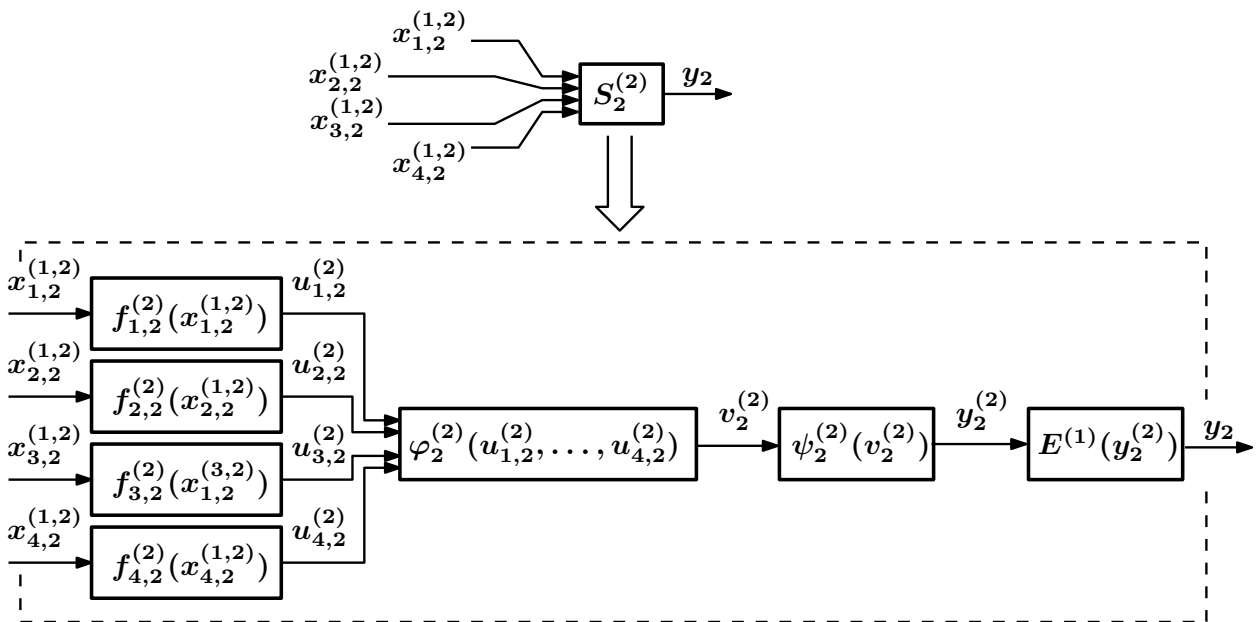


Рис. 2.24. Структура нейрона $S_2^{(2)}$

На рис. 2.23 и рис. 2.24 показаны еще два примера структурной организации нейронов сети с рис. 2.21: нейрон $S_2^{(2)}$ и $S_1^{(4)}$, соответственно.

2.2.6 Формирование подхода к автоматизации структурного синтеза НС-моделей

1. Порождающий подход к формированию НС-моделей, представленный выше, вводит в оборот громадное число потенциально возможных вариантов формируемой модели³¹. В данной ситуации выбор соответствующего варианта так, чтобы он отвечал (желательно — наилучшим образом) рассматриваемой задаче, представляет собой весьма нетривиальную проблему, требующей зачастую значительных усилий для ее решения. В то же время, решение задач математического и компьютерного моделирования в процессе создания технических систем имеет массовый характер, что не дает возможности уделять требуемое внимание каждой отдельной задаче.

Разрешить это противоречие можно было бы за счет автоматизации процесса порождения (синтеза) НС-моделей. Сформированное выше унифицированное структурное представление

³¹При этом следует подчеркнуть, что множество потенциально возможных НС-моделей является *открытым*, т.е. состав его элементов можно корректировать, в том числе и пополнять. Речь идет об отображениях-примитивах, сгруппированных в четыре открытых для корректировки класса. Меняя состав этих классов, можно практически неограниченно расширять состав потенциально порождаемых видов НС-моделей.

сетевых структур ориентировано как раз на такую автоматизацию. Идеи, на которых может быть основана автоматизация синтеза сетевых моделей (СМ), частными вариантами которых являются НС-модели и функциональные сети, излагаются ниже.

2. В проблеме автоматизации синтеза сетевых моделей могут быть выделены следующие составные части:

1. Комплекс требований. Следует научиться формулировать комплекс требований (целей, ограничений, условий), адекватно характеризующих решаемую прикладную проблему.
2. Комплекс средств. Необходимо создать комплекс средств, порождающих потенциально богатый класс решений.
3. Механизм синтеза. Необходимо создать механизм синтеза (поиска) решений, наилучшим образом удовлетворяющих заданному комплексу требований.

В этом перечне центральным элементом, требуемым для автоматизации синтеза сетевых моделей является комплекс средств, порождающих потенциально богатый класс решений, т. е. искомым моделям. Как уже отмечалось выше, чтобы породить какие-либо решения, требуется иметь в своем распоряжении:

- базис, т. е. набор элементов, из которых формируются решения;
- правила, используемые для формирования решений на основе данного базиса.

При этом данный комплекс средств должен удовлетворять двум взаимоисключающим требованиям: базис должен быть как можно более простым и компактным; порождаемое множество решений должно быть как можно более богатым.

Средство, позволяющее примирить эти противоречия — иерархическая организация сетевых моделей.

3. Принятая в данной работе иерархия структурных единиц СМ показана на рис. 2.25, в его левой части.

Схеме в левой части рис. 2.25 можно поставить в соответствие некоторую «биологическую аналогию», т. е. указать для ее элементов параллели в природных системах (показаны в правой части данного рисунка). Структурные единицы СМ, показанные на рис. 2.25, можно охарактеризовать следующим образом:

- Атом СМ — это набор отображений-примитивов из классов, показанных на рис. 2.18. Примитивы представляют собой исходный набор «кирпичиков», из которых собираются структурные единицы следующих уровней для сетевых моделей. Набор примитивов

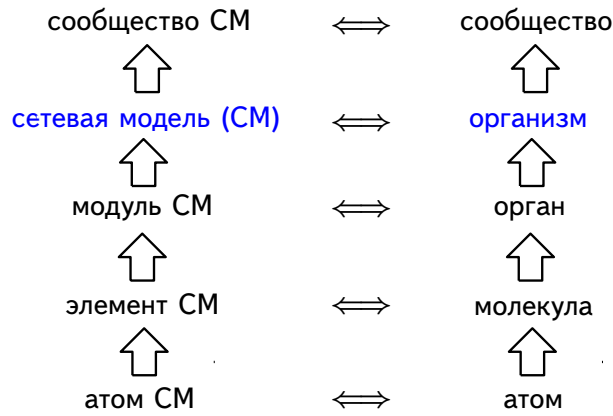


Рис. 2.25. Слева: Иерархия структурных единиц сетевых моделей. Справа: Соответствия между элементами сетевых моделей и элементами природных систем

может быть построен различными способами. Можно ставить задачу выбора наилучшего (в заданном смысле) набора примитивов.

- Элемент СМ — это сетевое объединение примитивов по определенным правилам (пример — единичный искусственный нейрон). Наиболее перспективный путь здесь состоит в формировании из примитивов очень ограниченного набора базисных элементов СМ. Разнообразные конкретные виды элементов СМ, требуемые для получения модулей СМ, формируются как частные случаи базисных элементов. Введение в рассмотрение базисных элементов СМ существенно облегчает синтез модулей СМ.
- Модуль СМ — это сетевое объединение элементов СМ в структуры, предназначенные для решения конкретных задач (пример — мультиперсептрон). Модуль СМ — это некоторая значимая часть СМ (ее «подсистема»), решающая определенную задачу (подзадачу) в рамках сетевой модели.
- Сетевая модель (СМ) — это сетевое объединение СМ-модулей в некоторую структуру, предназначенную для решения прикладной задачи или (чаще всего) класса таких задач. Пример — нейросетевая система с модульной организацией, модули в которой представляют собой отдельные нейронные сети.
- Сообщество СМ — это мультиагентное объединение сетевых моделей, предназначенную для решения сложных многоаспектных классов прикладных проблем. Примеры — программное обеспечение сложной системы из состава комплекса бортового оборудования роботизированного подвижного объекта; полный комплекс средств бортового программного обеспечения для такого объекта.

Относительно сетевых структур из данного перечня следует отметить, что, во-первых, операцией комбинирования структуры данного иерархического уровня из структур предшествующего уровня является композиция отображения; во-вторых, организация связей в сообществе СМ осуществляется через протоколы мультиагентного взаимодействия, во всех остальных видах структурных единиц СМ это будут прямые типа «вход-выход».

Таким образом, иерархическая структура сетевой модели позволяет решить проблему разрешения противоречия между требованием простоты порождающего комплекса и богатством потенциально доступного комплекса моделей. Порождающий комплекс в рамках рассматриваемого подхода представляет собой набор из четырех классов отображений-примитивов в сочетании с единственным правилом комбинирования как примитивов, так и элементов остальных иерархических уровней. Практически неограниченная гибкость получаемого комплекса моделей обеспечивается открытостью классов отображений-примитивов, доступностью их для корректировки и пополнения.

4. Сетевой подход к моделированию (как НС-подход, так и ФС-подход) обеспечивает, как отмечалось, возможность унификации математических и компьютерных моделей, требуемых в процессах создания и эксплуатации технических систем.

Характерная особенность СМ-подхода заключается в том, что акцент в нем делается на формулирование и итерационную корректировку комплекса требований к формируемой СМ. Вследствие этого, радикально меняется процесс создания систем, в первую очередь, программных. Место традиционного программирования, как процесса реализации алгоритмов при создании моделей и систем, занимают процессы «выращивания» и «воспитания» СМ.

Процесс формирования сетевой модели при таком подходе состоит из следующих этапов:

1. Сформулировать комплекс требований, условий, ограничений, описывающих проблему (синтезируемую модель или систему, цели и условия ее функционирования).
2. Взять некую исходную «протосистему» (она может изменяться далее, в процессе эволюции, как угодно, в том числе может как расти, так и сокращаться).
3. Запускается процесс эволюционного (синергетического, ...) синтеза СМ, при этом никакого программирования «внутренностей» модели (системы) нет.
4. Если требуемый результат получить не удалось, надо проанализировать комплекс требований, провести его корректировку, после чего повторить процесс синтеза.

5. Перечислим характерные особенности и свойства полученного семейства моделей. А именно, характерные особенности введенного семейства моделей состоят в следующем:

- модель ДС трактуется как разложение по некоторому функциональному базису, семейство моделей — как параметризованное разложение по функциональному базису;
- используемый функциональный базис является многоуровневым структурированным;
- используемый базис является настраиваемым;
- порождаемое данным базисом семейство моделей может относиться как к классу функциональных сетей (ФС), так и к классу нейронных сетей (НС).

Если поставить вопрос о том, что дает введенное семейство моделей, то ответы на него можно сформулировать следующим образом. Данное семейство моделей обеспечивает:

- возможность строить сетевые модели (как ФС, так и НС) произвольной сложности;
- очень высокий уровень гибкости (настраиваемости) моделей;
- возможность получения для сетевых моделей высокоэффективных алгоритмов обучения (структурной корректировки и параметрической настройки);
- приспособленность моделей к введению в них механизмов адаптации (в том числе и с использованием вставочных нейронов и вставочных подсетей);
- высокое быстродействие моделей, основанное на их хорошей распараллеливаемости;
- приспособленность моделей к реализации процессов их автоматизированного синтеза.

2.3 Формирование обучающих наборов для НС-моделирования динамических систем

2.3.1 Специфика формирования обучающих наборов для НС-моделирования динамических систем

Получение обучающего набора, обладающего требуемым уровнем информативности, является критически важным этапом решения задачи формирования НС-модели. Если какие-то особенности динамики (поведения) ДС не нашли отражения в обучающем наборе, то они, соответственно, не будут воспроизводиться моделью. В одном из фундаментальных руководств по идентификации систем это положение сформулировано как Основное правило идентификации: «Нельзя идентифицировать то, чего нет в данных» (The basic rule of system identification: “If it is not in the data, it can not be identified.”(см. [116], с. 85).

Обучающий набор данных, требуемый для формирования НС-модели ДС, должен быть информативным (репрезентативным). Будем пока считать, что обучающий набор обладает информативностью, если содержащихся в нем данных достаточно для получения НС-модели, которая с требуемым уровнем точности воспроизводит поведение ДС на всей области возможных значений для величин и их производных, характеризующих это поведение. Для обеспечения выполнения этого условия при формировании обучающего набора требуется получить данные не только об изменении величин, но также и о скорости их изменения, т.е. можно считать, что обучающий набор обладает требуемой информативностью, если полученная с его использованием НС-модель воспроизводит поведение ДС не только на всей области изменений значений величин, характеризующих поведение ДС, но и их производных (а также всех допустимых сочетаний как величин, так и их производных).

Такое интуитивное понимание информативности обучающего набора в дальнейшем будет уточнено.

2.3.2 Прямой подход к формированию обучающих наборов для НС-моделирования динамических систем

2.3.2.1 Общая характеристика прямого подхода к формированию обучающих наборов

Уточним понятие информативности обучающего набора, а также оценим его потребный объем для обеспечения необходимого уровня информативности. Вначале выполним эти действия в рамках прямого подхода к решению проблемы формирования обучающего набора, в следующем разделе будет выполнено распространение этого понятия на непрямой подход.

Рассмотрим управляемую динамическую систему вида

$$\dot{x} = F(x, u, t), \quad (2.18)$$

где $x = (x_1, x_2, \dots, x_n)$ — переменные состояния, $u = (u_1, u_2, \dots, u_m)$ — переменные управления, $t \in T = [t_0, t_f]$ — время.

Величины x_1, x_2, \dots, x_n и u_1, u_2, \dots, u_m , взятые в конкретный момент времени $t_k \in T$ характеризуют, соответственно, состояние ДС и управляющие воздействия на нее в данный

момент времени. Каждая из этих величин принимает значения из соответствующей области:

$$\begin{aligned} x_1(t_k) \in X_1 \subset \mathcal{R}, \dots, x_n(t_k) \in X_n \subset \mathcal{R}; \\ u_1(t_k) \in U_1 \subset \mathcal{R}, \dots, u_m(t_k) \in U_m \subset \mathcal{R} \end{aligned} \quad (2.19)$$

Кроме того, существуют, как правило, ограничения на значения комбинаций этих величин

$$\begin{aligned} x = \langle x_1, \dots, x_n \rangle \in R_X \subset X_1 \times \dots \times X_n, \\ u = \langle u_1, \dots, u_m \rangle \in R_U \subset U_1 \times \dots \times U_m, \end{aligned} \quad (2.20)$$

а также на сочетания этих комбинаций

$$\langle x, u \rangle \in R_{XU} \subset R_X \times R_U. \quad (2.21)$$

Пример, входящий в обучающий набор, должен показывать реакцию ДС на некоторую комбинацию $\langle x, u \rangle$. Под реакцией такого рода будем понимать состояние $x(t_{k+1})$, в которое ДС (2.18) перейдет из состояния $x(t_k)$ при значении $u(t_k)$ управляющего воздействия:

$$\langle x(t_k), u(t_k) \rangle \xrightarrow{F(x, u, t)} x(t_{k+1}). \quad (2.22)$$

Соответственно, пример p из обучающего набора P будет включать две части — входную (это пара $\langle x(t_k), u(t_k) \rangle$) и выходную (это реакция ДС $x(t_{k+1})$).

2.3.2.2 Информативность обучающего набора

Обучающий набор должен (в идеальном варианте) показывать реакции ДС на любые комбинации $\langle x, u \rangle$, удовлетворяющие условию (2.21). Тогда, согласно Основному правилу идентификации (см. с. 94), обучающий набор будет информативным, т. е. позволяющим воспроизвести в модели всю специфику поведения моделируемой ДС³².

Уточним это положение. Введем обозначение

$$p_i = \{ \langle x^{(i)}(t_k), u^{(i)}(t_k) \rangle, x^{(i)}(t_{k+1}) \}, \quad (2.23)$$

где $p_i \in P$ представляет собой i -й пример из задачника (обучающего набора) P , в этом примере

$$\begin{aligned} x^{(i)}(t_k) &= (x_1^{(i)}(t_k), \dots, x_n^{(i)}(t_k)), \\ u^{(i)}(t_k) &= (u_1^{(i)}(t_k), \dots, u_m^{(i)}(t_k)), \end{aligned} \quad (2.24)$$

³²Следует отметить, что наличие информативного обучающего набора обеспечивает *потенциальную* возможность получения модели, которая будет адекватной моделируемой ДС. Однако этой потенциальной возможностью надо еще суметь воспользоваться, что составляет отдельную нетривиальную проблему, успешное решение которой зависит от выбранного класса моделей и алгоритмов обучения.

Реакция рассматриваемой ДС $x^{(i)}(t_{k+1})$ на пример p_i :

$$x^{(i)}(t_{k+1}) = (x_1^{(i)}(t_{k+1}), \dots, x_n^{(i)}(t_{k+1})). \quad (2.25)$$

Аналогичным образом введем еще один пример $p_j \in P$:

$$p_j = \{\langle x^{(j)}(t_k), u^{(j)}(t_k) \rangle, x^{(j)}(t_{k+1})\}. \quad (2.26)$$

Исходные данные примеров p_i и p_j будем считать несовпадающими, т. е.

$$x^{(i)}(t_k) \neq x^{(j)}(t_k), \quad u^{(i)}(t_k) \neq u^{(j)}(t_k).$$

В общем случае будут не совпадать и реакции ДС на исходные данные из этих примеров:

$$x^{(i)}(t_{k+1}) \neq x^{(j)}(t_{k+1}).$$

Введем понятие ε -близости для пары примеров p_i и p_j . А именно, будем считать примеры p_i и p_j ε -близкими, если выполняется условие:

$$\|x^{(i)}(t_{k+1}) - x^{(j)}(t_{k+1})\| \leq \varepsilon, \quad (2.27)$$

где $\varepsilon > 0$ — наперед заданное действительное число.

Выделим из множества примеров $P = \{p_i\}_{i=1}^{N_p}$ подмножество, состоящее из таких примеров p_s , для которых выполнено условие ε -близости по отношению к примеру p_s , т. е.

$$\|x^{(i)}(t_{k+1}) - x^{(j)}(t_{k+1})\| \leq \varepsilon, \quad \forall s \in I_s \subset I. \quad (2.28)$$

Здесь I_s — множество индексов (номеров) тех примеров, для которых выполняется условие ε -близости по отношению к примеру p_s , при этом $I_s \subset I = \{1, \dots, N_p\}$.

Пример p_i назовем ε -представителем³³ всей совокупности примеров p_s , $\forall s \in I_s$, т. е. для любого примера p_s , $s \in I_s$ выполняется условие ε -близости. Соответственно, можно теперь заменить совокупность примеров $\{p_s\}$, $s \in I_s$ единственным ε -представителем p_i , причем погрешность, вносимая такой заменой, не будет превышать ε . Входные части совокупности примеров $\{p_s\}$, $s \in I_s$ выделяют подобласть $R_{XU}^{(s)}$, $s \in I_s$ в области R_{XU} , определяемой соотношением (2.21), при этом

$$\bigcup_{s=1}^{N_p} R_{XU}^{(s)} = R_{XU}. \quad (2.29)$$

³³Пример p_i входит в набор примеров $\{p_s\}$, $s \in I_s$.

Теперь можно поставить задачу формирования обучающего набора как такой совокупности ε -представителей, которая покрывает область R_{XU} (2.21) всех возможных значений пар $\langle x, u \rangle$.

Соотношение (2.29) представляет собой условие ε -покрытия обучающим набором P области R_{XU} . Набор P , осуществляющий ε -покрытие области R_{XU} , будем называть ε -информативным или, для краткости, просто информативным.

Если обучающий набор P обладает ε -информативностью, это означает, что для любой пары $\langle x, u \rangle \in R_{XU}$ найдется хотя бы один пример $p_i \in P$, являющийся для данной пары ε -представителем.

Применительно к ε -покрытию (2.29) области R_{XU} могут быть сформулированы следующие две задачи:

1. Задано число примеров N_p в обучающем наборе, найти их распределение в области R_{XU} , минимизирующее погрешность ε .
2. Задана допустимая величина погрешности ε , сформировать минимальную по числу N_p совокупность примеров, обеспечивающую получение такого значения ε .

2.3.2.3 Пример прямого формирования обучающего набора

Прямое формирование обучающего набора, предназначенного для синтеза НС-модели ДС, а также регулятора для этой системы, рассмотрим на примере задачи корректировки динамических свойств объекта управления [173].

1. Пусть рассматриваемый объект управления (ОУ) представляет собой динамическую систему, описываемую векторным дифференциальным уравнением вида [50, 51]:

$$\dot{x} = \varphi(x, u, t). \quad (2.30)$$

Здесь $x = (x_1 \ x_2 \ \dots \ x_n) \in \mathcal{R}^n$ — вектор переменных состояния ОУ; $u = (u_1 \ u_2 \ \dots \ u_m) \in \mathcal{R}^m$ — вектор переменных управления ОУ; $\mathcal{R}^n, \mathcal{R}^m$ — евклидовы пространства размерности n и m , соответственно; $t \in [t_0, t_f]$ — время.

В уравнении (2.30) $\varphi(\cdot)$ представляет собой нелинейную векторную функцию векторных аргументов x, u и скалярного аргумента t . Она считается заданной и принадлежащей некоторому классу функций, допускающему существование решения уравнения (2.30) при заданных $x(t_0)$ и $u(t)$ в рассматриваемой части пространства состояний ОУ.

На поведение ОУ, определяемое его динамическими свойствами, можно воздействовать, задавая то или иное корректирующее значение управляющей переменной $\Delta u(x, u^*)$. Операцию формирования требуемого значения $\Delta u(x, u^*)$ для некоторого момента времени t_{i+1} по значениям вектора состояния x и командного вектора управления u^* в момент времени t_i

$$\Delta u(t_{i+1}) = \Psi(x(t_i), u^*(t_i)) \quad (2.31)$$

будем выполнять в устройстве, которое назовем корректирующим контроллером (КК). Будем считать, что характер преобразования $\Psi(\cdot)$ в (2.31) определяется составом и значениями компонент некоторого вектора параметров $w = (w_1 w_2 \dots w_{N_w})$. Совокупность (2.30), (2.31) из ОУ и КК именуется далее управляемой системой.

Поведение системы (2.30), (2.31) с начальными условиями $x_0 = x(t_0)$ под воздействием управления $u(t)$ — это многошаговый процесс, если считать, что значения этого процесса $x(t_k)$ наблюдаются в моменты времени t_k :

$$\begin{aligned} \{x(t_k)\}, \quad t_k = t_0 + k\Delta t, \\ k = 0, 1, \dots, N_t, \quad \Delta t = \frac{t_f - t_0}{N_t}. \end{aligned} \quad (2.32)$$

В задаче (2.30), (2.31) в качестве обучающего примера, вообще говоря, можно было бы использовать пару

$$\langle (x_0^{(3)}, u^{(3)}(t)), \{x^{(3)}(t_k), k = 0, 1, \dots, N_t\} \rangle,$$

где $(x_0^{(3)}, u^{(3)}(t))$ есть начальное состояние системы (2.30) и формируемый закон управления, соответственно, а $\{x^{(3)}(t_k), k = 0, 1, \dots, N_t\}$ — многошаговый процесс (2.32), который должен реализовываться из данного начального состояния $x_0^{(3)}$ под воздействием некоторого управления $u^{(3)}(t)$ на интервале времени $[t_0, t_f]$. Сравнивая процесс $\{x^{(3)}(t_k)\}$ с процессом $\{x(t_k)\}$, получаемым для тех же самых начальных условий $x_0^{(3)}$ и управления $u^{(3)}(t)$ фактически, т. е. для некоторым образом фиксированного значения параметров w , можно было бы тем или иным способом определять расстояние между требуемым и фактически реализуемым процессами, а затем пытаться его минимизировать, варьируя значения параметров w . Такого рода «прямолинейный» подход ведет, однако, к резкому росту объема вычислений на этапе обучения НС и, в особенности, на этапе формирования соответствующего обучающего набора.

2. Существует, однако, возможность резко снизить указанные объемы вычислений, если воспользоваться тем фактом, что состояние, в которое перейдет система (2.30), (2.31) за время $\Delta t = t_{i+1} - t_i$, зависит только от ее состояния $x(t_i)$ в момент времени t_i , а также от

значения $u(t_i)$ управляющего воздействия в тот же самый момент времени. Это обстоятельство дает основание заменить многошаговый процесс $\{x^{(s)}(t_k)\}$, $k = 0, 1, \dots, N_t$, набором из N_t одношаговых процессов, каждый из которых состоит в выполнении для системы (2.30), (2.31) одного шага по времени длиной Δt из некоторой начальной точки $x(t_k)$.

3. Для того, чтобы получить совокупность начальных точек $x_t(t_0)$, $u_t(t_0)$, полностью характеризующую поведение системы (2.30), (2.31) на всей области допустимых значений $R_{XU} \subseteq X \times U$, $x \in X$, $u \in U$, построим соответствующую сетку.

Пусть переменные состояния x_i , $i = 1, \dots, n$ в уравнении (2.30) принимают значения из диапазонов, определенных для каждой из них:

$$x_i^{min} \leq x_i \leq x_i^{max}, \quad i = 1, \dots, n. \quad (2.33)$$

Аналогичные неравенства имеют место для управляющих переменных u_j , $j = 1, \dots, m$ в (2.30):

$$u_j^{min} \leq u_j \leq u_j^{max}, \quad j = 1, \dots, m. \quad (2.34)$$

Зададим на этих диапазонах сетку $\{\Delta^{(i)}, \Delta^{(j)}\}$:

$$\begin{aligned} \Delta^{(i)} : x_i^{(s_i)} &= x_i^{min} + s_i \Delta x_i, \quad i = 1, \dots, n; \quad s_i = 0, 1, \dots, N_i, \\ \Delta^{(j)} : u_j^{(p_j)} &= u_j^{min} + p_j \Delta u_j, \quad j = 1, \dots, m; \quad p_j = 0, 1, \dots, M_j. \end{aligned} \quad (2.35)$$

В выражениях (2.35) обозначено:

$$\begin{aligned} \Delta x_i &= \frac{x_i^{max} - x_i^{min}}{N_i}, \quad i = 1, \dots, n, \\ \Delta u_j &= \frac{u_j^{max} - u_j^{min}}{M_j}, \quad j = 1, \dots, m. \end{aligned}$$

Здесь же обозначено: N_i — число отрезков, на которое делится диапазон значений для переменной состояния x_i , $i = 1, \dots, n$; M_j — число отрезков, на которое делится диапазон значений для управляющей переменной u_j , $j = 1, \dots, m$.

Узлы данной сетки — это кортежи длиной $(n + m)$ вида $\langle x_i^{(s_i)}, u_j^{(p_j)} \rangle$, где компоненты $x_i^{(s_i)}$, $i = 1, \dots, n$, берутся из соответствующих $\Delta^{(i)}$, а компоненты $u_j^{(p_j)}$, $j = 1, \dots, m$ — из $\Delta^{(j)}$ в (2.35). Если область R_{XU} является подмножеством декартова произведения $X \times U$, то этот факт может быть учтен путем исключения «лишних» кортежей из сетки (2.35).

4. В [173] рассмотрен пример решения задачи НС-моделирования, в которой обучающий набор формировался согласно представленной выше методике. Исходная модель движения в

этом примере представляет собой систему уравнений следующего вида:

$$\begin{aligned} m(\dot{V}_y + V_x \omega_z) &= F_y, \\ I_z \dot{\omega}_z &= M_z, \end{aligned} \quad (2.36)$$

где F_x, F_y — проекции всех сил, действующих на самолет, на оси Ox и Oy , соответственно; M_z — проекция всех моментов, действующих на самолет, на ось Oz ; ω_z — угловая скорость тангажа; m — масса самолета; I_z — момент инерции самолета относительно оси Oz ; V_y — проекция вектора скорости самолета на ось Oy . Система уравнений (2.36) замкнута, так как угол атаки α , входящий в выражения для F_y и M_z , будет равен в рассматриваемом случае (прямолинейный горизонтальный полет) углу тангажа ϑ , который связан с V_y следующей кинематической зависимостью:

$$V_y = -V \sin \vartheta,$$

где V — скорость полета.

В (2.36) значение момента тангажа M_z является функцией от управляющей переменной — угла отклонения цельноповоротного стабилизатора, т. е. $M_z = M_z(\varphi_{ст})$.

Таким образом, система уравнений (2.36) описывает переходные процессы по угловой скорости и углу тангажа, которые возникают сразу же после нарушения балансировки, соответствующей установившемуся горизонтальному полету.

Итак, в рассматриваемом конкретном случае состав переменных состояния и управления будет следующим:

$$\mathbf{x} = [V_y \ \omega_z]^T, \quad \mathbf{u} = [\varphi_{ст}]. \quad (2.37)$$

В терминах задачи (2.36) при аппроксимации математической модели объекта управления неравенства (2.33) принимают вид:

$$\begin{aligned} V_y^{min} &\leq V_y \leq V_y^{max}, \\ \omega_z^{min} &\leq \omega_z \leq \omega_z^{max}, \end{aligned} \quad (2.38)$$

неравенство (2.34) запишется как

$$\varphi_{ст}^{min} \leq \varphi_{ст} \leq \varphi_{ст}^{max}, \quad (2.39)$$

а сетка (2.35) переписывается в форме:

$$\begin{aligned} \Delta^{(V_y)} : V_y^{(s_{V_y})} &= V_y^{min} + s_{V_y} \Delta V_y, \quad s_{V_y} = 0, 1, \dots, N_{V_y}, \\ \Delta^{(\omega_z)} : \omega_z^{(s_{\omega_z})} &= \omega_z^{min} + s_{\omega_z} \Delta \omega_z, \quad s_{\omega_z} = 0, 1, \dots, N_{\omega_z}, \\ \Delta^{(\varphi_{ст})} : \varphi_{ст}^{(p)} &= \varphi_{ст}^{min} + p_{\varphi_{ст}} \Delta \varphi_{ст}, \quad p_{\varphi_{ст}} = 0, 1, \dots, M_{\varphi_{ст}}. \end{aligned} \quad (2.40)$$

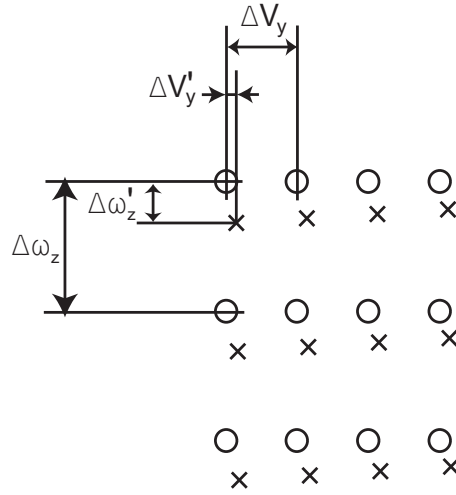


Рис. 2.26. Фрагмент сетки $\{\Delta^{(V_y)}, \Delta^{(\omega_z)}\}$ при $\varphi_{\text{ст}} = \text{const}$: \circ — стартовый узел сетки; \times — целевая точка сетки; ΔV_y , $\Delta \omega_z$ — шаг сетки по переменным состояния V_y и ω_z , соответственно; $\Delta V'_y$, $\Delta \omega'_z$ — смещение целевой точки по отношению к породившему ее узлу сетки.

5. Как уже отмечалось выше, каждый из узлов сетки (2.35) используется в качестве начального значения $x_0 = x(t_0)$, $u_0 = u(t_0)$ для системы уравнений (2.30); с этими начальными значениями выполняется один шаг интегрирования величиной Δt . Указанные начальные значения $x(t_0)$, $u(t_0)$ составляют входной вектор в обучающем примере, а полученное значение $x(t_0 + \Delta t)$ — целевой вектор, т.е. вектор-образец, показывающий алгоритму обучения НС-модели, каким должно быть значение выхода НС при данных стартовых условиях $x(t_0)$, $u(t_0)$.

Формирование обучающего набора для решения задачи нейросетевой аппроксимации динамической системы (2.30) (в частности, в ее конкретном варианте (2.36)) является нетривиальной задачей. Как показал вычислительный эксперимент [173], сходимость процесса обучения весьма чувствительна к шагу сетки Δx_i , Δu_j и шагу по времени Δt .

Поясним это положение на примере системы (2.36), когда

$$\Delta x_1 = \Delta V_y, \quad \Delta x_2 = \Delta \omega_z, \quad \Delta u_1 = \Delta \varphi_{\text{ст}}.$$

Изобразим, как это показано на рис. 2.26, часть сетки $\{\Delta^{(V_y)}, \Delta^{(\omega_z)}\}$, узлы которой используются в качестве начальных значений (входная часть обучающего примера) для получения целевой части обучающего примера. На рис. 2.26 узел сетки показан кружком, а крестиком — состояние системы (2.36), полученное интегрированием ее уравнений за шаг времени Δt с начальными условиями $(V_y^{(i)}, \omega_z^{(j)})$, для фиксированного положения стабилизатора $\varphi_{\text{ст}}^{(k)}$.

В серии вычислительных экспериментов было установлено, что при $\Delta t = \text{const}$ условиями

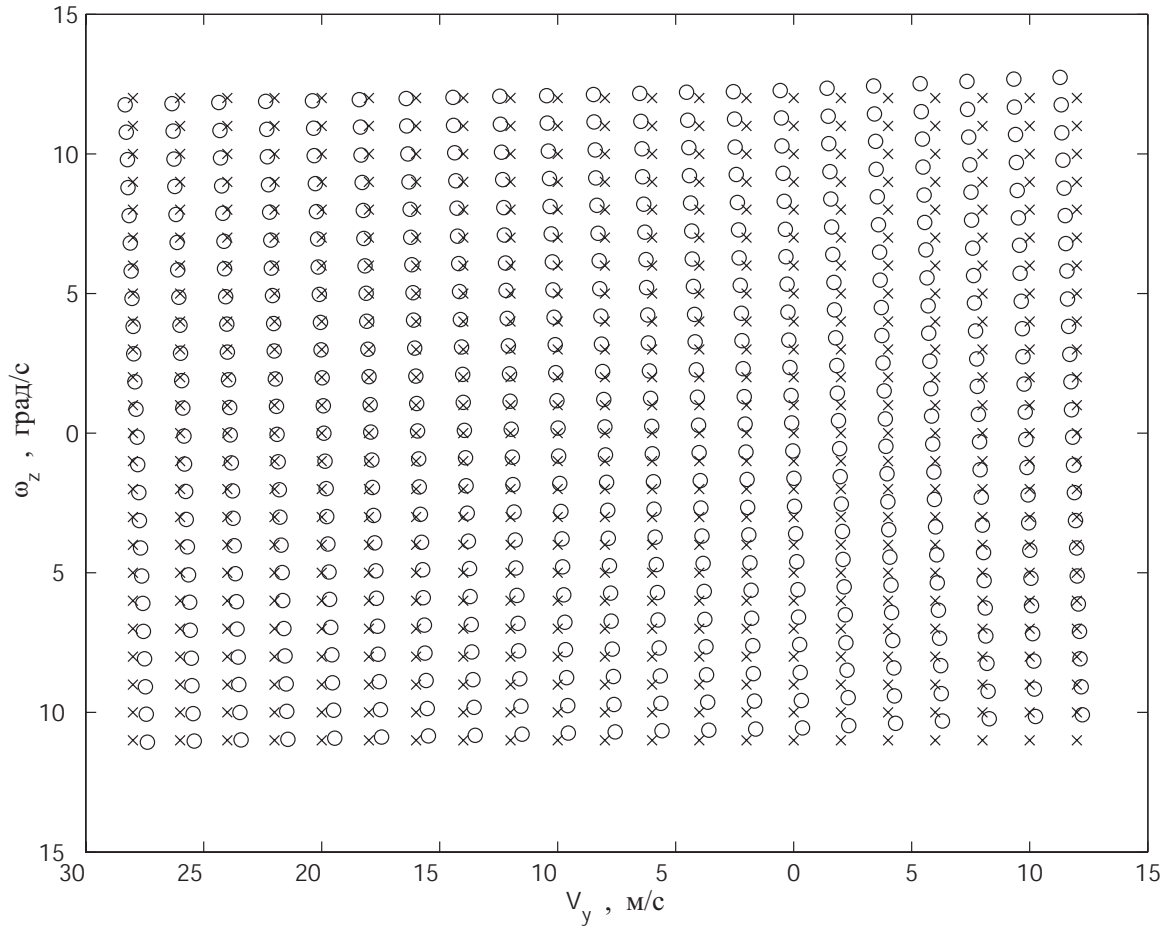


Рис. 2.27. Графическое изображение сетки $\{\Delta^{(V_y)}, \Delta^{(\omega_z)}\}$ при $\varphi_{ст} = \text{const}$, совмещенной с целевыми точками; данный лист сетки построен при $\varphi_{ст} = -8^0$.

сходимости процесса обучения НС будут следующие:

$$\begin{aligned} V_y(t_0 + \Delta t) - V_y(t_0) &< \Delta V_y, \\ \omega_z(t_0 + \Delta t) - \omega_z(t_0) &< \Delta \omega_z, \end{aligned} \quad (2.41)$$

где $\Delta V_y, \Delta \omega_z$ — шаг сетки (2.40) по соответствующим переменным состояния для заданного фиксированного значения $\varphi_{ст}$.

Сетка $\{\Delta^{(V_y)}, \Delta^{(\omega_z)}\}$, построенная для некоторой фиксированной точки $\varphi_{ст}^{(p)}$ из $\Delta^{(\varphi_{ст})}$, графически может быть изображена так, как это показано на рис. 2.27. Здесь же изображены (крестиками) и соответствующие целевые точки для каждого из узлов сетки. Совокупность («пачка») таких изображений, каждое для своего значения $\varphi_{ст}^{(p)} \in \Delta^{(\varphi_{ст})}$, дает важную информацию о структуре обучающего набора для системы (2.36), позволяя, в ряде случаев, резко сократить объем данного набора.

6. Теперь, после того как сформирована сетка (2.35) (или (2.40), для случая продольного короткопериодического движения), можно построить соответствующий обучающий набор,

после чего решать задачу обучения сети с учителем. Такая задача была решена в [173]. Полученные в этой работе результаты показывают, что на задачах небольшой размерности (определяемой размерностями векторов состояния и управления, а также величиной области допустимых значений компонент этих векторов) прямой метод формирования обучающих наборов может быть успешно использован.

2.3.2.4 Оценка объема обучающего набора при прямом подходе к его формированию

Оценим объем обучающего набора, получаемый при прямом подходе к его формированию. Рассмотрим вначале наиболее простой вариант прямого одношагового метода формирования обучающего набора, т. е. такого, в котором реакция ДС (2.25) в момент времени t_{k+1} зависит от значений переменных состояния и управления (2.24) только в момент времени t_k .

Рассмотрим данный вопрос на конкретном примере, связанном с задачей, которая решается в разд. 3 (формирование НС-модели продольного короткопериодического движения маневренного самолета). Исходная модель движения в виде системы обыкновенных дифференциальных уравнений записывается следующим образом:

$$\begin{aligned}\dot{\alpha} &= \omega_z - \frac{qS}{mV} C_{y_a}(\alpha, \omega_z, \varphi) + \frac{g}{V} \cos \vartheta, \\ \dot{\omega}_z &= \frac{qS b_A}{J_{zz}} m_z(\alpha, \omega_z, \varphi), \\ T^2 \ddot{\varphi} &= -2T\zeta \dot{\varphi} - \varphi + \varphi_{act},\end{aligned}\tag{2.42}$$

где α — угол атаки, град; ϑ — угол тангажа, град; ω_z — угловая скорость тангажа, град/с; φ — угол отклонения управляемого стабилизатора, град; C_{y_a} — коэффициент подъемной силы; m_z — коэффициент момента тангажа; m — масса самолета, кг; V — воздушная скорость, м/с; $q = \rho V^2 / 2$ — скоростной напор; ρ — плотность воздуха, кг/м³; g — ускорение силы тяжести, м/с²; S — площадь крыла, м²; b_A — средняя аэродинамическая хорда крыла, м; J_{zz} — момент инерции самолета относительно боковой оси, кг·м²; безразмерные коэффициенты C_{y_a} и m_z являются нелинейными функциями своих аргументов; T, ζ — постоянная времени и коэффициент относительного демпфирования привода, φ_{act} — командный сигнал на привод цельноповоротного управляемого стабилизатора (ограничивается $\pm 25^\circ$). В модели (2.42) величины $\alpha, \omega_z, \varphi$ и $\dot{\varphi}$ — это состояния объекта управления, величина φ_{act} — управление.

Проведем дискретизацию рассматриваемой ДС так, как это было описано в предыдущем разделе. При этом для уменьшения размерности задачи будем учитывать только переменные

α , ω_z и φ_{act} , непосредственно характеризующие поведение рассматриваемой ДС, а переменные φ и $\dot{\varphi}$ будем считать «скрытыми».

Если считать зависимости для φ и $\dot{\varphi}$ «скрытыми», то для оставшихся переменных α , ω_z и φ_{act} зададим величины $N_\alpha, N_{\omega_z}, M_{\varphi_{act}}$, представляющие собой число отсчетов по этим переменным. В предположении, что допустимыми будут все комбинации значений этих переменных, величина $N_\Sigma = N_\alpha \cdot N_{\omega_z} \cdot M_{\varphi_{act}}$, т. е. число примеров в задачнике для различных значений числа отсчетов $N_\alpha, N_{\omega_z}, M_{\varphi_{act}}$ составит (для простоты будем считать, что $N_\alpha = N_{\omega_z} = M_{\varphi_{act}} = N$):

$$\begin{aligned} N = 20 & : 20 \times 20 \times 20 = 8000, \\ N = 25 & : 25 \times 25 \times 25 = 15625, \\ N = 30 & : 30 \times 30 \times 30 = 27000. \end{aligned} \tag{2.43}$$

Если в формируемой модели ДС требуется учитывать не только переменные α , ω_z и φ_{act} , но и φ и $\dot{\varphi}$, то оценки объема получаемых обучающих наборов принимают вид:

$$\begin{aligned} N = 20 & : 20 \times 20 \times 20 \times 20 \times 20 = 3200000, \\ N = 25 & : 25 \times 25 \times 25 \times 25 \times 25 = 9765625, \\ N = 30 & : 30 \times 30 \times 30 \times 30 \times 30 = 25200000. \end{aligned} \tag{2.44}$$

Как видно из этих оценок, с точки зрения получаемого объема обучающего набора приемлемыми являются только варианты, относящиеся к ДС с векторами состояний и управлений небольшой размерности и с умеренным числом отсчетов по этим переменным (первый и второй варианты в (2.43)). Даже небольшое увеличение значений этих параметров приводит, как это видно из (2.44), к неприемлемым значениям величины обучающего набора.

В серьезных прикладных задачах, где возможности НС-моделирования особенно востребованы, результат получается еще более впечатляющим.

В частности, в полной модели углового движения самолета (соответствующая НС-модель для этого случая рассматривается в разд. 6.2) 14 переменных состояния и 3 управляющих переменных, следовательно объем обучающего набора для нее при прямом подходе к его формированию и при $N_{V_y} = N_{\omega_z} = M_{\varphi_{ст}} = 20$ будет составлять $N_\Sigma = 20^{17} = 2 \cdot 10^{18}$, что, разумеется, совершенно неприемлемо.

Таким образом, прямой подход к формированию обучающих наборов для моделирования ДС имеет очень небольшую «нишу», в которой возможно его применение — простые задачи невысокой размерности. Более сложные задачи, имеющие, соответственно, более высокую

размерность, требуют другого подхода к формированию обучающих наборов. Им является непрямой подход, основанный на формировании и использовании специальным образом организованных тестовых управляющих воздействий на ДС. Он рассматривается в следующем разделе. У непрямого подхода имеются как достоинства, так и недостатки. В ряде случаев ему нет альтернативы, в первую очередь, в ситуациях, когда получение и использование обучающих данных требуется организовать в реальном или даже в опережающем времени. Однако когда нет жестких временных ограничений на получение и использование обучающих данных, наиболее целесообразным оказывается смешанный подход, представляющий собой комбинацию прямого и непрямого подхода.

2.3.3 Непрямой подход к формированию обучающих наборов для ИС-моделирования динамических систем

1. Непрямой подход, как отмечалось в предыдущем разделе, вместо прямой дискретизации области $R_{X,U}$ допустимых значений комбинаций переменных состояния и управления, основан на данных, получаемых путем формирования и использования совокупности специальным образом организованных тестовых управляющих воздействий на ДС.

При таком подходе действительное движение ДС $(x(t), u(t))$ складывается из программного (тестовый маневр) движения $(x^*(t), u^*(t))$, порожденного управляющим сигналом $u^*(t)$, а также движения $(\tilde{x}(t), \tilde{u}(t))$, порожденного добавочным возмущающим воздействием $\tilde{u}(t)$:

$$x(t) = x^*(t) + \tilde{x}(t), \quad u(t) = u^*(t) + \tilde{u}(t). \quad (2.45)$$

В качестве примеров тестовых маневров можно назвать такие, как:

- прямолинейный горизонтальный полет с постоянной скоростью;
- полет с монотонно увеличивающимся углом атаки;
- разворот в горизонтальной плоскости;
- восходящая / нисходящая спираль.

Возможные варианты тестовых возмущающих воздействий $\tilde{u}(t)$ рассматриваются ниже.

Вид тестового маневра $(x^*(t), u^*(t))$ в (2.45) определяет получаемые диапазоны изменения значений переменных состояния и управления, вид возмущающего воздействия $\tilde{u}(t)$ — разнообразие примеров в пределах этих диапазонов.

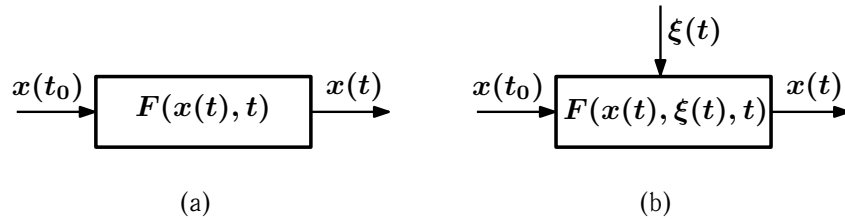


Рис. 2.28. Неуправляемая динамическая система: (а) — без внешних воздействий; (б) — с внешними воздействиями

2. Каким обучающий набор должен быть в идеальном варианте и каким он может быть получен в реальности при непрямом подходе к его формированию? Рассмотрение данного вопроса выполним в несколько этапов, начиная с наиболее простого варианта ДС с последующим его усложнением.

Рассмотрим вначале более простой случай неуправляемой ДС (рис. 2.28).

Пусть имеется некоторая ДС, т. е. система, состояние которой меняется во времени. Эта ДС является неуправляемой, на ее поведение влияют только начальные условия и, возможно, некоторые внешние воздействия (воздействия среды, в которой и во взаимодействии с которой ДС реализует свое поведение). Примером такой ДС может служить артиллерийский снаряд, на траекторию полета которого влияют начальные условия стрельбы. Воздействия среды в данном случае определяются гравитационным полем, в котором движется снаряд, а также атмосферой.

Состояние рассматриваемой ДС в конкретный момент времени $t \in T = [t_0, t_f]$ характеризуется набором величин $x = (x_1, \dots, x_n)$. Состав этого набора величин, как уже отмечалось выше, определяется тем, на какие вопросы относительно рассматриваемой ДС требуется получить ответ.

Каждая из величин x_i , $i = 1, \dots, n$ принимает, как правило, значения из некоторого допустимого диапазона $x_i \in X_i$. Кроме того, часто существуют также и ограничения на допустимые значения комбинаций этих величин, т. е., для упорядоченных n -к (кортежей) $\langle x_1, \dots, x_n \rangle$ должно выполняться условие

$$\langle x_1, \dots, x_n \rangle \in X \subset X_1 \times \dots \times X_n$$

В начальный момент времени $t \in T$ состояние ДС принимает значение $x^0 = x(t_0) = (x_1^0, \dots, x_n^0)$. При этом $x^0 = x(t_0) \in X$.

Так как величины $\{x_i\}_{i=1}^n$ описывают именно ДС, они, согласно определению ДС, изменяются во времени, т. е. ДС характеризуется совокупностью переменных величин $\{x_i(t)\}_{i=1}^n$,

$t \in T$. Эту совокупность будем именовать поведением (фазовой траекторией или траекторией в пространстве состояний) ДС.

Поведение (неуправляемой) ДС определяется, как уже отмечалось выше, ее начальным состоянием $\{x_i(t_0)\}_{i=1}^n$ и «природой ДС», т. е. тем, каким образом связаны между собой величины x_i в законе эволюции (законе функционирования) ДС $F(x, t)$. Данный закон эволюции определяет, каким будет состояние ДС в момент времени $(t + \Delta t)$, если известны эти состояния в предыдущие моменты времени.

2.3.4 Формирование набора тестовых маневров

Выше отмечалось, что выбранное программное движение (опорная траектория) как составная часть тестового маневра, определяет диапазон значений переменных состояния, в котором будут получены обучающие данные. Требуется выбрать такой набор опорных траекторий, который покрывает всю область изменения значений переменных состояния ДС. Необходимое число траекторий в таком наборе определяется из условия ε -близости фазовых траекторий ДС:

$$\|x_i(t) - x_j(t)\| \leq \varepsilon, \quad x_i(t), x_j(t) \in X, \quad t \in T. \quad (2.46)$$

Зададим семейство опорных траекторий ДС:

$$\{x_i^*(t)\}_{i=1}^{N_R}, \quad x_i^*(t) \in X, \quad t \in T. \quad (2.47)$$

Будем считать, что опорная траектория $x_i^*(t)$, $i = 1, \dots, N_R$ является ε -представителем семейства (совокупности) $X_i \subset X$ фазовых траекторий ДС в области $X_i \subset X$, если для каждой из фазовых траекторий $x(t) \in X_i$ выполняется условие

$$\|x_i^*(t) - x(t)\| \leq \varepsilon, \quad x_i^*(t) \in X_i, \quad x(t) \in X_i, \quad t \in T. \quad (2.48)$$

Семейство опорных траекторий ДС $\{x_i^*(t)\}_{i=1}^{N_R}$ должно быть таким, чтобы

$$\bigcup_{i=1}^{N_R} X_i = X_1 \cup X_2 \cup \dots \cup X_{N_R} = X, \quad (2.49)$$

где X — семейство (совокупность) всех фазовых траекторий (траекторий в пространстве состояний), потенциально реализуемых рассматриваемой ДС. Данное условие означает, что семейство опорных траекторий $\{x_i^*(t)\}_{i=1}^{N_R}$ должно представлять в совокупности все потенциально возможные варианты поведения рассматриваемой ДС. Это условие можно трактовать

как условие полноты ε -покрытия опорными траекториями области возможных вариантов поведения ДС.

Может быть поставлена задача оптимального ε -покрытия области X возможных вариантов поведения ДС, заключающаяся в минимизации числа N_R опорных траекторий в наборе $\{x_i^*(t)\}_{i=1}^{N_R}$:

$$\{x_i^*(t)\}_{i=1}^{N_R^*} = \min_{N_R} \{x_i^*(t)\}_{i=1}^{N_R}, \quad (2.50)$$

что позволяет минимизировать объем обучающего набора при сохранении его информативности.

Желательным (но трудно реализуемым) является также условие

$$\bigcap_{i=1}^{N_R} X_i = X_1 \cap X_2 \cap \dots \cap X_{N_R} = \emptyset. \quad (2.51)$$

2.3.5 Формирование тестового возбуждающего сигнала

Как уже отмечалось, вид тестового маневра в (2.45) определяет получаемые диапазоны изменения значений переменных состояния и управления, в то время как вид возмущающего воздействия обеспечивает разнообразие примеров в пределах этих диапазонов. В следующих разделах рассматриваются вопросы формирования (при заданном тестовом маневре) тестовых возбуждающих воздействий таким образом, чтобы получить информативный набор обучающих данных для ДС.

2.3.5.1 Типовые тестовые возбуждающие сигналы для идентификации систем

Устранение неопределенностей в НС-модели путем уточнения (восстановления) ряда элементов, входящих в нее (например, функций, описывающих аэродинамические характеристики самолета) — это типичная задача идентификации систем [106–113]. При решении задач идентификации для управляемых динамических систем используется ряд типовых тестовых возмущающих воздействий. Среди них наиболее употребительными являются следующие воздействия [114–118]:

- ступенчатое воздействие;
- импульсное воздействие;
- дублет (сигнал типа 1–1);
- триплет (сигнал типа 2–1–1);

- квадруплет (сигнал типа 3–2–1–1);
- случайный сигнал;
- полигармонический сигнал.

Ступенчатое воздействие (рис. 2.29а) представляет собой функцию $u(t)$, скачкообразно изменяющуюся в некоторый момент времени t_i со значения $u = 0$ на значение $u = u^*$:

$$u(t) = \begin{cases} 0, & t < t_i; \\ u^*, & t \geq t_i. \end{cases} \quad (2.52)$$

Пусть $u^* = 1$, тогда (2.52) представляет собой функцию единичного скачка $\sigma(t)$. С ее использованием можно определить еще одну разновидность тестового воздействия — прямоугольный импульс (рис. 2.29б):

$$u(t) = A(\sigma(t) - \sigma(t - T_r)), \quad (2.53)$$

где A — амплитуда импульса, $T_r = t_f - t_i$ — длительность импульса.

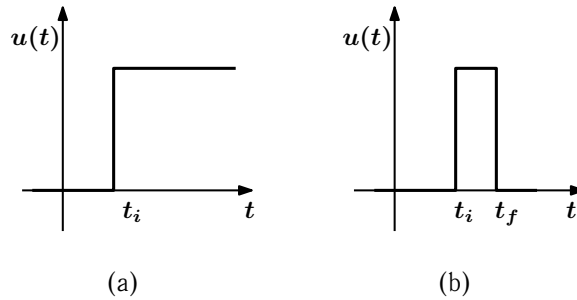


Рис. 2.29. Типовые тестовые возмущающие воздействия, используемые при изучении динамики управляемых систем: (а) — ступенчатое воздействие; (б) — импульсное воздействие

На основе прямоугольного импульсного сигнала (2.53) определяются возмущающие воздействия колебательного характера, состоящие из серии прямоугольных колебаний с определенным соотношением между их периодами. К числу наиболее употребительных воздействий такого вида относятся дублет (рис. 2.30а), триплет (рис. 2.30б) и квадруплет (рис. 2.30с).

Дублет (его обозначают также как сигнал типа 1–1) представляет собой одно полное прямоугольное колебание с периодом $T = 2T_r$, равным удвоенной длительности прямоугольного импульса.

Триплет (сигнал типа 2–1–1) представляет собой сочетание прямоугольного импульса продолжительностью $T = 2T_r$ и полного прямоугольного колебания с периодом $T = 2T_r$.

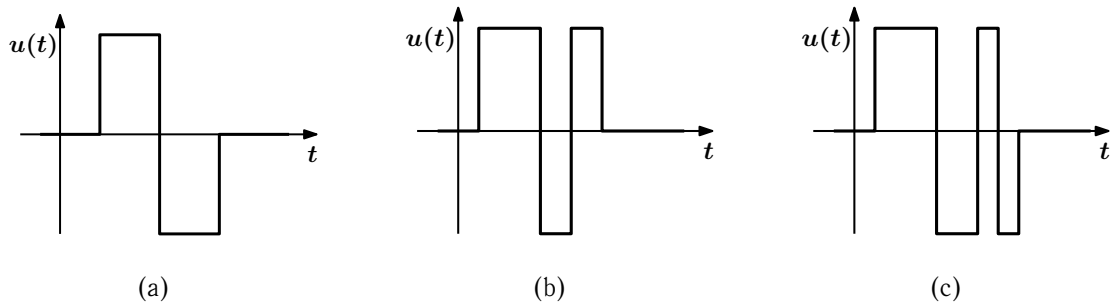


Рис. 2.30. Типовые тестовые возмущающие воздействия, используемые при изучении динамики управляемых систем: (a) — дублет (сигнал типа 1–1); (b) — триплет (сигнал типа 2–1–1); (c) — квадруплет (сигнал типа 3–2–1–1)

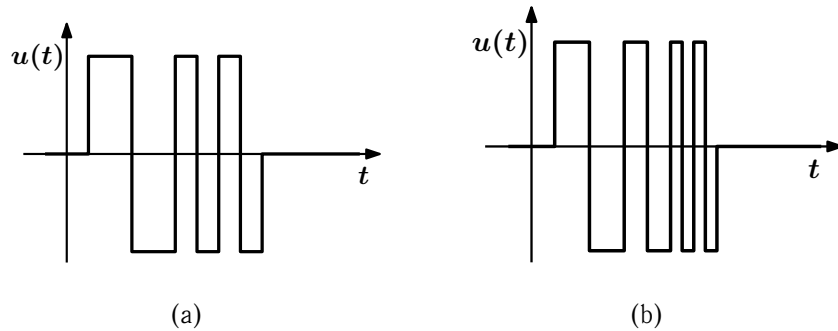


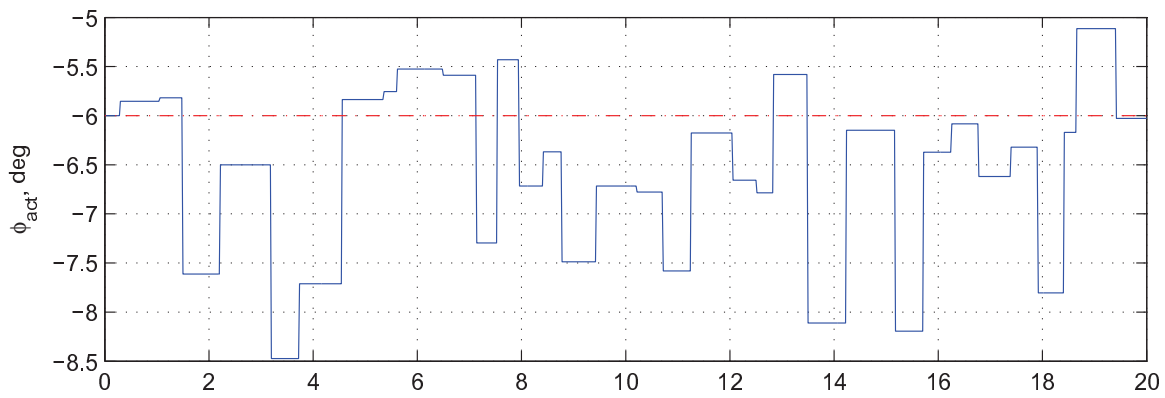
Рис. 2.31. Модифицированные варианты типовых тестовых возмущающих воздействий: (a) — триплет (сигнал типа 2–1–1); (b) — квадруплет (сигнал типа 3–2–1–1)

Квадруплет (сигнал типа 3–2–1–1) образуется из триплета добавлением в его начало прямоугольного импульса шириной $T = T_r$. Кроме того, можно использовать также варианты триплета и квадруплета, в которых каждая из составляющих частей сигнала представляет собой полнопериодное колебание (см. рис. 2.31). Будем обозначать их как сигналы типа 2–1–1 и 3–2–1–1, соответственно.

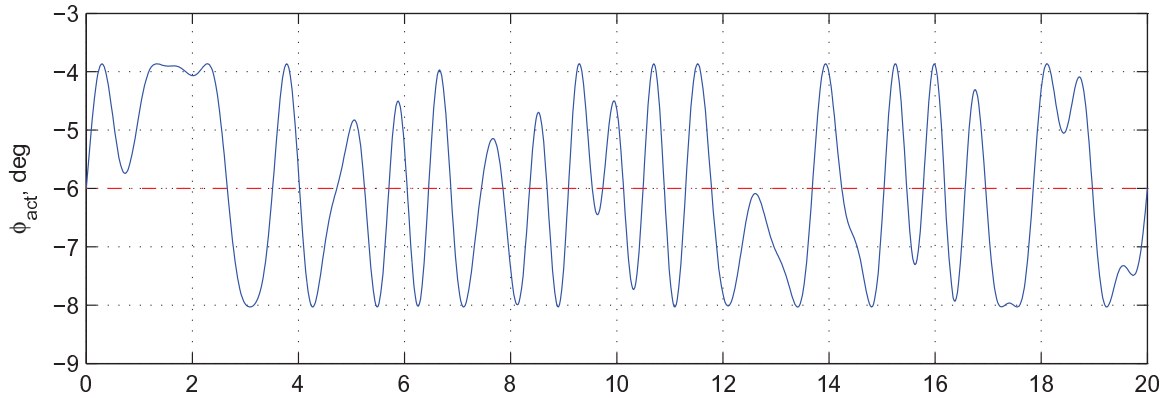
Еще один типовой возбуждающей сигнал показан на рис. 2.32а. Его значения сохраняются постоянными на всех временных интервалах $[t_i, t_{i+1})$, $i = 0, 1, \dots, n - 1$, а в моменты времени t_i могут изменяться случайным образом. Более подробно сигнал данного типа будет рассмотрен в разд. 3 на примере решения задачи НС-моделирования продольного углового движения самолета.

2.3.5.2 Полигармонические входные сигналы для идентификации систем

Для решения задач идентификации динамических систем (ДС), включая летательные аппараты, успешно применяются частотные методы. Имеющиеся результаты показывают [126–



(a)



(b)

Рис. 2.32. Тестовые возмущающие воздействия как функции времени, используемые при изучении динамики управляемых систем: (a) — случайный сигнал; (b) — полигармонический сигнал. Здесь φ_{act} — командный сигнал привода руля высоты (цельноповоротного горизонтального оперения) самолета из примера (2.42) на с. 104

129], что для заданного диапазона частот можно эффективно осуществлять оценивание параметров моделей ДС в реальном масштабе времени.

Определение состава экспериментов для моделирования ДС в частотной области представляет собой важную составную часть процесса решения задачи идентификации. Реализуемые эксперименты должны осуществляться с помощью возбуждающих сигналов, подаваемых на вход ДС, покрывающих некоторый заданный диапазон частот.

В случае, когда оценивание параметров ДС выполняется в реальном масштабе времени, желательно, чтобы возбуждающие воздействия на ДС были невелики. Если это условие будет выполнено, то реакция ДС (в частности, ЛА) на воздействие возбуждающих входов будет сопоставимым по интенсивности с реакцией, например, на атмосферную турбулентность. Тогда тестовые возбуждающие воздействия будут мало отличимыми от естественных возмущений

и не будут доставлять излишних забот экипажу ЛА.

У современного самолета, как одного из важнейших видов моделируемых ДС, значительное число органов управления (рули и т. п.). При получении данных, требуемых для частотного анализа и идентификации ДС, весьма желательной является возможность воздействовать тестовым возбуждающим сигналом на все эти органы одновременно с тем, чтобы сократить суммарное время, требуемое для сбора данных.

В работе Шрёдера [130] была показана перспективность использования для указанных целей полигармонического возбуждающего сигнала (multisine excitation), представляющего собой набор синусоид, сдвинутых по фазе друг относительно друга. Такой сигнал обеспечивает возможность получить возбуждающий сигнал с богатым частотным содержанием и с небольшим по величине значением пик-фактора (коэффициента амплитуды). Такой сигнал именуется как свип-сигнал Шрёдера (Schroeder sweep).

Пик-фактор представляет собой величину отношения максимальной амплитуды входного сигнала к энергии входного сигнала. Входы с малыми значениями пик-фактора эффективны в том смысле что они обеспечивают хорошее частотное наполнение отклика ДС без больших значений амплитуд выходного сигнала (реакции) ДС во временной области.

В работе [129] предложено развитие подхода к формированию свип-сигнала Шрёдера, позволяющее получить такой сигнал для случая нескольких органов управления, используемых одновременно, с оптимизацией значений пик-фактора для них. Это развитие ориентировано на работу в реальном масштабе времени.

Формируемые в [129] возбуждающие входные сигналы взаимно ортогональны как во временной, так и в частотной области, они интерпретируются как возмущения, добавочные к значениям соответствующих управляющих входов, требуемым для реализации заданного поведения ДС³⁴.

Для формирования тестовых возбуждающих сигналов требуется только априорная информация в виде приближенных оценок полосы частот, присущей рассматриваемой ДС, а также относительной эффективности органов управления для корректного масштабирования амплитуд входных сигналов.

³⁴В рассматриваемой задаче это тестовые маневры, обсуждавшиеся в разд. 2.3.4.

2.3.5.3 Формирование комплекса полигармонических возбуждающих сигналов

Математической моделью входного возмущающего сигнала \mathbf{u}_j , воздействующего на j -й орган управления, является гармонический многочлен

$$\mathbf{u}_j = \sum_{k \in I_k} A_k \sin\left(\frac{2\pi kt}{T} + \varphi_k\right), \quad I_k \subset K, \quad K = \{1, 2, \dots, M\}, \quad (2.54)$$

как конечная линейная комбинация из основной гармоники $A_1 \sin(\omega t + \varphi_1)$ и гармоник высших порядков $A_2 \sin(2\omega t + \varphi_2)$, $A_3 \sin(3\omega t + \varphi_2)$ и т. д.

Входное воздействие для каждого из m органов управления (например, рулевых поверностей ЛА) формируется как сумма гармонических сигналов (синусоид), каждый из которых обладает своим собственным сдвигом по фазе φ_k . Входной сигнал \mathbf{u}_j , отвечающий j -му органу управления, имеет вид:

$$\mathbf{u}_j = \sum_{k \in I_k} A_k \cos\left(\frac{2\pi kt}{T} + \varphi_k\right), \quad j = 1, \dots, m, \quad I_k \subset K, \quad K = \{1, 2, \dots, M\}, \quad (2.55)$$

где M — общее число гармонически связанных частот; T — промежуток времени, в течение которого на ДС действует тестовый возбуждающий сигнал; A_k — амплитуда k -й синусоидальной компоненты. Выражение (2.55) записано в дискретном времени для N отсчетов

$$\mathbf{t} = \{t(0), t(1), \dots, t(i), \dots, t(N-1)\},$$

где i — порядковый номер отсчета, $i = 0, 1, \dots, N-1$, а величина формируемого входного воздействия \mathbf{u}_j принимает в эти моменты времени значения

$$\mathbf{u}_j = \{u_j(0), u_j(1), \dots, u_j(i), \dots, u_j(N-1)\},$$

где $u_j(i) = u_j(t(i))$.

Каждый из m входов (возмущающих воздействий) формируется из синусоид с частотами

$$\omega_k = \frac{2\pi k}{T}, \quad k \in I_k, \quad I_k \subset K, \quad K = \{1, 2, \dots, M\},$$

где $\omega_M = 2\pi M/T$ представляет собой верхнее граничное значение полосы частот возбуждающих входных сигналов (воздействий). Интервал $[\omega_1, \omega_M]$ задает диапазон частот, в котором, как ожидается, будет лежать динамика исследуемого ЛА.

Если фазовые углы φ_k в (2.55) выбрать случайным образом в интервале $(-\pi, \pi]$, то, в общем случае, отдельные гармонические компоненты (колебания), будучи просуммированными, могут дать в отдельных точках $t(i)$ значение амплитуды суммарного сигнала $u_j(i)$,

достаточное для того, чтобы нарушились условия близости возмущенного движения к опорному. Это нежелательно, поскольку исследуемая ДС может под действием такого входного сигнала уклониться от опорного движения на недопустимую величину.

В (2.55) φ_k — это сдвиг по фазе, который надо подобрать для каждой из гармонических компонент таким образом, чтобы обеспечить небольшое значение пик-фактора³⁵ (амплитудного фактора) $\text{PF}(\mathbf{u}_j)$, определяемого соотношением:

$$\text{PF}(\mathbf{u}_j) = \frac{(u_j^{max} - u_j^{min})}{2\sqrt{(\mathbf{u}_j^T \mathbf{u}_j)/N}}, \quad (2.56)$$

или

$$\text{PF}(\mathbf{u}_j) = \frac{(u_j^{max} - u_j^{min})}{2 \text{rms}(\mathbf{u}_j)} = \frac{\|\mathbf{u}_j\|_\infty}{\|\mathbf{u}_j\|_2}, \quad (2.57)$$

где последнее равенство удовлетворяется только в случае, если \mathbf{u}_j колеблется симметрично относительно нуля. В соотношениях (2.56) и (2.57)

$$u_j^{min} = \min_i [u_j(i)], \quad u_j^{max} = \max_i [u_j(i)].$$

Для отдельной синусоидальной компоненты в (2.54) значение пик-фактора равняется $\text{PF} = \sqrt{2}$, тогда значение пик-фактора, отнесенное к такой компоненте $\text{RPF}(\mathbf{u}_j)$ (относительный пик-фактор³⁶, относительный амплитудный фактор), определяется в виде

$$\text{RPF}(\mathbf{u}_j) = \frac{(u_j^{max} - u_j^{min})}{2\sqrt{2} \text{rms}(\mathbf{u}_j)} = \frac{\text{PF}(\mathbf{u}_j)}{\sqrt{2}}. \quad (2.58)$$

Минимизация показателя (2.58) путем подбора соответствующих значений фазового сдвига φ_k для всех k позволяет предотвратить возникновения ситуации, упомянутой выше, с уклонением возмущенного движения от опорного на недопустимую величину.

2.3.5.4 Процедура формирования полигармонического входного воздействия

Процедура формирования полигармонического входного воздействия для заданной совокупности органов управления состоит из следующих шагов.

1. Задать величину интервала времени T , в течение которого на вход объекта управления будет подаваться возмущающее воздействие. Величина T определяет наименьшее значение разрешающей способности по частоте $\Delta f = 1/T$, а также предельное значение минимальной частоты $f_{min} \geq 2/T$.

³⁵PF — Peak Factor.

³⁶RPF — Relative Peak Factor.

2. Задать частотный диапазон $[f_{min}, f_{max}]$, из которого будут выбираться частоты возмущающих воздействий для рассматриваемой динамической системы. Он соответствует диапазону частот ожидаемых реакций данной системы на прилагаемые воздействия. Эти воздействия равномерно, с шагом Δf , покрывают интервал $[f_{min}, f_{max}]$. Общее число используемых частот составляет

$$M = \left\lfloor \frac{f_{max} - f_{min}}{\Delta f} \right\rfloor + 1,$$

где $\lfloor \cdot \rfloor$ — целая часть вещественного числа.

3. Разделить множество индексов $K = \{1, 2, \dots, M\}$ на примерно равные по числу элементов подмножества $I_j \subset K$, каждое из которых определяет набор частот для соответствующего j -го органа управления. Это разделение следует выполнять таким образом, чтобы частоты для различных органов управления чередовались. Например, для двух органов управления множество $K = \{1, 2, \dots, 12\}$ разделяется в соответствии с этим правилом на подмножества $I_1 = \{1, 3, \dots, 11\}$ и $I_2 = \{2, 4, \dots, 12\}$, а для трех органов управления — на подмножества $I_1 = \{1, 4, 7, 10\}$, $I_2 = \{2, 5, 8, 11\}$ и $I_3 = \{3, 6, 9, 12\}$. Такой подход обеспечивает получение малых значений пик-фактора для отдельных входных сигналов, а также позволяет обеспечить равномерное покрытие частотного диапазона $[f_{min}, f_{max}]$ для каждого из этих сигналов. При необходимости от такого рода равномерности можно уйти, если например, требуется сделать акцент на некоторых частотах или же, наоборот, при необходимости какие-то частотные компоненты исключить (в частности, из опасения вызвать нежелательную реакцию объекта управления). В работе [126] было эмпирическим путем установлено, что если наборы индексов I_j сформированы таким образом, что содержат числа больше 1, кратные 2 или 3 (например, $k = 2, 4, 6$ или $k = 5, 10, 15, 20$), то фазовый сдвиг для них может быть оптимизирован таким образом, что относительный пик-фактор для соответствующего входного воздействия будет очень близок к 1, а в ряде случаев будет даже меньше чем 1. При распределении индексов по подмножествам I_j должны выполняться условия

$$\bigcup_j I_j = K, \quad K = \{1, 2, \dots, M\}, \quad \bigcap_j I_j = \emptyset,$$

т. е. каждый индекс $k \in K$ должен быть использован и притом ровно один раз. Соблюдение этого условия обеспечивает взаимную ортогональность входных воздействий как во временной, так и в частотной области.

4. Сформировать, согласно (2.55), входное воздействие u_j для каждого из используемых органов управления, после чего вычислить начальные значения фазовых углов φ_k согласно методу Шрёдера, предполагая равномерность спектра мощности.
5. Найти значения фазовых углов φ_k для каждого из входных воздействий u_j , минимизирующие относительный пик-фактор для них.
6. Для каждого из входных воздействий u_j выполнить процедуру одномерного поиска для нахождения постоянной величины смещения по времени такой, чтобы соответствующий входной сигнал начинался при нулевом значении его амплитуды. Эта операция эквивалентна смещения графика входного сигнала вдоль временной оси таким образом, чтобы точка пересечения этого графика с осью абсцисс (т. е. с временной осью) совпала с началом координат. Фазовый сдвиг, отвечающий такому смещению, добавляется к величинам φ_k всех синусоидальных компонент (гармоник) рассматриваемого входных воздействий u_j . Следует отметить, что для получения постоянного временного сдвига всех компонент u_j их фазовые сдвиги будут различными по величине, так как каждая из компонент имеет свою частоту, отличную от частоты остальных компонент. Поскольку все компоненты сигнала u_j представляют собой гармоники одной и той же основной (базовой) частоты для периода колебаний T , если фазовые углы φ_k всех компонент изменить так, чтобы начальное значения входного сигнала было нулевым, то значение его в завершающий момент времени также будет нулевым. В таком случае энергетический спектр, ортогональность и относительный пик-фактор входных сигналов остаются неизменными.
7. Вернуться к шагу 5 и повторять соответствующие действия до тех пор, пока либо величина относительного пик-фактора достигнет предписанного значения, либо будет достигнуто предельное число итераций данного процесса. Например, целевое значение относительного пик-фактора может быть задано как 1.01, предельное число итераций 50.

Существует ряд методов, позволяющих оптимизировать частотный спектр входных (тестовых) сигналов при решении задачи оценивания параметров динамической системы. Однако все эти методы требуют значительного объема вычислений, а также определенного уровня знаний об исследуемой ДС, привязанных обычно к некоторому номинальному состоянию системы. Применительно к ситуации, рассматриваемой в данной работе, такого рода методы бесполезны в силу того, что ставится задача идентифицировать динамику системы в реаль-

ном времени для различных режимов ее функционирования, изменяющихся в широких пределах. Кроме того, решение задачи реконфигурации системы управления в случае отказов и повреждений ДС требует решения задачи идентификации при существенных и непредсказуемых изменениях в динамике системы. В таких условиях трудоемкое вычисление оптимизированного по частотному спектру входного воздействия не имеет смысла, а в ряде случаев невозможно, так как не укладывается в реальное время. Вместо этого, частотный спектр всех формируемых входных воздействий подбирается таким образом, чтобы он был равномерным в заданном диапазоне частот с тем, чтобы оказать достаточное возбуждающее воздействие на ДС.

Шаг 6 процесса, описанного выше, обеспечивает получение входного возмущающего сигнала, добавляемого к основному управляющему воздействию, выбранному, например, из соображений обеспечения балансировки самолета или для выполнения заданного маневра.

Примеры формирования полигармонического возбуждающего сигнала в прикладных задачах приводятся в разд. 6.

2.4 Алгоритмы обучения НС-моделей

Динамические НС-модели (рекуррентные нейронные сети) представляют собой достаточно сложный для обучения объект. Основные проблемы, возникающие при обучении рекуррентных сетей, а также пути их решения рассматриваются в следующих разделах.

2.4.1 Проблемы, возникающие при обучении НС-моделей динамических систем

2.4.1.1 Основные источники трудностей при обучении НС-моделей динамических систем

Возникает целый ряд проблем при обучении динамических НС-моделей в виде рекуррентных нейронных сетей. При функционировании (сдвигами) в сети происходят изменения значений

настраиваемых параметров (синаптических весов, смещений, внутренних параметров нейронов) в процессе обучения НС-модели;

- наличие долговременных зависимостей выходов сети от входов и состояний НС-модели в предыдущие моменты времени;
- очень сложный рельеф функции ошибки, изрезанный многочисленными глубокими, узкими и искривленными впадинами, а также имеющий часто встречающиеся плато.

2.4.1.2 Бифуркации динамики сети

В теории нелинейных динамических систем бифуркация — это качественная перестройка режимов функционирования ДС при малом изменении ее параметров [32, 33]. Бифуркация динамики сети — качественное изменение динамических свойств и характера поведения НС-модели при малых изменениях ее настраиваемых параметров (синаптических весов, смещений и, в ряде случаев, внутренних параметров нейронов). В терминах нейросетевого обучения это означает, что скачкообразно и значительно меняется рельеф функции ошибки.

2.4.1.3 Долговременные зависимости

При обучении динамических сетей существует так называемая проблема долговременных зависимостей, обусловленная тем, что выход НС-модели зависит от ее входов и состояний в предыдущие моменты времени, в том числе и далеко отстоящие от текущего момента времени³⁷. Градиентные методы поиска минимума функции ошибки ведут себя в этом случае неудовлетворительно. Причину такого поведения проясняет анализ асимптотического поведения ошибки обучения и ее градиента в процессе обратного распространения [85, 86], который показывает, что значения этих величин быстро (как правило, экспоненциально) убывают.

2.4.1.4 Сложный рельеф функции ошибки.

Одна из важнейших причин возникновения трудностей при обучении динамических НС-моделей — очень сложный рельеф функции ошибки, изрезанный многочисленными глубокими, узкими и искривленными впадинами³⁸.

Типичный пример, демонстрирующий сложный рельеф функции ошибки рекуррентной сети, представлен на рис. 2.33 [87]. Здесь показан профиль функции ошибки вдоль направ-

³⁷Строго говоря, выход НС-модели зависит от ее входов и состояний во все предыдущие моменты времени.

³⁸Эта причина фактически является наиболее сложной с точки зрения парирования ее влияния на процесс обучения НС-модели. Упрощение НС-модели здесь не помогает, как будет показано ниже на примере простейшей рекуррентной сети. Основной источник проблем — сама рекуррентность НС-модели, от которой нельзя избавиться, не потеряв при этом динамических свойств модели. Прочие структурные свойства сети играют существенно меньшую роль, определяющим фактором является объем обучающего набора. В данной ситуации можно рассчитывать только на такую организацию работы с обучающими данными, которая позволяла бы наращивать их используемый объем последовательно. Один из возможных вариантов такой организации предлагается ниже, в разд. 2.4.1.5.

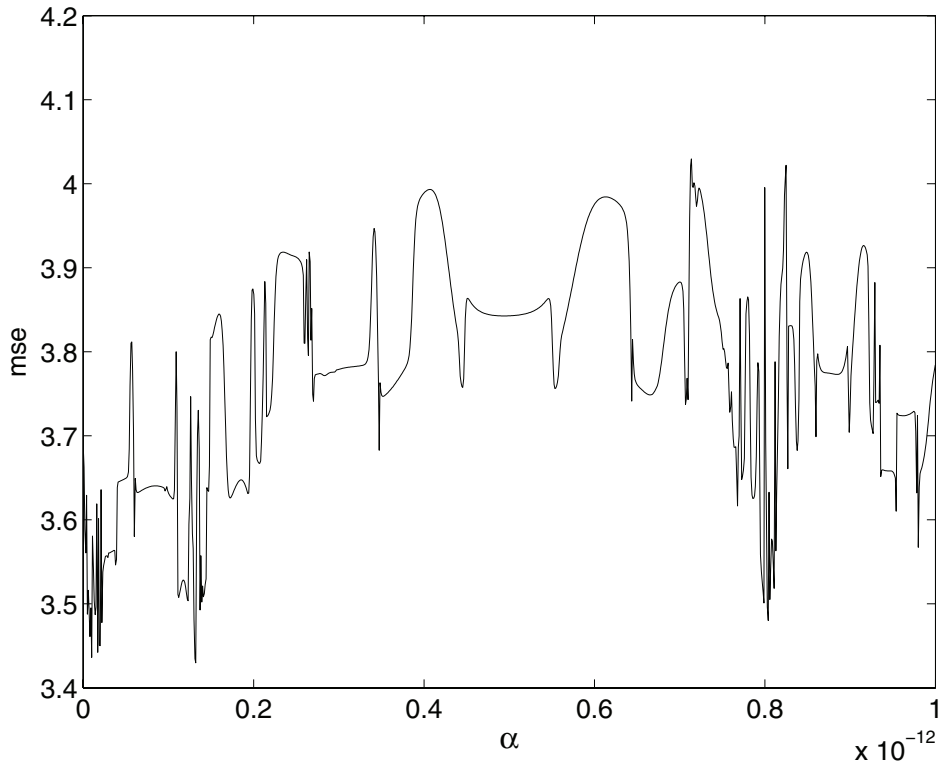


Рис. 2.33. Пример профиля функции ошибки вдоль направления поиска ее минимума

ления поиска ее минимума на одном из шагов процесса обучения сети. Обращает на себя внимание очень узкий интервал (порядка 10^{-12}) изменения значений величины α , определяющей размер шага вдоль направления поиска.

При этом существенным моментом является то, что упрощение сети в надежде получить функцию ошибки для нее с более приемлемым рельефом, проблему не решает. Корни ее в другом, а именно, в рекуррентном характере сети, последовательно (в рекуррентном цикле) обрабатывающей примеры из обучающего набора.

Это обстоятельство можно проиллюстрировать примером [88], в котором сеть максимально упрощена и тем не менее она демонстрирует отмеченное выше поведение и соответствующий усложненный рельеф функции ошибки.

А именно, рассмотрим линейную рекуррентную сеть первого порядка, состоящую из единственного линейного нейрона (рис. 2.34).

Формирование обучающего набора для этой сети осуществляется путем подачи на ее вход случайной последовательности $p(t)$ (гауссовский белый шум с нулевым средним и единичной дисперсией), порождающей последовательность выходов $a(t)$. Значения весов в этом эксперименте было принято равным $w_1 = 0.5$, $w_2 = 0.5$. Цель обучения состояла в том, чтобы

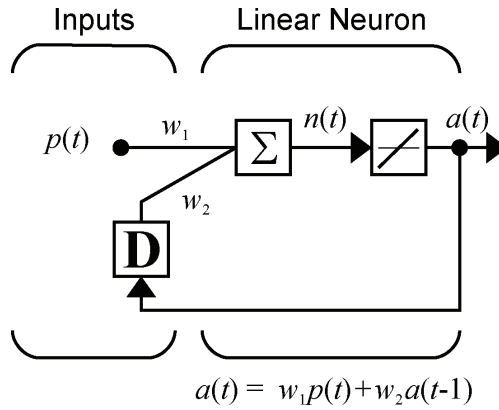


Рис. 2.34. Линейная рекуррентная сеть первого порядка

используя полученный обучающий набор $(p(t), a(t))$, обучить сеть для случайных стартовых значений весов w_1, w_2 и получить, по возможности, те значения данных весов, которые были использованы в упомянутом выше эксперименте.

Как уже отмечалось, важнейшим источником, порождающих сложный рельеф функции ошибки являются входные (обучающие) данные $p(k)$, $k = 0, 1, 2 \dots, n$. Чтобы показать это, рассмотрим как меняется выход сети по мере последовательного предъявления на ее вход обучающих примеров, а также чем определяется этот выход.

Непосредственно из структуры сети, показанной на рис. 2.34, можно найти, что ее выход на шаге с номером k задается соотношением:

$$a(k) = w_1 p(k) + w_2 a(k - 1). \quad (2.59)$$

Тогда накопление откликов сети, начиная со стартового состояния $k = 0$, до момента времени $k = n$ определяется следующим образом:

$$\begin{aligned}
 k = 0 : \quad a(0) &= a(0) \\
 k = 1 : \quad a(1) &= w_1 p(1) + w_2 a(0) \\
 k = 2 : \quad a(2) &= w_1 p(2) + w_2 a(1) = \\
 &= w_1 p(2) + w_2 (w_1 p(1) + w_2 a(0)) = \\
 &= w_1 p(2) + w_2 w_1 p(1) + w_2^2 a(0) \\
 &\dots\dots\dots \\
 k = n : \quad a(n) &= w_1 \{p(n) + w_2 p(n - 1) + \dots + w_2^{n-1} p(1)\} + w_2^n a(0)
 \end{aligned} \quad (2.60)$$

Из (2.60) видно, что на k -м шаге, т.е. при обработке k -го примера из обучающего набора, преобразование входа сети в ее выход представляет собой полином степени k от искомым

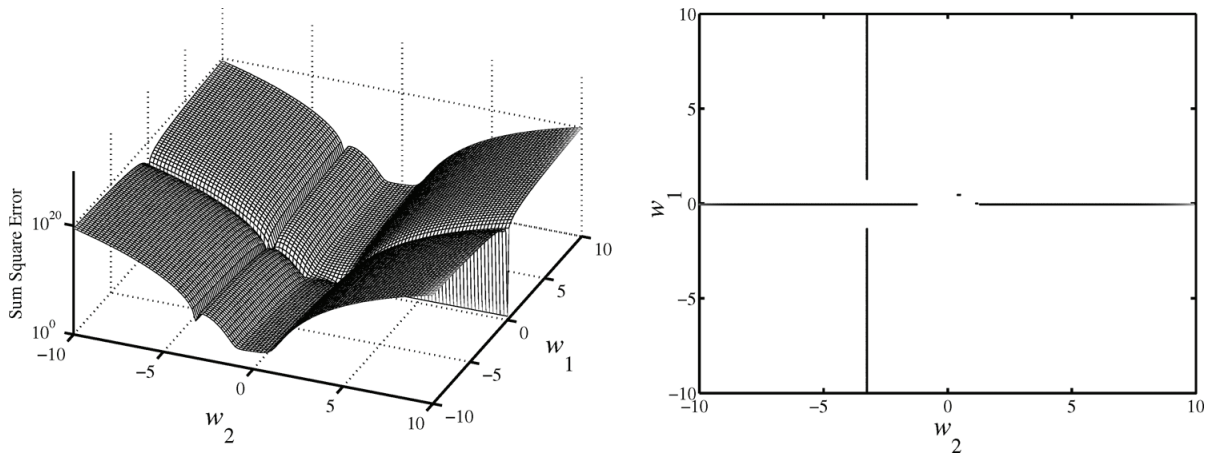


Рис. 2.35. Поверхность ошибки и расположение впадин (ущелий) на ней для линейной рекуррентной сети первого порядка, показанной на рис. 2.34

значений настраиваемых параметров сети³⁹. Это и является непосредственной причиной того рельефа поверхности ошибки, который отмечался выше.

Из рис. 2.35 видно, что для простейшей линейной однонейронной сети первого порядка поверхность ошибки выглядит не такой изрезанной, как это было показано на рис. 2.33. Тем не менее, на ней существует ряд зон, наличие которых существенно осложнит работу алгоритмов обучения сети. В действительности, уже повышение размерности входного сигнала

³⁹Здесь следует отметить, что типичное число примеров и, соответственно, максимальная степень полинома $k = n$ составляет от 1000 до 2000, в ряде случаев — еще больше.

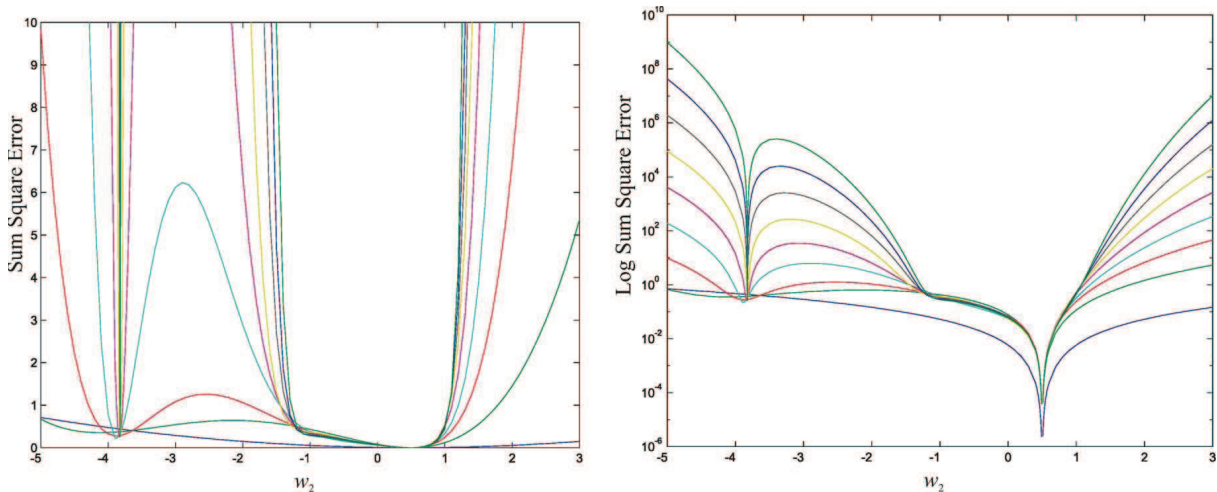


Рис. 2.36. Сечения квадратической функции ошибки при $w_1 = 0.5$ в зависимости от длины последовательности k для линейной рекуррентной сети первого порядка, показанной на рис. 2.34

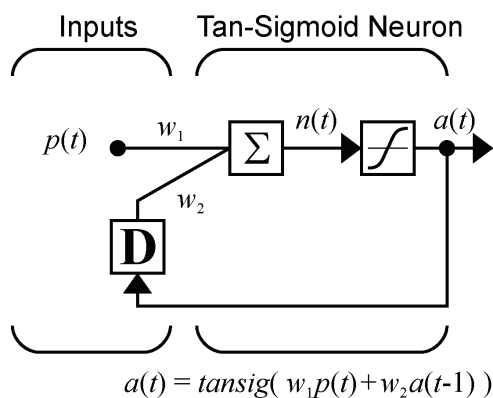


Рис. 2.37. Нелинейная рекуррентная сеть первого порядка

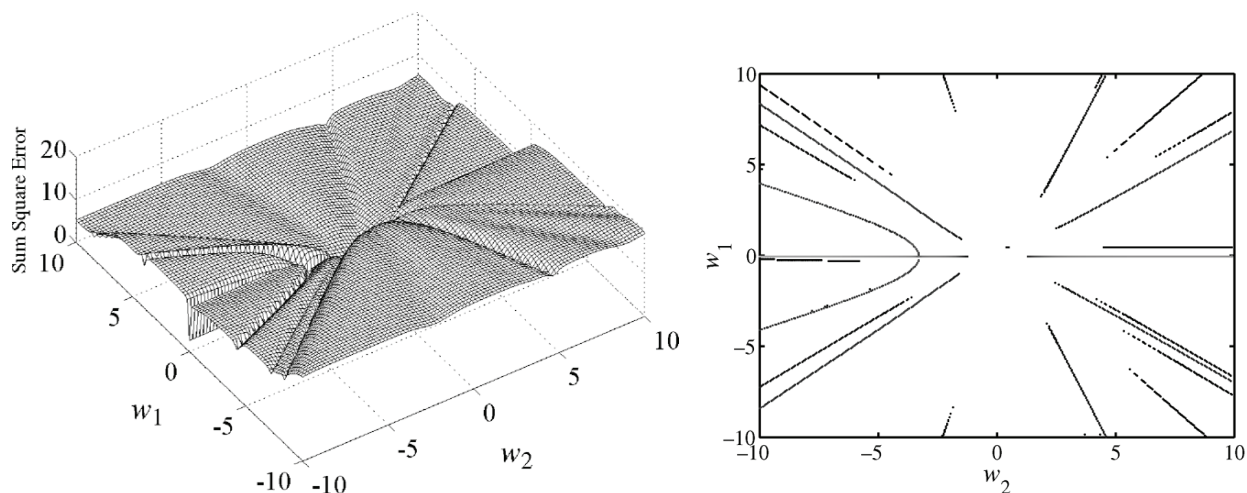


Рис. 2.38. Поверхность ошибки и расположение впадин для нелинейной сети

и, как следствие, увеличение числа настраиваемых параметров сети, приводит к значительному усложнению рельефа функции ошибки. Пример того, каким образом на рельеф функции ошибки влияет число примеров в обучающем наборе, показан на рис. 2.35. Рассмотрим теперь, каким образом влияет усложнение преобразования (2.59), осуществляемого сетью, на рельеф функции ошибки и, соответственно, на усложнение поиска минимума этой функции. Вначале усложним преобразование (2.59) за счет того, что линейную активационную функцию в нем заменим на нелинейную⁴⁰ [88]. Структура измененной таким образом сети, а также преобразование, выполняемое ею, показаны на рис. 2.37. Формирование обучающего набора, выбор значений весов связей и цели обучения для данной сети осуществляются точно так же, как и для линейной сети с рис. 2.34. Из рис. 2.38 видно, что для сети, отличающейся от про-

⁴⁰В качестве нелинейной активационной функции возьмем одну из двух сигмоидальных функций, а именно, гиперболический тангенс.

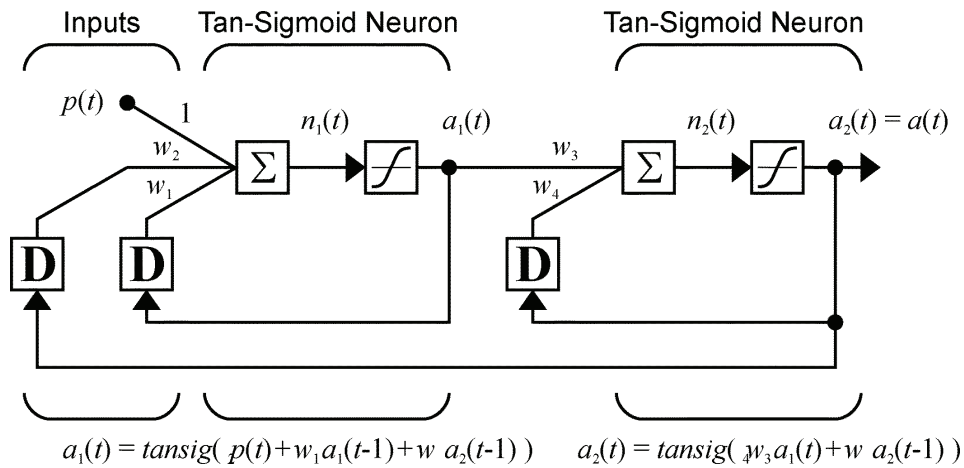


Рис. 2.39. Двухслойная нелинейная рекуррентная сеть

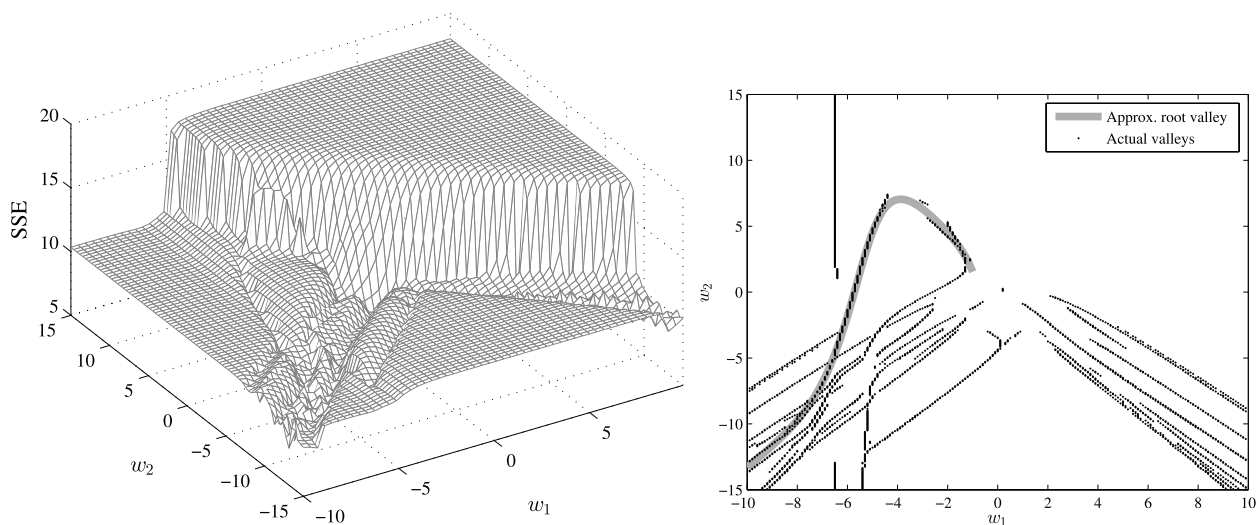


Рис. 2.40. Поверхность ошибки и расположение впадин для двухслойной нелинейной сети

стейшей за счет введения в нее нелинейной активационной функции, рельеф функции ошибки значительно усложнился за счет появления на нем большого числа узких впадин-ущелий.

Следующее усложнение сети призвано показать влияние увеличения числа ее слоев. В качестве примера на рис.2.39 показана двухслойная нелинейная рекуррентная сеть, более близкая к используемым для решения реальных прикладных задач. Формирование обучающего набора, выбор значений весов связей и цели обучения для данной сети осуществляются точно так же, как и двух предыдущих примерах. Рис. 2.40 показывает, что для двухслойной нелинейной сети, рельеф функции ошибки еще более усложнился: он изрезан большим числом узких ущелий, на нем появились также обширные плато. Поиск минимума функции с таким рельефом сопряжен со значительными трудностями. Здесь следует подчеркнуть, что

как и в двух предыдущих случаях изменение входного сигнала на многомерный приведет к значительному росту числа настраиваемых параметров сети и, как следствие, к еще большему усложнению рельефа функции ошибки.

2.4.1.5 Способы преодоления сложного рельефа функции ошибки

Проблему обучения рекуррентной НС-модели с учетом сложного рельефа функции ошибки для нее можно пытаться решить различными способами [87, 88, 188, 189]. К числу этих способов можно отнести следующие:

- регуляризация

$$J(\mathbf{w}) = SSE + \alpha \cdot SSW,$$

где SSE — суммарная среднеквадратическая ошибка сети, SSW — сумма квадратов весов;

- случайное варьирование стартовых значений весов;
- сочетание регулярного и генетического поиска;
- сегментирование обучающей последовательности (изменение входных данных сети меняет расположение впадин на рельефе функции ошибки).

1. Регуляризация представляет собой традиционный прием, связанный с модификацией исходной целевой функции (в рассматриваемом случае это функция ошибки сети), осуществляемой таким образом, чтобы «сгладить» по-возможности рельеф функции ошибки.

2. Подход с использованием случайного варьирования стартовых значений настраиваемых параметров НС-модели, основан на следующих соображениях. Очевидно, что стартовое значение настраиваемых параметров НС-модели (веса межнейронных связей, смещения, внутренние варьируемые параметры нейронов) существенным образом влияет на то, насколько успешным будет поиск решения, что будет определяться тем, в область притяжения какого из локальных минимумов (либо впадины-ущелья) стартовая точка попадет. Поскольку в подавляющем большинстве случаев не представляется возможным провести анализ, который дал бы «хорошую» стартовую точку, более рациональным является случайный выбор стартового значения настраиваемых параметров, точнее, использование набора таких стартовых точек, компоненты которых выбираются с использованием датчика равномерно распределенных случайных чисел.

3. Сочетание регулярного и генетического поиска представляет собой дальнейшее развитие варианта, описанного в предыдущем пункте. А именно, случайное варьирование формируемого решения относится не только к стартовой точке, но распространяется и непосредственно на процесс поиска решения. Использование генетического поиска, как известно [104, 105], позволяет, во-первых, работать не с одной траекторией поиска решения в пространстве варьируемых переменных, а с совокупностью таких траекторий (популяция решений) и, во-вторых, что более важно, использовать своего рода «прыжки» в пространстве варьируемых переменных. Это дает шансы «спрыгнуть» с плато или «выпрыгнуть» из ущелья, т. е. справиться с ситуациями, в которых регулярный поиск (любые методы градиентного типа) теряет работоспособность. Наиболее эффективным здесь будет сочетание регулярного поиска с периодическим «встряхиванием» оптимизируемой системы для разрешения ситуаций, связанных с «застреванием» регулярного поиска.

4. В сложных задачах (нелинейное многопараметрическое отображение, реализуемое сетью в сочетании с большим обучающим набором, характеризующим сложное поведение ДС), подходы, описанные в трех предыдущих пунктах, оказываются, как правило, малоэффективными. Существенно более мощным является подход, основанный на сегментировании обучающей последовательности. Недостатком его является значительный объем вычислительных ресурсов, требуемых для решения конкретной задачи.

Суть данного подхода состоит в следующем. Как уже отмечалось выше, процесс обучения рекуррентных нейронных сетей на длинных входных последовательностях сопряжен с определенными затруднениями, в числе которых наличие ложных долин в поверхности ошибки [88], эффект экспоненциального уменьшения либо роста нормы градиента [89], неограниченный рост выходов сети в тех случаях, когда траектории, соответствующие начальным условиям из обучающего множества, оказываются неустойчивыми. По этим причинам, лишь для небольшого набора начальных значений параметров сети удастся найти глобальный минимум с помощью градиентных методов оптимизации. Если перейти к решению задачи поиска начальных значений параметров, достаточно близких к минимуму, то можно предположить, что они являются решениями схожих задач. То есть, требуется найти такую последовательность задач, что:

- первая задача является достаточно простой — и ее решение может быть найдено для любых начальных значений параметров;
- каждая последующая задача схожа с предыдущей — их решения близки в пространстве

значений параметров;

- последовательность сходится к исходной, требуемой задаче.

Последовательно обучая сеть на данных задачах, можно надеяться достигнуть достаточно глубокого минимума. Подходы, основанные на схожих идеях, предлагались и ранее [90–93]: как правило, предполагалось обучение сети на последовательности задач возрастающей сложности (хотя это, по-видимому, не является обязательным требованием). Применение такого рода алгоритмов в большинстве случаев приводило к значительному улучшению результатов обучения. В данном случае, для задачи многошагового прогноза, естественным образом можно предложить следующую последовательность задач:

- задача одношагового прогноза;
- задача двухшагового прогноза;
-
- задача N -шагового прогноза.

Очевидно, первая задача является наиболее простой — более того, при ее решении рекуррентная сеть будет обучаться как обычная сеть прямого распространения. Требуемая задача N -шагового прогноза является наиболее сложной, поскольку обучение сети будет производиться на наиболее длинной последовательности. Итак, целевая функция для задачи прогноза на k шагов будет иметь следующий вид:

$$J_k = \frac{1}{k(n-k)} \sum_{i=1}^{n-k} \sum_{j=1}^k (x_{i+j} - \text{net}(\dots \text{net}(x_i, u_i; w), \dots, u_{i+j-1}; w))^2, \quad (2.61)$$

где x_i — вектор переменных состояния в дискретный момент времени i ; u_i — вектор переменных управления в дискретный момент времени i ; w — вектор настраиваемых параметров НС-модели.

Алгоритм обучения, согласно рассматриваемой процедуре, состоит из следующих шагов:

Шаг 1. Подготовить обучающее множество $X^{\text{train}} \leftarrow \{x_i^{\text{train}}, u_i^{\text{train}}\}_{i=1}^n$ и контрольное множество $X^{\text{val}} \leftarrow \{x_i^{\text{val}}, u_i^{\text{val}}\}_{i=1}^n$.

Шаг 2. Выбрать значение целевой точности $\varepsilon^{\text{goal}}$.

Шаг 3. Выбрать значение максимально допустимого роста погрешности Δ^{max} .

Шаг 4. Выбрать значение максимально допустимого числа эпох с ростом погрешности на контрольном множестве s^{max} .

Шаг 5. Выбрать начальные значения параметров w_0 (например, случайные).

Шаг 6. Установить текущее число эпох с ростом погрешности на контрольном множестве $s \leftarrow 0$.

Шаг 7. Установить текущее число шагов прогноза $k \leftarrow 1$.

Шаг 8. Решить задачу оптимизации $w_1 \leftarrow \operatorname{argmin}_w J_1(X^{\text{train}}, w)$, $\varepsilon_1^{\text{train}} \leftarrow J_1(X^{\text{train}}, w_1)$.

Шаг 9. Если $\varepsilon_1^{\text{train}} > \varepsilon^{\text{goal}}$, то вернуться к шагу 5.

Шаг 10. Вычислить погрешность $(n - 1)$ -шагового прогноза на контрольном множестве $\varepsilon_1^{\text{val}} \leftarrow J_{n-1}(X^{\text{val}}, w_1)$.

Шаг 11. Установить новое число шагов прогноза $k^+ \leftarrow k$.

Шаг 12. До тех пор, пока $k^+ \leq n - 1$ и $\varepsilon_{k^+}^{\text{train}} < \varepsilon_k^{\text{train}} + \Delta^{\text{max}}$, делать $k^+ \leftarrow k^+ + 1$.

Шаг 13. Если $k^+ = k$, то вернуться к шагу 5.

Шаг 14. Решить задачу оптимизации $w_{k^+} \leftarrow \operatorname{argmin}_w J_{k^+}(X^{\text{train}}, w)$,
 $\varepsilon_{k^+}^{\text{train}} \leftarrow J_{k^+}(X^{\text{train}}, w_{k^+})$.

Шаг 15. Если $\varepsilon_{k^+}^{\text{train}} > \varepsilon^{\text{goal}}$, установить $k^+ \leftarrow k^+ - 1$ и вернуться к шагу 13.

Шаг 16. Вычислить погрешность $(n - 1)$ -шагового прогноза на контрольном множестве $\varepsilon_{k^+}^{\text{val}} \leftarrow J_{n-1}(X^{\text{val}}, w_{k^+})$.

Шаг 17. Если $\varepsilon_{k^+}^{\text{val}} > \varepsilon_k^{\text{val}}$, установить $s \leftarrow s + 1$.

Шаг 18. Если $s \geq s^{\text{max}}$, то вернуться к шагу 5.

Шаг 19. Если $k^+ < n - 1$, установить $k \leftarrow k^+$, иначе закончить: w_{n-1} — искомые параметры НС-модели, являющиеся решением задачи $(n - 1)$ -шагового прогноза.

Данный алгоритм показал свою высокую эффективность в серии вычислительных экспериментов и был успешно применен для решения ряда задач моделирования и идентификации ДС. Примеры такого применения рассматриваются в разд. 6.

5. Чтобы показать, как работает описанный выше алгоритм, рассмотрим демонстрационный пример. Это рекуррентная сеть (см. рис. 2.41), которая содержит единственный настраиваемый параметр, соответственно, ее функция ошибки — это функция одной переменной, что удобно для визуализации процесса поиска решения. Данная сеть реализует рекуррентные соотношения вида:

$$\begin{cases} x_1(k+1) = \tanh(x_1(k) + wx_2(k) + p(k)), \\ x_2(k+1) = \tanh(x_1(k)). \end{cases}$$

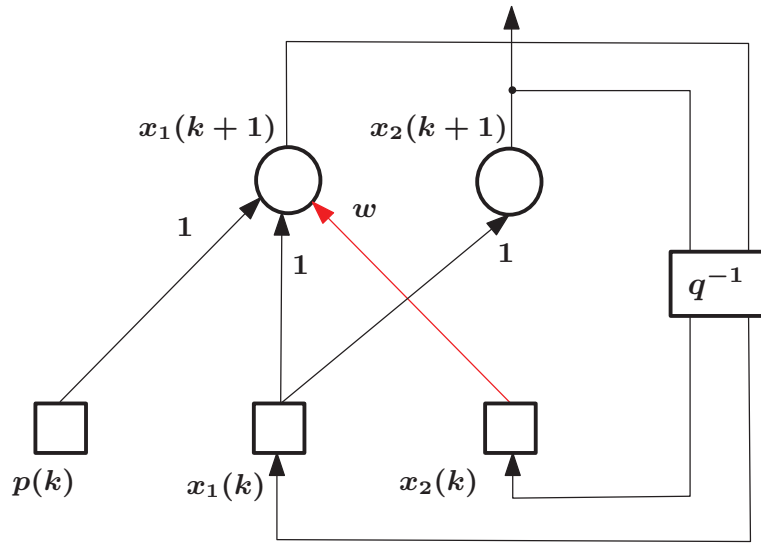


Рис. 2.41. Демонстрационный пример: Рекуррентная НС с одним настраиваемым параметром

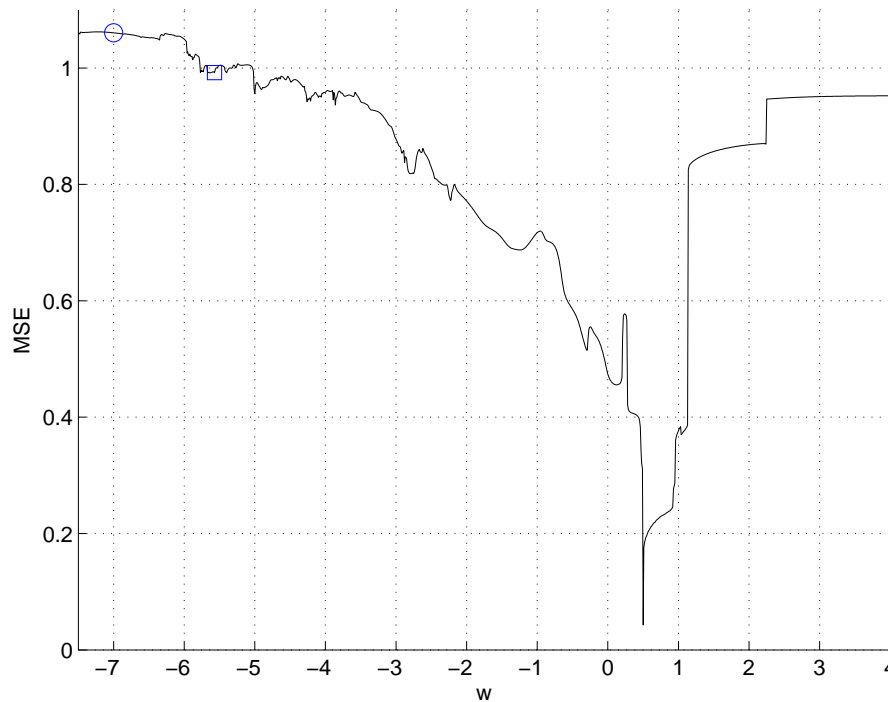


Рис. 2.42. Функция ошибки для задачи прогноза на 199 шагов, достигнут локальный минимум

Функция ошибки для этой сети в задаче прогноза на 199 шагов показана на рис.2.42. В процессе поиска на этом рельефе достигнут лишь локальный минимум функции ошибки.

На рис. 2.43 показано, как меняется рельеф функции ошибки для последовательности задач прогноза, когда число шагов составляет 2, 7, 148 и 199. Кривые, отвечающие различному числу шагов, помечены цифрами 1 (2-шаговый прогноз), 2 (7-шаговый прогноз), 3 (148-

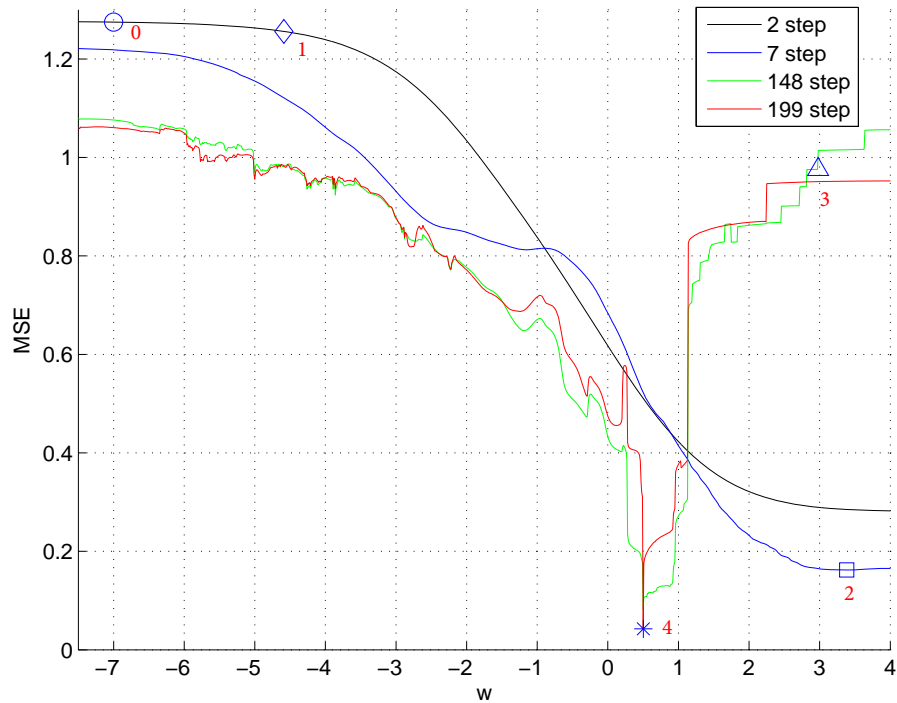


Рис. 2.43. Функции ошибки для последовательности задач прогноза, достигнут глобальный минимум

шаговый прогноз) и 4 (199-шаговый прогноз). Видно, что в итоге достигнут глобальный минимум функции ошибки.

В разд. 6.2 приводится пример, позволяющий оценить работоспособность данного подхода к организации обучения рекуррентных НС-моделей на примере существенно более сложной прикладной задачи.

2.4.2 Алгоритмы обучения динамических НС-моделей

Основным инструментом, используемым для обучения динамических НС-моделей, выполненным в виде рекуррентных сетей, являются градиентные алгоритмы обучения. К числу таких алгоритмов относятся, в частности, следующие, наиболее часто применяемые для решения задачи обучения динамических сетей [77–79]:

- обратное распространение во времени (BPTT — Back Propagation Through Time);
- рекуррентное обучение в реальном времени (RTRL — Real-Time Recurrent Learning);
- расширенный фильтр Калмана (EKF — Extended Kalman Filter).

Рассмотрим кратко основные особенности этих алгоритмов.

2.4.2.1 Обратное распространение во времени

В алгоритме ВРТТ отклик НС-модели вычисляется для всего набора моментов времени, в которые фиксировались данные, включаемые в обучающий набор. Затем производится вычисление градиента, начиная с завершающего момента времени и далее назад, в обратном времени, по направлению к входу сети.

Алгоритм ВРТТ достаточно эффективен в вычислительном плане, однако его затруднительно использовать в оперативном (on-line) режиме (вначале должен быть достигнут завершающий момент времени, что не всегда возможно по смыслу решаемой задачи).

Основная идея алгоритма ВРТТ состоит в том, что любую рекуррентную сеть можно представить в виде эквивалентной ей по поведению слоистой сети прямого распространения с числом слоев, равным числу временных тактов работы рекуррентной сети.

На рис. 2.44 показан пример (см. [94], с. 355; [95]) развертывания рекуррентной сети в эквивалентную ей по поведению многослойную сеть прямого распространения. Характерные особенности сети В, показанной на этом рисунке состоят в том, что: 1) веса связей от слоя к слою не меняются; 2) значения одноименных весов в сетях А и В одни и те же.

Ошибка данной сети на отрезке времени от n_0 до n_1 вычисляется следующим образом:

$$E(n_0, n_1) = \frac{1}{2} \sum_{n=n_0}^{n_1} \sum_{j \in A} e_j^2, \quad (2.62)$$

где A — множество индексов j тех нейронов, для которых заданы желаемые отклики; $e_j(n)$ — сигнал ошибки для этих нейронов.

Невязка j -го нейрона (для всех $j \in A$):

$$\delta_j(n) = -\frac{\partial E(n_0, n_1)}{\partial v_j(n)}, \quad (2.63)$$

где $v_j(n)$ — выход сумматора j -го нейрона, $\varphi(v_j(n))$ — активационная функция, вычисляется следующим образом:

$$\delta_j(n) = \begin{cases} \varphi'(v_j(n))e_j(n), & n = n_1; \\ \varphi'(v_j(n)) \left[e_j(n) + \sum_{k \in A} w_{jk} \delta_k(n+1) \right], & n_0 < n < n_1. \end{cases} \quad (2.64)$$

Вычисления для $\delta_j(n)$ повторяются с момента времени n_1 , шаг за шагом, пока не будет достигнут момент времени n_0 .

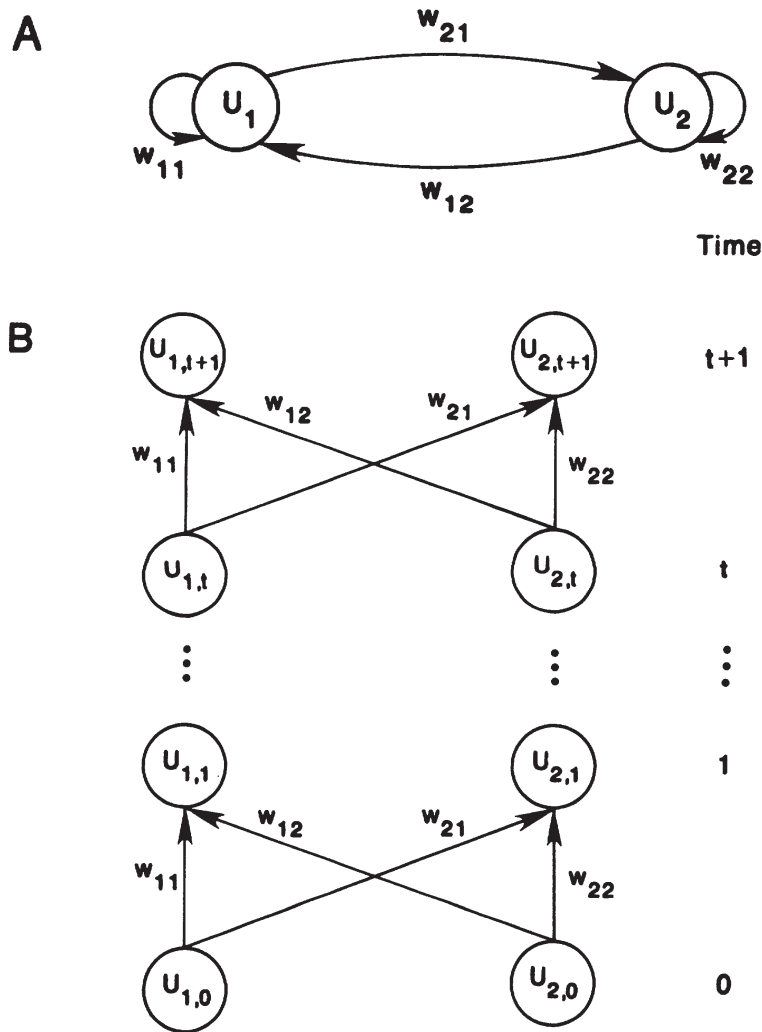


Рис. 2.44. Пример развертывания рекуррентной сети **A** в эквивалентную ей по поведению многослойную сеть **B** прямого распространения: **A** — полносвязная сеть из двух элементов; **B** — сеть прямого распространения с поведением таким же, как и у сети **A**

После вычисления невязок $\delta_j(n)$ всех нейронов $j \in A$ можно найти корректировки синаптических весов w_{ji} :

$$\Delta w_{ji} = -\eta \frac{\partial E(n_0, n_1)}{\partial w_{ji}} = \eta \sum_{n=n_0+1}^{n_1} \delta_j(n) x_i(n-1), \quad (2.65)$$

где η — скорость обучения; $x_i(n-1)$ — входной сигнал, поданный на i -й синапс j -го нейрона в момент времени $n-1$. Сравнение алгоритма ВРТТ со стандартным алгоритмом обратного распространения ошибки (ВР) показывает, что главное различие между ними состоит в том, что для ВРТТ, в отличие от ВР, известны желаемые отклики для всех слоев эквивалентной сети прямого распространения, так как эти слои получены дублированием выходного слоя исходной рекуррентной сети.

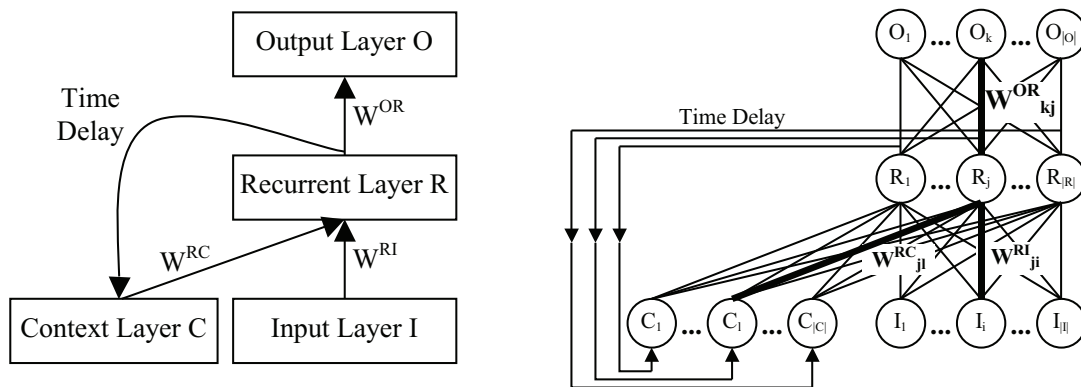


Рис. 2.45. Сеть Элмана: I — входной слой; R — рекуррентный (скрытый) слой; O — выходной слой; C — контекстный слой; W^{RI} , W^{RC} , W^{OR} — матрицы весов связей между входным и рекуррентным, контекстным и рекуррентным, рекуррентным и выходным слоями, соответственно

2.4.2.2 Рекуррентное обучение в реальном времени

В алгоритме BPTT вычисление градиента производится, начиная с завершающего момента времени и далее назад, в обратном времени, по направлению к входу сети, т.е. для вычисления градиента надо вначале дойти до завершающего момента времени. В отличие от этого, в алгоритме RTRL градиент может быть вычислен в тот же самый момент времени, когда получена реакция НС-модели, т.е. его стартовой точкой является начальный момент времени, а затем используется распространение вперед (в прямом времени).

Алгоритм RTRL требует большего объема вычислений, чем алгоритм BPTT, однако он пригоден для обучения НС-модели в оперативном режиме (on-line). В то же время, для вычисления якобиана RTRL в общем случае более эффективен, чем BPTT за счет того, что в RTRL вычисление якобиана является составной частью процесса вычисления градиента.

Алгоритм RTRL, как и алгоритмы BP и BPTT, существует в различных модификациях для различных нейросетевых архитектур. Все эти модификации основаны на одних и тех же идеях, но отличаются техническими деталями реализации. Проиллюстрировать принципы, на которых основан алгоритм RTRL, удобно на примере рекуррентной сети Элмана, показанной на рис. 2.45 [96, 97].

Для данной сети входные паттерны слоев для момента времени t имеют вид:

$$\begin{aligned} I^{(t)} &= (I_1^{(t)}, \dots, I_j^{(t)}, \dots, I_{NI}^{(t)}) \text{ — во входном слое,} \\ R^{(t)} &= (R_1^{(t)}, \dots, R_j^{(t)}, \dots, R_{NR}^{(t)}) \text{ — в рекуррентном слое,} \\ O^{(t)} &= (O_1^{(t)}, \dots, O_j^{(t)}, \dots, O_{NO}^{(t)}) \text{ — в выходном слое,} \\ C^{(t)} &= (C_1^{(t)}, \dots, C_j^{(t)}, \dots, C_{NC}^{(t)}) \text{ — в контекстном слое.} \end{aligned}$$

Выходная активность элементов рекуррентного слоя определяется соотношениями:

$$\tilde{R}_j^{(t)} = \sum_j W_{ij}^{RI} I_j^{(t)} + \sum_j W_{ij}^{RC} R_j^{(t-1)}, \quad R_j^{(t)} = f(\tilde{R}_j^{(t)}), \quad (2.66)$$

а выходная активность элементов выходного слоя — соотношениями:

$$\tilde{O}_j^{(t)} = \sum_j W_{ij}^{OR} R_j^{(t)}, \quad O_j^{(t)} = f(\tilde{O}_j^{(t)}). \quad (2.67)$$

Корректировка весов связей для момента времени t проводится согласно приводимым ниже соотношениям. А именно, веса связей между рекуррентным и выходным слоями вычисляются следующим образом:

$$\Delta W_{ij}^{OR} = \alpha (D_i^{(t)} - O_i^{(t)}) f'(\tilde{O}_i^{(t)}) R_j^{(t)}, \quad (2.68)$$

где $D^{(t)} = (D_1^{(t)}, \dots, D_j^{(t)}, \dots, D_{|O|}^{(t)})$ — желаемые значения выходов.

Веса связей между входным и рекуррентным слоями определяются соотношениями вида:

$$\begin{aligned} \Delta W_{ji}^{RI} &= \alpha \sum_k^{|O|} \left[(D_k^{(t)} - O_k^{(t)}) f'(\tilde{O}_k^{(t)}) \sum_{h=1}^{|R|} W_{kh}^{RC} \frac{\partial R_h^{(t)}}{\partial W_{ji}^{RI}} \right], \\ \frac{\partial R_h^{(t)}}{\partial W_{ji}^{RI}} &= f'(\tilde{R}_i^{(t)}) \left[I_i^{(t)} \delta_{hj}^{kron} + \sum_{l=1}^{|R|} W_{hl}^{RC} \frac{\partial R_l^{(t-1)}}{\partial W_{ji}^{RI}} \right], \end{aligned} \quad (2.69)$$

где $|I|$, $|R|$, $|O|$ — количество элементов в соответствующих слоях; $\delta_{hj}^{kron} = 1$ для $h = i$, $\delta_{hj}^{kron} = 0$ для $h \neq i$.

Веса связей между контекстным и рекуррентным слоями вычисляются согласно выражениям:

$$\begin{aligned} \Delta W_{ji}^{RC} &= \alpha \sum_k^{|O|} \left[(D_k^{(t)} - O_k^{(t)}) f'(\tilde{O}_k^{(t)}) \sum_{h=1}^{|R|} W_{kh}^{RC} \frac{\partial R_h^{(t)}}{\partial W_{ji}^{RC}} \right], \\ \frac{\partial R_h^{(t)}}{\partial W_{ji}^{RC}} &= f'(\tilde{R}_i^{(t)}) \left[R_i^{(t-1)} \delta_{hj}^{kron} + \sum_{l=1}^{|R|} W_{hl}^{RC} \frac{\partial R_l^{(t-1)}}{\partial W_{ji}^{RC}} \right]. \end{aligned} \quad (2.70)$$

Алгоритм RTRL может использоваться для обучения широкого класса динамических сетей, в том числе на основе архитектуры LDDN (Layered Digital Dynamic Network), к которой можно привести большинство существующих рекуррентных сетей [74, 98, 99].

2.4.2.3 Расширенный фильтр Калмана

Еще один вариант алгоритма обучения рекуррентных сетей, ориентированный на работу в оперативном (on-line) режиме — это расширенный фильтр Калмана (РФК) [75].

Будем считать, что для идеальной НС-модели наблюдаемый процесс является стационарным, т. е. $w_{k+1} = w_k$, но состояния его (веса w_k) «испорчены» шумами ξ_k .

Фильтр Калмана (ФК) в его стандартном варианте применим только для систем, наблюдения которых линейны по оцениваемым параметрам, в то время как уравнение наблюдения нейросети является нелинейным:

$$\begin{aligned}w_{k+1} &= w_k + \xi_k, \\ \hat{y}_k &= f(u_k, w_k) + \nu_k,\end{aligned}$$

где u_k — управляющие воздействия, ξ — шум объекта и ν — шум наблюдений, эти шумы являются гауссовскими случайными последовательностями с нулевым средним и ковариационными матрицами Q и R .

Для того, чтобы использовать ФК, требуется линеаризовать уравнение наблюдения. Можно использовать статистическую линеаризацию, т. е. линеаризацию относительно математического ожидания. Она дает:

$$\begin{aligned}w_{k+1} &= w_k + \xi_k, \\ \hat{y}_k &= H_k w_k + \nu_k,\end{aligned}$$

где матрица наблюдения имеет вид

$$H_k = \left. \frac{\partial \hat{y}}{\partial w^T} \right|_{\substack{w=w_k \\ x=x_k}} = - \frac{\partial e_k}{\partial w_k^T} = -J_k.$$

Здесь e_k — вектор ошибки наблюдения на k -м шаге оценивания.

Уравнения расширенного фильтра Калмана для оценки w_{k+1} на следующем шаге имеют вид:

$$\begin{aligned}S_k &= H_k P_k H_k^T + R_k, \\ K_k &= P_k H_k^T S_k^{-1}, \\ P_{k+1} &= (P_k - K_k H_k P_k) e^\beta + Q_k, \\ w_{k+1} &= w_k + K_k e_k,\end{aligned}$$

Здесь e_k — вектор ошибки наблюдения на k -м шаге оценивания:

$$e_k = y_k - \hat{y}_k = y_k - f(x_k, w_k),$$

а величина β — коэффициент забывания, влияет на значимость предыдущих шагов. Здесь обозначено также: K_k — калмановский коэффициент усиления, S_k — ковариационная матрица ошибок e_k , P_k — ковариационная матрица ошибок оценивания ($\hat{w}_k - w_k$).

Существуют альтернативные варианты РФК, которые могут оказаться более эффективными при решении рассматриваемых задач, в частности, такой:

$$\begin{aligned} P_k^- &= P_k + Q_k, \\ S_k &= H_k P_k^- H_k^T + R_k, \\ K_k &= P_k^- H_k^T S_k^{-1}, \\ P_{k+1} &= (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k K_k^T, \\ w_{k+1} &= w_k + K_k e_k. \end{aligned}$$

Вариант РФК данного вида более устойчив в вычислительном отношении, обладает робастностью к ошибкам округления, что положительно влияет на обеспечение вычислительной устойчивости процесса обучения НС-модели в целом.

Как видно из соотношений, определяющих РФК, ключевым моментом опять является вычисление якобиана J_k ошибок сети по настраиваемым параметрам.

При обучении нейросети использовать в РФК только текущее измерение нельзя из-за недопустимо низкой точности определения направления поиска (влияние шумов ξ и ν), необходимо формировать векторную оценку на интервале наблюдений, тогда обновление матрицы P_k происходит более корректно.

В качестве вектора наблюдений можно взять последовательность значений на некотором скользящем интервале:

$$\hat{y}_k = [\hat{y}_{i-l}, \hat{y}_{i-l+1}, \dots, \hat{y}_i]^T,$$

где l — длина скользящего интервала, индекс i относится к моменту времени (шагу дискретизации), а индекс k указывает номер оценки. Ошибка формируемой НС-модели

$$e_k = [e_{i-l}, e_{i-l+1}, \dots, e_i]^T$$

также будет векторной величиной.

2.5 Адаптивность НС-моделей

2.5.1 НС-модели со вставочными подсетями

С точки зрения обеспечения адаптивности НС-моделей весьма плодотворной является идея вставочного нейрона и объединения таких нейронов — вставочной подсети.

2.5.1.1 Концепция вставочного нейрона и НС-модели с такими нейронами

Эффективный подход к реализации адаптивных НС-моделей, основанный на концепциях вставочного нейрона и сети с преднастройкой, был предложен А. И. Самариним в [100]. Как отмечается в данной работе, одно из основных свойств НС-моделей, делающих их привлекательным средством для решения разнообразных прикладных задач, состоит в том что сеть может меняться, подстраиваясь под решаемую задачу. Такого рода подстройка может осуществляться в следующих направлениях:

- нейронная сеть может обучаться, т. е. менять значения своих настроечных параметров (это, как правило, синаптические веса межнейронных связей сети);
- нейронная сеть может менять свою структурную организацию за счет добавления или изъятия нейронов и перестройки межнейронных связей;
- нейронная сеть может динамически преднастраиваться под решение текущей задачи за счет замещения некоторых ее составных частей (подсетей) ранее подготовленными «заготовками», либо за счет изменения значений настроечных параметров сети и ее структурной организации на основе подготовленных заранее соотношений, связывающих решаемую задачу с требуемыми изменениями в НС-модели.

Первый из этих вариантов приводит к традиционному обучению НС-моделей, второй — к классу растущих сетей, третий — к сетям с преднастройкой.

Важнейшее ограничение, связанное с особенностями первого из перечисленных подходов к подстройке НС-моделей (обучение), состоит в том, что сеть, до того как ее начали обучать, потенциально пригодна для решения широкого класса задач, но после завершения процесса обучения она может решать уже только конкретную задачу, в случае возникновения другой задачи требуется переобучение сети на нее, при котором навык решения предыдущей задачи утрачивается.

Второй из подходов (растущие сети) позволяет справиться с указанной проблемой лишь частично. А именно, если появились новые обучающие примеры, которые не укладываются в

НС-модель, полученную согласно первому из подходов, то эта модель наращивается новыми элементами, с добавлением соответствующих связей, после чего проводится дообучение сети, не затрагивающее ранее построенную ее часть.

Третий из подходов является наиболее мощным и, соответственно, наиболее сложным. Следуя ему, необходимо либо организовать процесс динамической (т. е. непосредственно в ходе работы НС-модели) замены составных частей модели заранее подготовленными их альтернативными вариантами, отвечающими изменившейся задаче, либо организовать НС-модель в виде целостной интегрированной системы, в которой имеются специальные структурные элементы, называемые вставочными нейронами и вставочными подсетями, функция которых состоит в воздействии на рабочие элементы сети таким образом, чтобы их текущие характеристики отвечали специфике решаемой в данный момент конкретной задачи.

2.5.1.2 Вставочная подсеть как инструмент адаптации НС-моделей

Выше, в разд. 2.2.6, было введено понятие сетевой⁴¹ модели (СМ), которое обобщает понятие НС-модели. Здесь было показано, что СМ представляет собой совокупность взаимосвязанных элементов (элементов СМ), организованных как сетевые объединения, построенные по определенным правилам из очень небольшого числа исходных элементов-примитивов. Одним из возможных примеров элемента СМ может служить единичный искусственный нейрон.

Если при таком подходе следовать принципу минимализма, то наиболее перспективный путь, как уже отмечалось ранее, состоит в формировании очень ограниченного набора базисных элементов СМ. Тогда разнообразные конкретные виды элементов СМ, требуемые для получения НС-моделей, формируются как частные случаи базисных элементов. Обработывающие элементы СМ могут иметь две разновидности: рабочие элементы и вставочные элементы. Важнейшее отличие между ними состоит в том, что рабочие элементы осуществляют преобразование входных данных в требуемый выход НС-модели, т. е. в требуемый результат. Другими словами, совокупность взаимодействующих рабочих элементов реализует алгоритм решения требуемой прикладной задачи. В противоположность этому, вставочные элементы СМ не участвуют непосредственно в упомянутом алгоритме, их роль заключается в воздействии на рабочие элементы, например, за счет корректировки значений параметров рабочих элементов, что, в свою очередь, изменяет характер преобразования, реализуемого данным

⁴¹ Данный класс моделей можно также именовать композиционными, имея в виду фундаментальную роль операции композиции в структурном формировании моделей.

элементом.

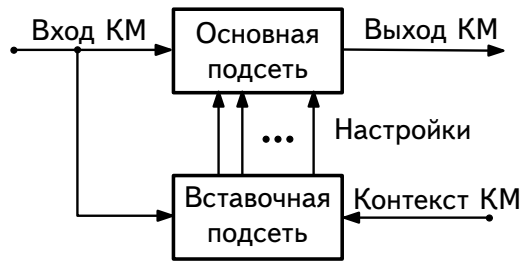


Рис. 2.46. Вставочные элементы в сетевой (композиционной) модели

Таким образом, вставочные элементы вводятся в СМ как средство контекстного воздействия на параметры и характеристики рабочих элементов СМ. Использование вставочных элементов — наиболее эффективный способ сделать сетевую (композиционную) модель адаптивной. Функциональную роль вставочных элементов поясняет рис. 2.46, на котором вставочные элементы объединены во вставочную подсеть. Видно, что вставочная подсеть получает те же самые входы, что и рабочая подсеть рассматриваемой СМ, реализующая основной алгоритм обработки входных данных. Кроме того, вставочная подсеть может получать также некоторую дополнительную информацию, называемую здесь контекстом СМ. Сообразно полученным исходным данным (входы НС-модели + контекст) вставочная подсеть вносит корректировки в рабочую подсеть таким образом, чтобы эта рабочая подсеть отвечала изменившейся задаче. Обучение вставочной подсети проводится заранее, на этапе формирования НС-модели, так что смена решаемой задачи не требует дообучения (и, тем более, переобучения) рабочей подсети, производится лишь ее перенастройка, требующая небольшого времени.

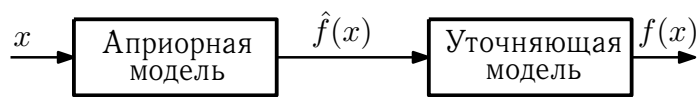
2.5.1.3 Преднастройка НС-моделей и ее возможные варианты

Будем выделять два варианта преднастройки: сильную преднастройку и слабую преднастройку.

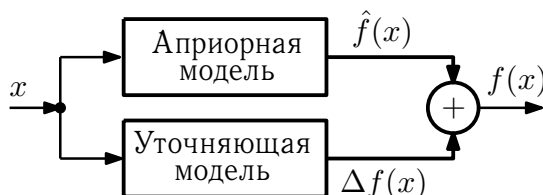
Сильная преднастройка ориентирована на адаптацию НС-модели в широком диапазоне условий. Характерным архитектурным признаком НС-модели в этом случае является наличие в составе обрабатывающих элементов СМ, наряду с рабочими элементами также и вставочных элементов, воздействующих на параметры рабочих элементов СМ. Такой подход позволяет реализовывать как параметрическую, так и структурную адаптацию НС-модели.

Слабая преднастройка не использует вставочные элементы. При ней выделяются фрагменты НС-модели, меняющиеся при изменении условий и проводится подстройка данных

фрагментов по двухступенчатой схеме. Например, пусть решается задача моделирования движения летательного аппарата. В качестве основы требуемой модели используется система дифференциальных уравнений, описывающая движение ЛА. Данная система согласно схеме, которая излагается в разд. 5, преобразуется в НС-модель. Это общая модель, которая должна быть уточнена применительно к определенному ЛА за счет задания конкретных значений его геометрических, массовых, инерционных, аэродинамических характеристик. Наиболее сложной является проблема конкретизации аэродинамических характеристик моделируемого ЛА вследствие неполного и неточного знания соответствующих величин. В такой ситуации целесообразно АДХ представлять в виде двухкомпонентной структуры: первая из них основана на априорном знании (например, на данных, полученных путем эксперимента в АДТ), вторая содержит уточняющие данные, получаемые непосредственно в полете. Преднастройка НС-модели в данном случае осуществляется за счет того, что при переходе от моделирования одного конкретного ЛА к другому в НС-модели заменяется часть описания АДХ, основанная на априорном знании. Уточняющая часть этого описания является инструментом адаптации НС-модели, выполняемой уже в процессе функционирования моделируемого объекта.



(a)



(b)

Рис. 2.47. Структурные варианты преднастройки НС-модели: (a) — последовательный вариант; (b) — параллельный вариант

В обоих вариантах, как последовательном, так и параллельном, априорная модель обучается в автономном режиме (off-line) заранее с использованием доступных знаний о моделируемом объекте. Уточняющая модель подстраивается уже непосредственно в процессе функционирования объекта на основе данных, получаемых в оперативном режиме (on-line).

В последовательном варианте (рис. 2.47a) выход $\hat{f}(x)$ априорной модели, отвечающий данному конкретному значению входного вектора x , является входом для уточняющей модели,

реализующей преобразование $f(\hat{f}(x))$.

В параллельном варианте (рис. 2.47b) априорная и уточняющая модели действуют независимо друг от друга, вычисляя оценку $\hat{f}(x)$, отвечающую данному конкретному значению входного вектора x и исходному знанию о моделируемом объекте, а также поправку $\Delta f(x)$ для того же самого значения входного вектора x с учетом данных, которые стали доступны для использования в процессе функционирования объекта. Искомое значение $f(x)$ является суммой этих компонент, т. е. $f(x) = \hat{f}(x) + \Delta f(x)$.

Следует подчеркнуть, что нейросетевая реализация априорной и уточняющей моделей является, как правило, различной с точки зрения привлекаемых архитектурных решений, хотя в частном случае может быть и одинаковой, например обе эти модели могут быть построены в виде мультиперсептронов с сигмоидальными активационными функциями. Это позволяет наиболее эффективно удовлетворять требованиям, которые, вообще говоря, для априорной и уточняющей моделей различны. В частности, основное требование, предъявляемое к априорной модели, состоит в возможности представления с требуемой точностью сложных нелинейных зависимостей, при этом время, затрачиваемое на обучение такой модели не критично, поскольку это обучение осуществляется в автономном режиме. В то же время, уточняющая модель в своей работе должна укладываться в очень жесткие рамки реального (или даже опережающего) масштаба времени. По этой причине, в частности, в абсолютном большинстве случаев неприемлемыми будут нейроархитектуры, требующие полного их переобучения даже при незначительных изменениях в обучающих данных, с которыми они работают. В такой ситуации более целесообразен инкрементный подход к обучению и дообучению НС-моделей, позволяющий не проводить заново обучение всей сети, а лишь откорректировать те ее элементы, которые непосредственно связаны с изменившимися обучающими данными.

2.5.2 Инкрементное формирование НС-моделей

1. Одним из инструментов адаптации НС-моделей является инкрементное формирование, существующее в двух вариантах: параметрическом и структурно-параметрическом.

При параметрическом варианте инкрементного формирования структурная организация НС-модели задается сразу и фиксируется, после чего в несколько этапов осуществляется ее инкрементная подстройка (обучение/дообучение), например, для расширения области режимов функционирования ДС, в которой модель работает с требуемой точностью.

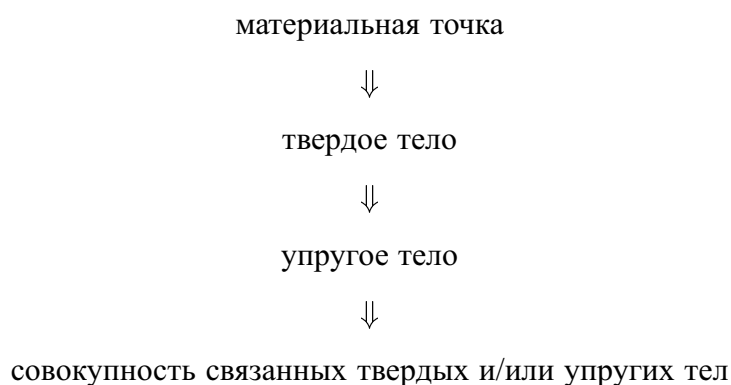
Например, если взять полную пространственную модель движения ЛА, учитывающую как

его траекторное, так и угловое движение, то в соответствии с инкрементным подходом вначале осуществляется автономное (off-line) обучение данной модели на относительно небольшой подобласти значений переменных состояния и управления, а затем в оперативном (on-line) режиме выполняется инкрементное дообучение НС-модели, при котором на каждом шаге этого процесса производится поэтапное расширение подобласти, на которой обеспечивается работоспособность модели с тем, чтобы в итоге расширить данную подобласть до полной области.

2. При структурно-параметрическом варианте инкрементного формирования вначале строится «усеченная» (по составу переменных состояния и соотношений, связывающих эти переменные) НС-модель, которая обучается, соответственно, лишь на части переменных и их областей определения. Эта исходная модель затем поэтапно расширяется за счет введения в нее новых переменных с последующим дообучением.

Например, в качестве исходной задается модель продольного углового движения ЛА, расширяемая затем за счет добавления траекторного продольного движения, после чего в нее добавляются компоненты бокового движения, т. е. модель за несколько шагов доводится до требуемой полной модели пространственного движения ЛА.

Структурно-параметрический вариант инкрементного формирования НС-моделей позволяет начинать с простой модели, последовательно усложняя ее, например, по схеме



Это дает возможность поэтапно наращивать модель в структурном отношении.

3 Нейросетевое моделирование управляемого движения летательных аппаратов — подход на основе моделей типа «черный ящик»

Как уже отмечалось выше, управление движением современных и перспективных самолетов приходится обеспечивать в условиях значительных и разнообразных неопределенностей в значениях их параметров и характеристик, режимов полета, воздействий внешней среды. Кроме того, в ходе полета могут возникать разнообразные нештатные ситуации, в частности, отказы оборудования и повреждения конструкции, которые необходимо парировать за счет реконфигурации системы управления и органов управления самолета.

Из сказанного следует, что ситуация, в которой оказывается самолет в каждый текущий момент времени, может меняться значительным и непредсказуемым заранее образом. Система управления самолета должна быть в состоянии эффективно приспосабливаться к этим изменениям за счет оперативного изменения параметров и/или структуры используемых законов управления. Удовлетворить этому требованию позволяет аппарат теории адаптивного управления [51, 54–62, 64–66].

К числу наиболее эффективных подходов к реализации концепций адаптивности можно отнести подход, основанный на методах и средствах нейросетевого моделирования и управления [54, 80, 82]. Важнейшим составным элементом процесса реализации данного подхода является получение нейросетевой модели объекта управления. Один из возможных вариантов формирования НС-моделей рассматривается в последующих разделах на примере задачи моделирования продольного углового движения маневренного самолета.

3.1 Нейросетевые эмпирические модели динамических систем (модели типа «черный ящик»)

1. Традиционно в качестве моделей динамических систем используются дифференциальные уравнения (для систем с непрерывным временем) или разностные уравнения (для систем с дискретным временем). Как уже отмечалось выше, в ряде случаев такие модели не удовлетворяют предъявляемым требованиям, в частности, требованию адаптивности, которая необходима при использовании модели в составе бортовых комплексов средств управления поведением ДС. Альтернативный вариант состоит в использовании НС-моделей, которые могут быть реализованы в адаптивном варианте.

В данном разделе для динамических систем рассматриваются НС-модели традиционного

эмпирического типа, т. е. модели типа «черный ящик» [131–138, 142–148]. В разд. 5 эти модели будут расширены до полуэмпирических за счет введения в них теоретического знания об объекте моделирования.

3.1.1 Основные виды моделей

1. Имеется два основных подхода к представлению (описанию) динамических систем [25, 26, 76]:

- представление в пространстве состояний ДС;
- представление в терминах входов и выходов ДС (вход-выходное представление).

Для упрощения описания подходов к моделированию ДС будем считать, что рассматриваемая система имеет единственный выход, т. е. реализуемый ею процесс характеризуется единственной величиной. На ДС с векторным выходом получаемые результаты обобщаются без каких-либо затруднений.

2. Про модель ДС принято говорить, что она является представлением системы в пространстве состояний, если данная модель имеет вид:

$$\begin{aligned}x(k) &= f(x(k-1), u(k-1), \xi_1(k-1)), \\y(k) &= g(x(k), \xi_2(k)),\end{aligned}\tag{3.1}$$

где вектор $x(k)$ — это вектор состояния (фазовый вектор) ДС, компоненты которого являются переменными величинами, характеризующими состояние объекта в момент времени t_k ; вектор $x(u)$ содержит в качестве компонент входные управляющие величины ДС; векторы $\xi_1(k)$ и $\xi_2(k)$ описывают возмущения, воздействующие на ДС; скалярная величина $y(k)$ — выход ДС; $f(\cdot)$ и $g(\cdot)$ — нелинейная вектор-функция и скалярная функция, соответственно. Размерность вектора состояния (т. е. число переменных состояния, входящих в этот вектор в качестве его компонент) принято именовать порядком модели. Переменные состояния могут быть как доступными для наблюдения и измерения их значений, так и ненаблюдаемыми. В частном случае в качестве выходной величины может использоваться какая-либо из переменных состояния ДС. Возмущения $\xi_1(k)$ и $\xi_2(k)$ могут влиять на значения выходов ДС и/или ее состояний. В противоположность входным управляющим воздействиям, возмущающие воздействия являются ненаблюдаемыми.

Формирование модели ДС в пространстве состояний заключается в нахождении приближенных представлений для функций $f(\cdot)$ и $g(\cdot)$, используя имеющиеся данные о ДС. В варианте, когда формируется модель типа «черный ящик», т. е. без привлечения какого-либо знания о природе и особенностях функционирования ДС, в качестве таких данных будут выступать последовательности значений входных и выходных величин ДС и тех переменных состояния, значения которых можно получить путем измерений.

3. Про модель ДС принято говорить, что она является вход-выходным представлением системы (представлением системы в терминах ее входов и выходов), если данная модель имеет вид:

$$y(k) = h(y(k-1), \dots, y(k-n), u(k-1), \dots, u(k-m), \xi(k-1), \dots, \xi(k-p)), \quad (3.2)$$

где $h(\cdot)$ — нелинейная функция, n — порядок модели, m и p — положительные целые константы, $x(u)$ — вектор входных управляющих сигналов ДС, $\xi(k)$ — вектор возмущений. Вход-выходное представление можно считать частным случаем представления в пространстве состояний, когда компоненты вектора состояний являются наблюдаемыми и трактуются как выходные сигналы ДС.

4. При моделировании линейных систем представление в пространстве состояний и вход-выходное представление эквивалентны друг другу [25, 26], поэтому можно выбирать то из них, которое удобнее и эффективнее с точки зрения решаемой задачи. В противоположность этому, при нелинейном моделировании представление в пространстве состояний является более общим и в то же время более экономичным (компактным) в сравнении с вход-выходным представлением. Однако реализация модели в пространстве состояний обычно несколько более трудоемка, чем вход-выходной модели из-за того, что требуется получить приближенное представление для двух отображений, $f(\cdot)$ и $g(\cdot)$ в (3.1) вместо единственного отображения $h(\cdot)$ в (3.2).

5. Вопрос о виде модели (в пространстве состояний или вход-выходная) не единственный, на который требуется ответить при моделировании нелинейной динамической системы. Помимо этого, важную роль играет способ введения возмущений в формируемую модель. Здесь возможны два варианта:

- возмущения влияют на состояния ДС;
- возмущения влияют на выходы ДС;
- возмущения влияют и на состояния, и на выходы ДС.

Как показано в [76], характер действия возмущений на ДС существенным образом влияет на структуру формируемой модели, вид требуемого алгоритма ее обучения и на характер функционирования сформированной модели. В следующем разделе эти вопросы рассматриваются более подробно.

3.1.2 Подходы к учету возмущений, действующих на ДС

Как уже отмечалось, способ введения возмущающих воздействий в модель существенным образом влияет как на структурную организацию модели, так и на алгоритм ее обучения.

3.1.2.1 Вход-выходное представление ДС

Рассмотрим вначале вариант, при котором шум воздействует на состояние ДС. Примем, что требуемое представление ДС имеет следующий вид:

$$y_p(k) = \psi(y_p(k-1), \dots, y_p(k-n), u(k-1), \dots, u(k-m)) + \xi(k), \quad (3.3)$$

где $y_p(k)$ — наблюдаемый (измеряемый) выход процесса, реализуемого ДС.

Будем считать, что на выход ДС воздействует аддитивный шум, при этом точка суммирования выходного сигнала и шума предшествует точке, из которой выходит сигнал обратной связи. В таком случае в момент времени k на выход системы этот шум будет влиять как непосредственно, так и через воздействие его на n предыдущих выходов.

В нелинейном моделировании такой структурный вариант отвечает модели типа NARX (Nonlinear Auto-Regressive with eXogenous inputs), т. е. нелинейной авторегрессии с внешними входами, в ее последовательно-параллельном варианте (см. рис. 3.1б).

Как уже отмечалось выше, аддитивный шум, воздействующий на выход ДС в рассматриваемом варианте, оказывает влияние не только непосредственно в момент времени k , но и через выходы на предшествующих n шагах, когда такое влияние также имело место. Необходимость учета предшествующих выходов обусловлена тем, что в идеальном варианте ошибка моделирования на шаге k должна быть равна значению шума в этот же момент времени. Соответственно, при формировании модели ДС необходимо принимать во внимание выходы системы в прошедшие моменты времени, чтобы компенсировать имевшие место шумовые воздействия. Соответствующая идеальная модель может иметь вид сети прямого распространения, реализующей отображение вида:

$$g(k) = \varphi_{NN}(y_p(k-1), \dots, y_p(k-n), u(k-1), \dots, u(k-m), w), \quad (3.4)$$

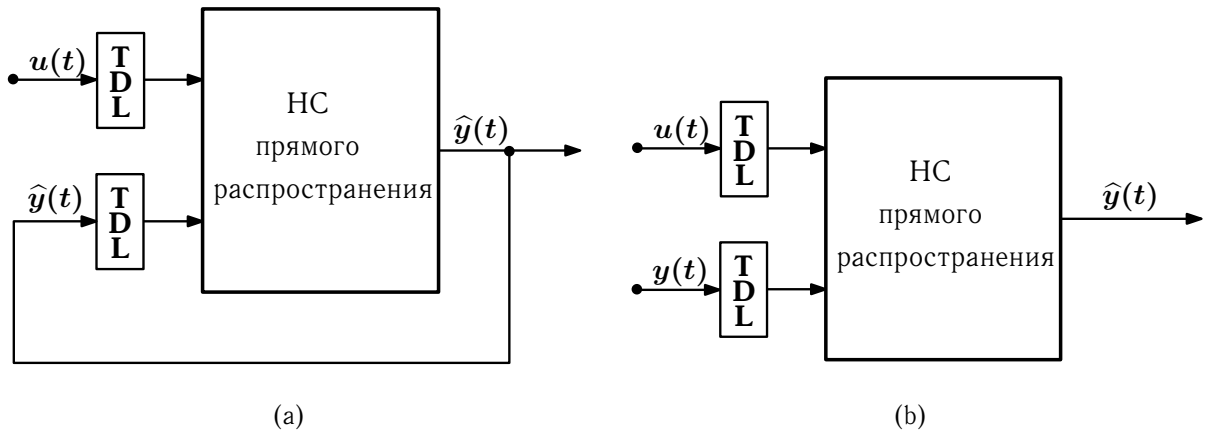


Рис. 3.1. Общая структура нейросетевой NARX-модели: **(а)** модель с параллельной архитектурой; **(б)** модель с последовательно-параллельной архитектурой

где w — вектор параметров, $\varphi_{NN}(\cdot)$ — функция, реализуемая сетью прямого распространения.

Пусть вектор параметров w сети выбран в процессе ее обучения таким образом, что $\varphi_{NN}(\cdot) = \varphi(\cdot)$, т.е. данная сеть точно воспроизводит выходы моделируемой ДС. В таком случае для всех моментов времени k будет выполняться соотношение

$$y_p(k) - g(k) = \xi(k), \quad \forall k \in \{0, N\},$$

т.е. ошибка моделирования равняется шуму, воздействию на выход ДС. Данную модель можно назвать идеальной в том смысле, что она точно отражает детерминированные составляющие процесса функционирования ДС и не воспроизводит тот шум, который искажает выходной сигнал системы. Входами данной модели являются значения управляющих величин, а также измеренные выходы процесса, реализуемого ДС. При этом идеальная модель, которая представляет собой одношаговый предиктор, обучается как сеть прямого распространения, а не как рекуррентная сеть. Вследствие этого, для настройки идеальной модели в данном случае целесообразно использовать методы обучения с учителем, имеющиеся для статических НС-моделей. Поскольку входы сети-предиктора включают помимо управляющих величин также и измеренные (наблюдаемые) значения выходов процесса, реализуемого ДС, выход модели рассматриваемого вида можно вычислить только на один временной шаг вперед (соответственно, предикторы такого типа принято именовать одношаговыми). Если же формируемая модель должна отражать поведение ДС на временном горизонте, превышающем один временной шаг, на вход предиктора придется подавать его же выходы в предшествующие моменты времени. В таком случае предиктор уже не будет обладать свойствами идеальной модели вследствие накопления ошибки прогнозирования.

Второй из видов воздействия шума на систему, требующий рассмотрения, это вариант, при котором шум воздействует на выход ДС. В этом случае соответствующее описание процессе, реализуемого ДС, имеет следующий вид:

$$\begin{aligned} x_p(k) &= \varphi(x_p(k-1), \dots, x_p(k-n), u(k-1), \dots, u(k-m)), \\ y_p(k) &= x_p(k) + \xi(k) \end{aligned} \quad (3.5)$$

При таком структурном варианте организации модели аддитивный шум добавляется непосредственно к выходному сигналу ДС (это параллельная разновидность архитектуры моделей данного вида, см. рис. 3.1а). По этой причине шум оказывает влияние только на текущий шаг процесса функционирования ДС. Поскольку выход модели в момент времени k зависит от шума только в этот же самый момент времени, для работы модели не требуются значения выходов, реализуемых ДС в предшествующие моменты времени, достаточно их оценок, формируемых самой моделью. Следовательно, в качестве «идеальной модели», аналогичной той, что была рассмотрена выше для последовательно-параллельного варианта, можно взять рекуррентную нейронную сеть, реализующую отображение вида:

$$g(k) = \varphi_{NN}(g(k-1), \dots, g(k-n), u(k-1), \dots, u(k-m), w), \quad (3.6)$$

где, как и в (3.4), w — вектор параметров, $\varphi_{NN}(\cdot)$ — функция, реализуемая сетью прямого распространения.

Пусть, как и в предыдущем случае, вектор параметров w сети выбран в процессе ее обучения таким образом, что $\varphi_{NN}(\cdot) = \varphi(\cdot)$. Примем также, что для первых n моментов времени ошибка прогнозирования равняется по своей величине шуму, воздействию на ДС. В таком случае для всех моментов времени k , $k = 0, \dots, n-1$ будет выполняться соотношение

$$y_p(k) - g(k) = \xi(k), \quad \forall k \in \{0, n-1\}.$$

Следовательно, ошибка моделирования будет численно равна шуму, воздействию на выход ДС, т. е. эту модель можно назвать идеальной, поскольку она точно отражает детерминированные составляющие процесса функционирования ДС и не воспроизводит тот шум, который искажает выходной сигнал системы.

Если начальные условия моделирования не удовлетворены (имеет место «неидеальность» модели в начальный момент времени), но условие $\varphi_{NN}(\cdot) = \varphi(\cdot)$ выполняется и модель устойчива к изменению начальных условий, ошибка моделирования с ростом значения k будет уменьшаться.

Как видно из приведенных соотношений, идеальная модель в параллельном варианте представляет собой динамическую рекуррентную сеть, в сравнении с последовательно-параллельным вариантом, когда идеальная модель представлялась статической сетью прямого распространения.

Соответственно, для обучения модели параллельного типа в общем случае требуется привлекать методы, рассчитанные на работу с динамическими сетями, что, конечно, сложнее в сравнении с методами обучения статических сетей. Однако для моделей рассматриваемого вида могут быть предложены методы обучения, использующие специфику этих моделей, менее трудоемкие в употреблении, чем обычные методы обучения динамических сетей. Возможные пути построения таких методов рассматриваются в разд. 2.4.

В силу характера воздействия шума на процесс функционирования параллельных моделей, они могут использоваться не только как одношаговые предикторы, как это имеет место в случае последовательно-параллельных моделей, но и как полноценные модели ДС, позволяющие анализировать поведение этих систем на интервале времени требуемой продолжительности, а не только на один временной шаг вперед.

Следующий вариант влияния шума на моделируемую систему состоит в том, что одновременно вводятся воздействия шума как на выходы, так и на состояния ДС. Такому варианту отвечает модель вида

$$\begin{aligned}x_p(k) &= \varphi(x_p(k-1), \dots, x_p(k-n), \\ &u(k-1), \dots, u(k-m), \xi(k-1), \dots, \xi(k-p)), \\ y_p(k) &= x_p(k) + \xi(k).\end{aligned}\tag{3.7}$$

Такие модели относятся к классу NARMAX (Nonlinear Auto-Regressive with Moving Average and eXogenous inputs), т. е. представляют собой нелинейную авторегрессию со скользящим средним и внешними входами. В рассматриваемом случае формируемая модель учитывает как предшествующие значения выходов ДС, так и предшествующие значения выходов самой модели. т. е. оценки выходов ДС. В силу того, что такая модель представляет собой комбинацию из двух моделей, рассмотренных ранее, она может использоваться только как одношаговый предиктор, аналогично модели с шумом, воздействующим на состояния.

3.1.2.2 Представление ДС в пространстве состояний

В предыдущем разделе было показано, каким образом влияет способ введения возмущающих воздействий в модель на ее структурную организацию и обучение для вход-выходного представления. Рассмотрим теперь представление ДС в пространстве состояний, которое в случае нелинейного моделирование, как отмечалось выше, обладает большей общностью по сравнению с вход-выходным представлением.

Рассмотрим вначале вариант, при котором шум воздействует на выход ДС. Примем, что требуемое представление ДС имеет следующий вид:

$$\begin{aligned}x(k) &= \varphi(x(k-1), u(k-1)), \\y(k) &= \psi(x(k)) + \xi(k).\end{aligned}\tag{3.8}$$

Поскольку в данном варианте шум присутствует только в уравнении наблюдения, он не оказывает влияния на динамику моделируемого объекта. Исходя из соображений, аналогичных тем, что приводились для случая вход-выходного представления ДС, идеальная модель в рассматриваемом случае будет иметь рекуррентную структуру, определяемую соотношениями:

$$\begin{aligned}x(k) &= \varphi_{NN}(x(k-1), u(k-1)), \\y(k) &= \psi_{NN}(x(k)),\end{aligned}\tag{3.9}$$

где $\varphi_{NN}(\cdot)$ — точное представление функции $\varphi(\cdot)$, а $\psi_{NN}(\cdot)$ — точное представление функции $\psi(\cdot)$.

Второй из видов воздействия шума на систему, требующий рассмотрения, это вариант, при котором шум воздействует на состояние ДС. В этом случае соответствующее описание процесса, реализуемого ДС, имеет вид:

$$\begin{aligned}x(k) &= \varphi(x(k-1), u(k-1), \xi(k-1)), \\y(k) &= \psi(x(k)).\end{aligned}\tag{3.10}$$

Основываясь на тех же соображениях, что и для случая вход-выходного представления ДС, можно сделать вывод, что в рассматриваемом случае входами идеальной модели, помимо управлений u , должны быть еще и переменные состояния процесса, реализуемого ДС. При этом возможны две ситуации:

- переменные состояния являются наблюдаемыми, тогда их можно интерпретировать как выходы системы и задача сводится к ранее рассмотренной для случая вход-выходного

представления; идеальной моделью при этом будет сеть прямого распространения, которая может использоваться как одношаговый предиктор;

- переменные состояния не являются наблюдаемыми, поэтому идеальную модель в данном случае построить нельзя; в таком случае следует воспользоваться вход-выходным представлением (с некоторой потерей общности модели), либо строить некоторую рекуррентную модель, хотя она и не будет в этой ситуации оптимальной.

Следующий вариант влияния шума на моделируемую систему состоит в том, что одновременно вводятся воздействия шума как на выходы, так и на состояния ДС. Такому варианту отвечает модель, описываемая соотношениями:

$$\begin{aligned}x(k) &= \varphi(x(k-1), u(k-1), \xi_1(k-1)), \\y(k) &= \psi(x(k), \xi_2(k)),\end{aligned}\tag{3.11}$$

Аналогично предыдущему случаю, опять возможны две ситуации:

- если переменные состояния являются наблюдаемыми, их можно интерпретировать как выходы ДС и задача сводится к ранее рассмотренной для случая вход-выходного представления;
- если переменные состояния не являются наблюдаемыми, идеальная модель должна включать как состояния, так и наблюдаемый выход системы.

3.2 Нейросетевая модель движения ЛА на основе многослойной нейронной сети

Многие схемы адаптивного управления требуют наличия модели объекта управления. Получение такой модели составляет содержание классической задачи идентификации динамических систем [107]. Одним из наиболее эффективных подходов к решению данной задачи применительно к нелинейным системам является, как показывает опыт [84, 137, 138], использование методов и средств искусственных нейронных сетей. Нейросетевое моделирование позволяет строить достаточно точные и эффективные в вычислительном плане НС-модели.

3.2.1 Общая структура НС-модели движения ЛА на основе многослойной нейронной сети

Основа вычислительной эффективности НС-моделей состоит в том, что искусственная нейронная сеть представляет собой алгоритмически универсальную математическую модель

[74, 77, 101], с помощью которой можно с любой наперед заданной точностью представить любое нелинейное отображение $\varphi : \mathcal{R}^n \rightarrow \mathcal{R}^m$, т.е. любую нелинейную зависимость между n -мерным вектором входных данных и m -мерным вектором выходных данных.

Формирование НС-модели управляемого движения объекта управления, предполагаемого нелинейным, трактуется далее как получение нейросетевой аппроксимации исходной математической модели движения самолета, заданной в той или иной форме, чаще всего в виде системы дифференциальных уравнений. Схема нейросетевой идентификации объекта управления, отвечающая такой трактовке, показана на рис. 3.2.

В качестве сигнала ошибки ε , направляющего обучение НС-модели, используется квадрат разности между выходом объекта управления y_p и нейросетевой модели y_m , находящихся под воздействием сигнала управления u . Обученная НС-модель реализует схему вычислений рекуррентного типа, в которой для вычисления значения выхода \hat{y} для момента времени t_{i+1} используются значения \hat{y} и u в момент времени t_i .

В качестве модели динамического объекта была выбрана нелинейная авторегрессионная сеть типа NARX с внешними входами (Nonlinear AutoRegressive network with eXogeneous inputs) как отвечающая характеру рассматриваемой задачи управления полетом ЛА. Она представляет собой рекуррентную динамическую слоистую НС-модель с элементами задержки (TDL — Time Delay Line) на входах сети и с обратными связями между слоями.

NARX-модель реализует динамическое отображение, описываемое разностным уравнением следующего вида:

$$\hat{y}(t) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-N_y), u(t-1), u(t-2), \dots, u(t-N_u)), \quad (3.12)$$

где значение выходного сигнала $\hat{y}(t)$ для данного момента времени t вычисляется на основе значений $\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-N_y)$ этого сигнала для последовательности предшествующих моментов времени, а также значений входного (управляющего) сигнала $u(t-1), u(t-2), \dots, u(t-N_u)$, внешнего по отношению к NARX-модели. В общем случае длина предыстории по выходам и управлениям может не совпадать, т.е. $N_y \neq N_u$.

Удобным способом реализации NARX-модели является использование многослойной сети прямого распространения мультиперсептронного типа для приближенного представления отображения $f(\cdot)$ в соотношении (3.12), а также линий задержки (TDL-элементов) для получения значений величин $\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-N_y)$ и $u(t-1), u(t-2), \dots, u(t-N_u)$. Конкретный вид нейросетевой реализации NARX-модели, которая использовалась для моделирования движения ЛА, показан на рис. 3.2. Видно, что данная NARX-модель представляет

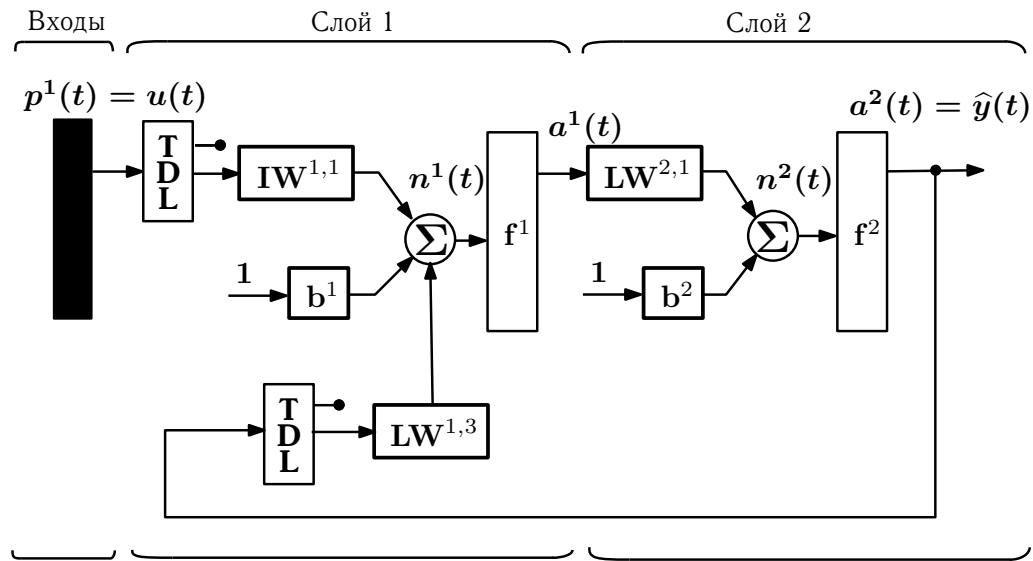


Рис. 3.2. Структурная схема нейросетевой NARX-модели объекта управления. Здесь: TDL — линия задержки; IW — матрица синаптических весов связей между входным и первым обрабатывающим слоем НС; LW — матрица синаптических весов связей между обрабатывающими слоями НС; b — набор смещений слоя НС; f — набор активационных функций слоя НС; Σ — набор сумматоров слоя НС; $n(t)$ — набор скалярных выходов сумматоров; $a(t)$ — набор скалярных выходов активационных функций; $p^1(t) = u(t)$ — входной сигнал; $\hat{y}(t)$ — выход НС-модели

собой двухслойную сеть, в ней активационные функции скрытого слоя являются нелинейными (сигмоиды), а выходного слоя — линейными.

Процесс обучения NARX-модели в рассматриваемом случае может быть построен одним из двух способов.

В первом способе (параллельная архитектура, рис. 3.1а) выход NARX-модели можно трактовать как оценку $\hat{y}(t)$ выхода моделируемой нелинейной системы. Эта оценка подается с помощью обратной связи через TDL-элемент на вход NARX-модели для предсказания следующего состояния $\hat{y}(t + 1)$ системы.

Во втором способе (последовательно-параллельная архитектура, рис. 3.1б) учитывается тот факт, что обучение нейросетевой NARX-модели осуществляется с учителем, т. е. доступна информация не только о входах модели $u(t)$, но и о значениях $y(t)$ выходов системы, отвечающих данным значениям входов. Соответственно, эти значения выходов $y(t)$ можно подавать на вход NARX-модели вместо их оценок \hat{y} , как это имело место в предыдущем варианте. У данного подхода два основных преимущества: повышается точность получаемой NARX-модели, а также появляется возможность использовать для ее обучения обычный ста-

тический метод обратного распространения ошибки, тогда как для обучения NARX-модели с чисто параллельной архитектурой требуется использовать какую-либо из разновидностей динамического метода обратного распространения ошибки.

3.2.2 Обучение нейросетевой модели движения ЛА в пакетном режиме

Обучение НС-модели в данном случае производится стандартным образом [74, 77]: проблема рассматривается как оптимизационная, целью является минимизация ошибки $e = y - \hat{y}$. Целевая функция представляет собой сумму квадратов ошибок на всей обучающей выборке

$$E(w) = \frac{1}{2} e^T(w) e(w), \quad e = [e_1, e_2, \dots, e_N]^T,$$

где $e(w) = y - \hat{y}(w)$, w — M -мерный вектор настраиваемых параметров сети, N — длина выборки.

Минимизация целевой функции $E(w)$ по вектору w производится методом Левенберга-Марквардта. Корректировка вектора w на каждом шаге оптимизации выглядит следующим образом:

$$w_{n+1} = w_n + (J^T J + \mu E)^{-1} J^T e,$$

где E — единичная матрица, $J = J(w_n)$ — матрица Якоби, т. е. $(N \times M)$ -матрица, i -я строка которой представляет собой вектор, полученный транспонированием градиента функции e_i .

Наиболее трудоемкий элемент рассматриваемого процесса — вычисление якобиана на каждом шаге. Для этого используется алгоритм обратного распространения ошибки [74], что занимает большую часть времени, затрачиваемого на обучение модели.

3.2.3 Обучение нейросетевой модели движения ЛА в режиме реального времени

В используемых НС-моделях активационными функциями являются сигмоиды. Такие глобальные активационные функции обеспечивают НС-модель хорошими обобщающими свойствами. Однако при этом корректировка любого настраиваемого параметра меняет поведение сети на всей области входных данных. Это значит, что при попытке дообучения сети новым поступившим данным пропадут свойства сети на предыдущих данных. Таким образом, чтобы принять в расчет поступающие измерения, НС-модели рассматриваемого типа надо доучивать на очень большой выборке, что неосуществимо с вычислительной точки зрения.

Для того, чтобы обойти эту проблему (использовать для обучения если не текущее измерение, то хотя бы небольшой скользящий интервал), можно использовать рекуррентный метод наименьших квадратов (РНМК), который можно рассматривать как частный случай фильтра Калмана (ФК) при оценке постоянных параметров. Однако ФК и РНМК напрямую применимы только для систем, наблюдения которых линейны по оцениваемым параметрам, тогда как уравнение наблюдения нейросети является нелинейным. Соответственно, чтобы можно было использовать ФК, следует линеаризовать уравнение наблюдения. С этой целью можно использовать, в частности, статистическую линеаризацию.

Данный подход применительно к НС-моделированию, использованный в данной работе, подробно излагается в [74], откуда видно, что опять, как и в случае пакетного обучения НС-модели, критическим по вычислительной трудоемкости элементом (особенно для динамических сетей) является вычисление якобиана J_k ошибок сети по настраиваемым параметрам.

Для получения модели с требуемой точностью в качестве обучающих данных берется последовательность значений на некотором скользящем интервале наблюдения

$$\hat{y}_k = [\hat{y}_{i-l}, \hat{y}_{i-l+1}, \dots, \hat{y}_i]^T,$$

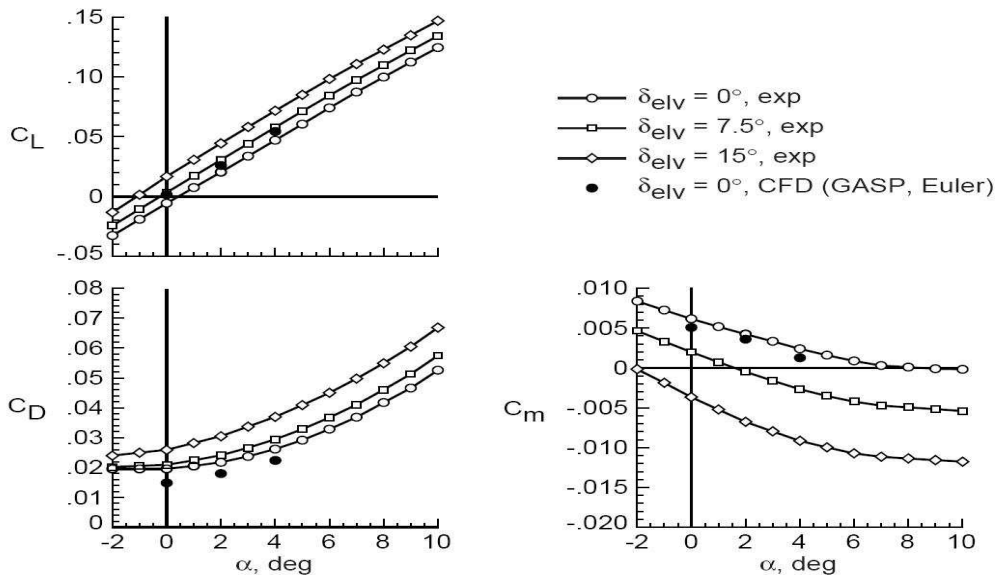
где l — длина скользящего интервала, индекс i относится к моменту времени (шагу дискретизации), а индекс k указывает номер оценки.

Для экономии времени оценка параметров производится не на каждом шаге дискретизации, а прорежена в десять раз (шаг дискретизации равен 0.01 с, а обновление параметров сети происходит через 0.1 с). Вычислительный эксперимент показывает, что такое «огрубление» вполне приемлемо, поскольку несущественно сказывается на точности модели.

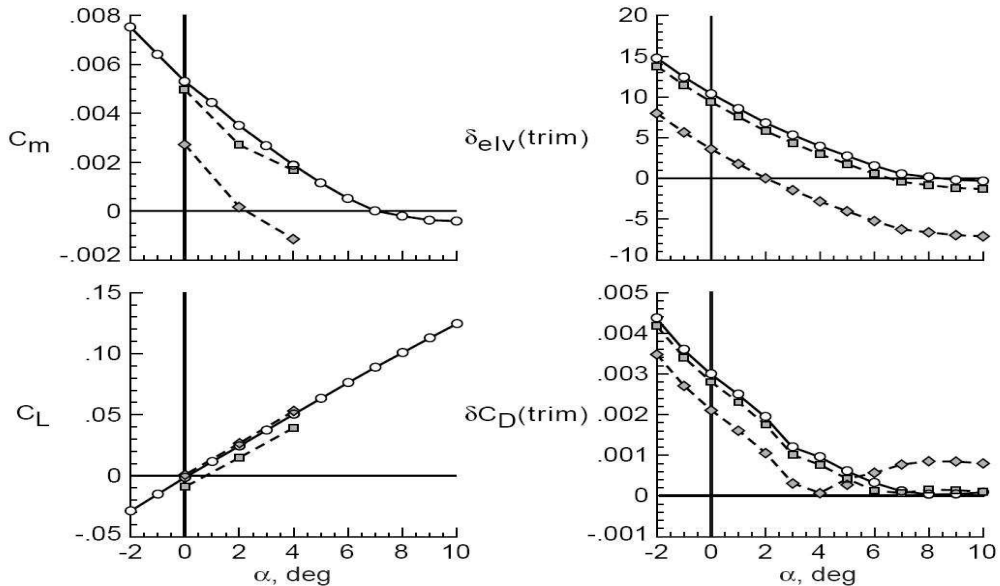
3.3 Оценка работоспособности нейросетевой модели движения ЛА на основе многослойной нейронной сети

1. Оценка работоспособности рассматриваемой НС-модели проводилось применительно к угловому продольному движению летательного аппарата, которое описывалось с помощью математической моделей, традиционных для динамики полета самолетов [39].

В этих моделях, учитывая разнохарактерность ЛА, применительно к которым осуществлялся синтез и анализ алгоритмов адаптивного управления, введена в рассмотрение взаимосвязь между углом атаки α и тягой двигателя P , характерная для ГЗЛА; применительно к самолету F-16 эта связь не учитывалась, как нехарактерная для него.



(a)



(b)

Рис. 3.3. Аэродинамические данные для модели движения ГЗЛА Х-43А: **(а)** значения аэродинамических коэффициентов для продольного движения при числе Маха $M = 6$ по результатам вычислительного эксперимента и испытаний в аэродинамической трубе; **(б)** значения аэродинамических коэффициентов для продольного движения при числе Маха $M = 6$ для случаев, когда воздухозаборник закрыт (○), воздухозаборник открыт при неработающем двигателе (□), воздухозаборник открыт при работающем двигателе (◇). Здесь обозначено: C_D , C_L , C_m — коэффициенты C_x , C_y , m_z , соответственно; δ_{elv} — угол симметричного отклонения элевонов; α — угол атаки, град.; exp — эксперимент в АДТ; CFD — вычислительный эксперимент; trim — балансирующее значение

Первая из рассматриваемых моделей («одноканальная») использует неявную связь между значениями этих величин. Она дается через значение коэффициента $m_z(\alpha, P)$ (см. рис. 3.3), дополнительные эффекты от влияния тяги на угол атаки и угла атаки на тягу в этой модели не учитываются, управление тягой также не вводится, используется управление по единственному каналу — изменением значения φ_{act} , командного сигнала для привода элевонов (руля высоты). Данная модель имеет вид:

$$\begin{aligned}\dot{\alpha} &= \omega_z - \frac{qS}{mV} C_{y_a}(\alpha, \omega_z, \varphi) + \frac{g}{V} \cos \vartheta, \\ \dot{\omega}_z &= \frac{qSb_A}{J_{zz}} m_z(\alpha, \omega_z, \varphi), \\ T^2 \ddot{\varphi} &= -2T\zeta \dot{\varphi} - \varphi + \varphi_{act},\end{aligned}\tag{3.13}$$

где α — угол атаки, град; ϑ — угол тангажа, град; ω_z — угловая скорость тангажа, град/с; φ — угол отклонения управляемого стабилизатора, град; C_{y_a} — коэффициент подъемной силы; m_z — коэффициент момента тангажа; m — масса самолета, кг; V — воздушная скорость, м/с; $q = \rho V^2 / 2$ — скоростной напор; ρ — плотность воздуха, кг/м³; g — ускорение силы тяжести, м/с²; S — площадь крыла, м²; b_A — средняя аэродинамическая хорда крыла, м; J_{zz} — момент инерции самолета относительно боковой оси, кг·м²; безразмерные коэффициенты C_{y_a} и m_z являются нелинейными функциями своих аргументов; T, ζ — постоянная времени и коэффициент относительного демпфирования привода, φ_{act} — командный сигнал на привод (ограничивается $\pm 25^\circ$). В модели (3.13) величины $\alpha, \omega_z, \varphi$ и $\dot{\varphi}$ — это состояния объекта управления, величина φ_{act} — управление.

2. Вторая из использовавшихся моделей движения ЛА («двухканальная»), которая использовалась только для ГЗЛА X-43 и NASP, представляет собой вариант модели (3.13), расширенный за счет включения в нее канала управления тягой и явной взаимосвязи между углом атаки и тягой двигателя, помимо неявной, указанной выше. Соответственно, наряду с командным сигналом φ_{act} , в этой модели введено также и управление по тяге двигателя через командный сигнал δ_{rud} . Данная модель имеет вид:

$$\begin{aligned}\dot{\alpha} &= \omega_z - \frac{qS}{mV} C_{y_a}(\alpha, \varphi) + \frac{P_{cur} \sin \alpha}{mV} + \frac{g}{V} \cos \vartheta, \\ \dot{\omega}_z &= \frac{qSb_A}{J_{zz}} m_z(\alpha, \omega_z, \varphi) + \frac{P_{cur} h}{J_{zz}}, \\ T^2 \ddot{\varphi} &= -2T\zeta \dot{\varphi} - \varphi + \varphi_{act}, \\ \dot{P}_{cur} &= \omega_{eng} (P_{ref}(\delta_{rud}) - P_{cur}), \\ n_{x_a} &= -\frac{qS}{mg} C_{x_a}(\alpha, \varphi) + \frac{P_{cur} \cos(\alpha)}{mg}.\end{aligned}\tag{3.14}$$

Здесь $P_{ref} = P_{ref}(\delta_{rud})$ — заданный уровень тяги (линейная функция), P_{cur} — текущий уровень тяги, ω_{eng} — частота апериодического звена, которым описывается динамика двигателя (здесь было принято $\omega_{eng} = 1$). Плечо тяги принято равным $h = 0.5$ м, оно вычисляется относительно центра масс ЛА в вертикальной плоскости, поэтому изменение δ_{rud} вызывает изменение угла атаки. В модели (3.14) величины α , ω_z , φ , $\dot{\varphi}$ и P_{cur} — это состояния объекта управления, величины φ_{act} и δ_{rud} — управления.

3. Для оценки свойств рассматриваемой НС-модели и ее пригодности для моделирования движения ЛА была проведена серия вычислительных экспериментов. Чтобы продемонстрировать работоспособность адаптивного управления в различных условиях, в качестве примеров объекта управления были выбраны летательные аппараты существенно отличающихся друг от друга классов: маневренный самолет F-16 [44], тяжелый гиперзвуковой самолет (один из вариантов [47], рассматривавшихся NASA в рамках программы NASP (National AeroSpace Plane), ориентированной на создание одноступенчатого воздушно-космического самолета с горизонтальным стартом, выведением полезной нагрузки на орбиту искусственного спутника Земли, горизонтальной посадкой), экспериментальный гиперзвуковой летательный аппарат (ГЗЛА) X-43 [45, 46], а также БПЛА «003» и X-04 микро- и мини-размерности, соответственно [22]. Значения соответствующих параметров и характеристик для этих ЛА, требуемые для моделирования в качестве исходных данных, были взяты из работ [44] (самолет F-16), [46] (ГЗЛА X-43), [47] (ГЗЛА NASP) и [22] (БПЛА).

4. Для перечисленных летательных аппаратов был осуществлен вычислительный эксперимент, позволяющий судить о свойствах рассматриваемого класса НС-моделей. Примеры результатов проведенных экспериментов показаны на рис. 3.4, 3.5 для X-43, F-16, соответственно, а также 3.6 и 3.7 для БПЛА.

На рис. 3.4а, 3.5а, 3.6а и 3.7а показаны примеры обучающих выборок, использованных для обучения НС-моделей. Из них видно, что для формирования каждой из выборок реализуется очень активная работа органом управления продольным движением (элевоны для X-43 и БПЛА, управляемый стабилизатор для F-16), выражающаяся в частом изменении значения командного сигнала ϕ_{act} привода органа управления при существенных перепадах между соседними значениями (этот командный сигнал формировался случайным образом). Цель применения такого способа формирования обучающего набора состоит в том, чтобы обеспечить возможно большее разнообразие состояний моделируемой системы (чтобы покрыть по-возможности равномерно и плотно все пространство состояний системы), а также воз-

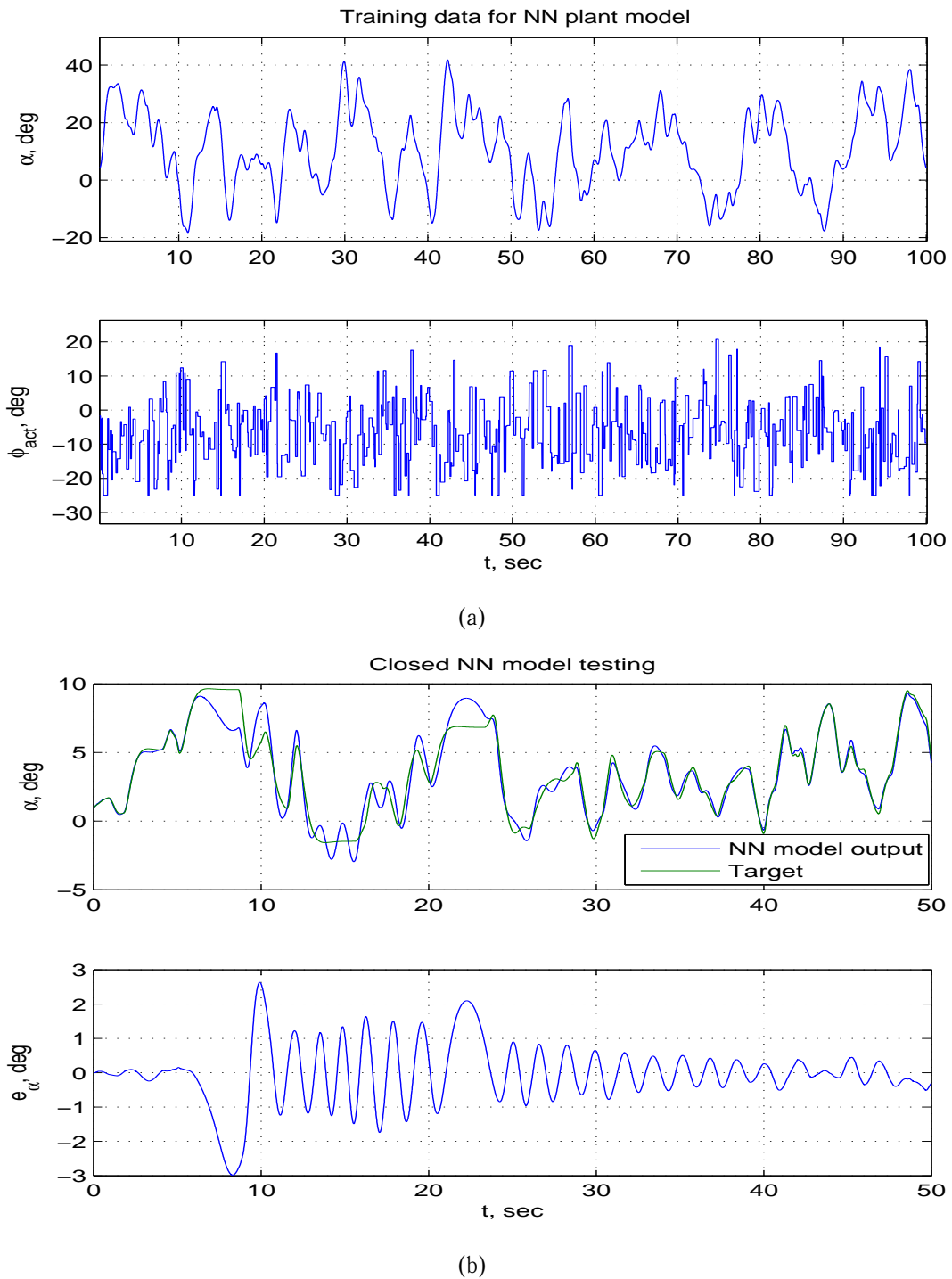


Рис. 3.4. Формирование нейросетевой модели для ГЗЛА Х-43А, режим полета с числом Маха $M = 6$: **(a)** обучающая выборка для НС-модели; **(b)** проверка работоспособности замкнутой НС-модели. Здесь α — угол атаки, град.; e_α — ошибка (расхождение углов атаки для объекта и НС-модели), град.; ϕ_{act} — командный сигнал привода органа управления; t — время, с; Target — выход объекта; NN model output — выход НС-модели

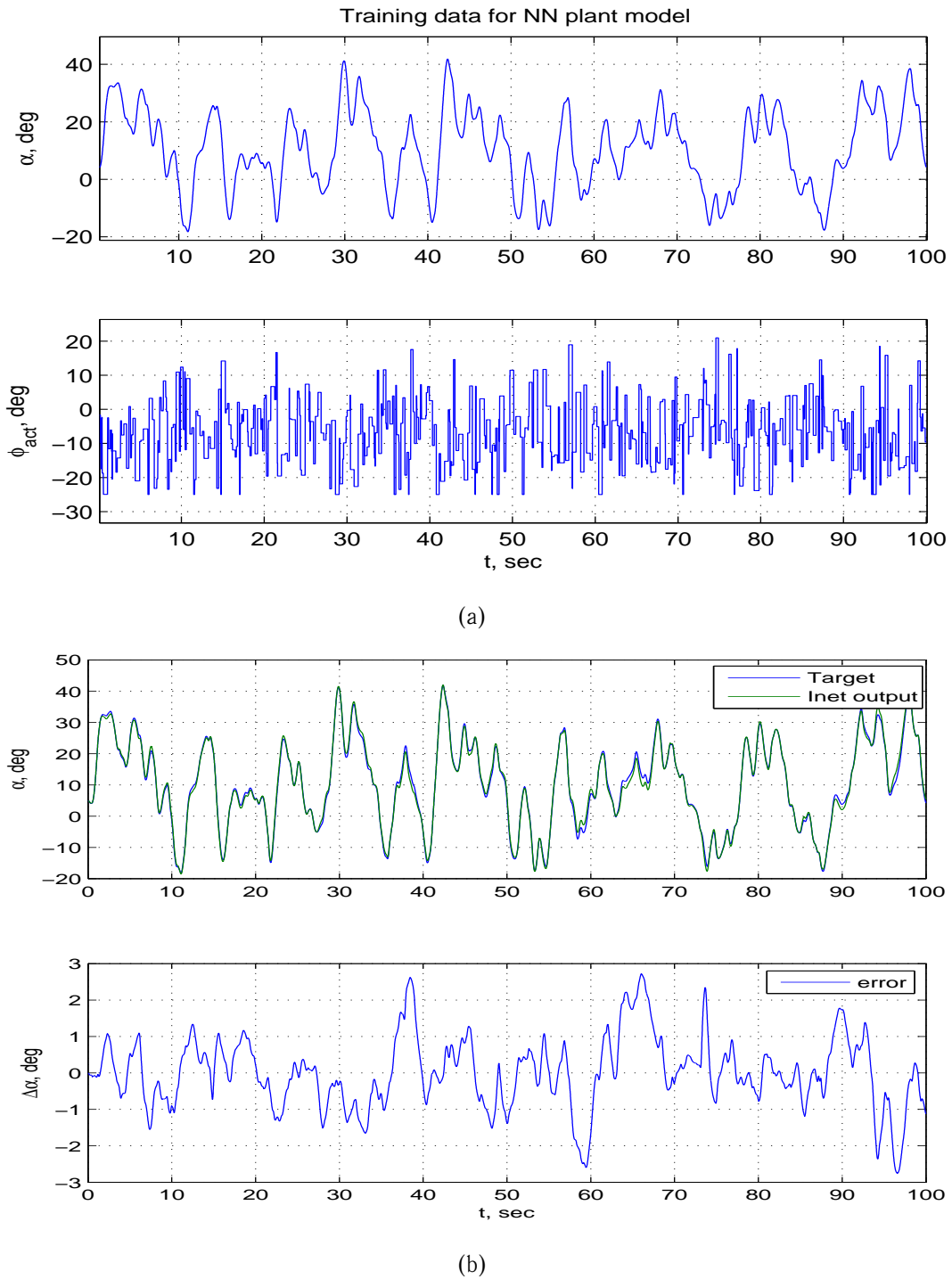


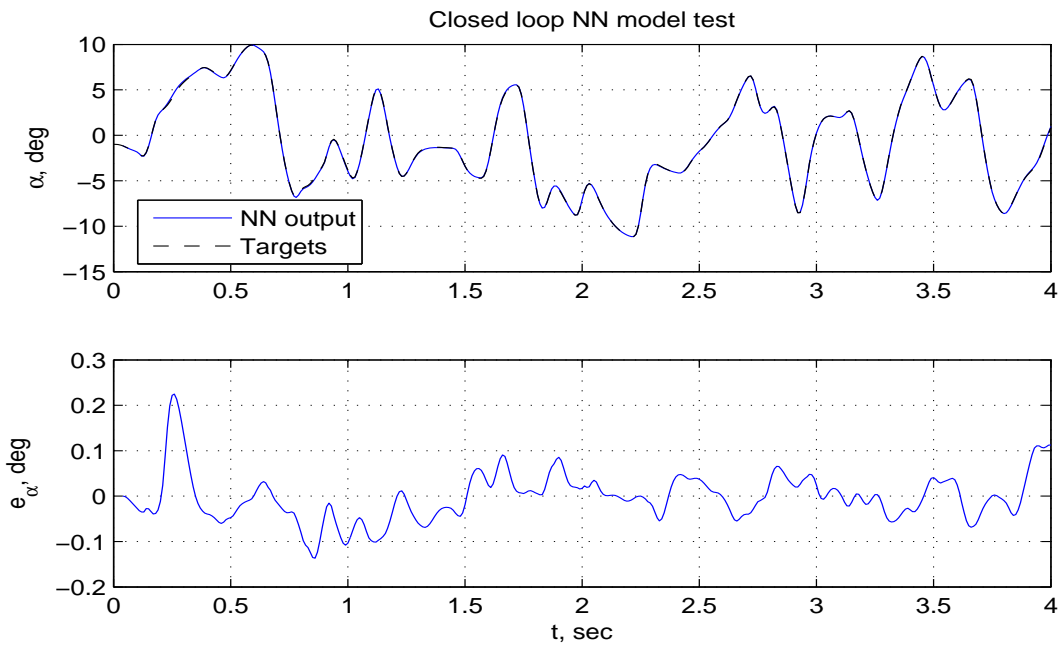
Рис. 3.5. Формирование нейросетевой модели объекта управления (самолет F-16, режим полета с индикаторной скоростью $V_i = 500$ км/ч): **(a)** обучающая выборка для НС-модели; **(b)** проверка работоспособности замкнутой НС-модели. Здесь α — угол атаки, град.; $\Delta\alpha$ — ошибка (расхождение углов атаки для объекта и НС-модели), град.; ϕ_{act} — командный сигнал привода органа управления; t — время, с; Target — выход объекта; Inet output — выход НС-модели; Error — ошибка отслеживания

ТАБЛИЦА 3.1. Основные данные, использованные при проведении компьютерных экспериментов по синтезу нейросетевых моделей движения БПЛА

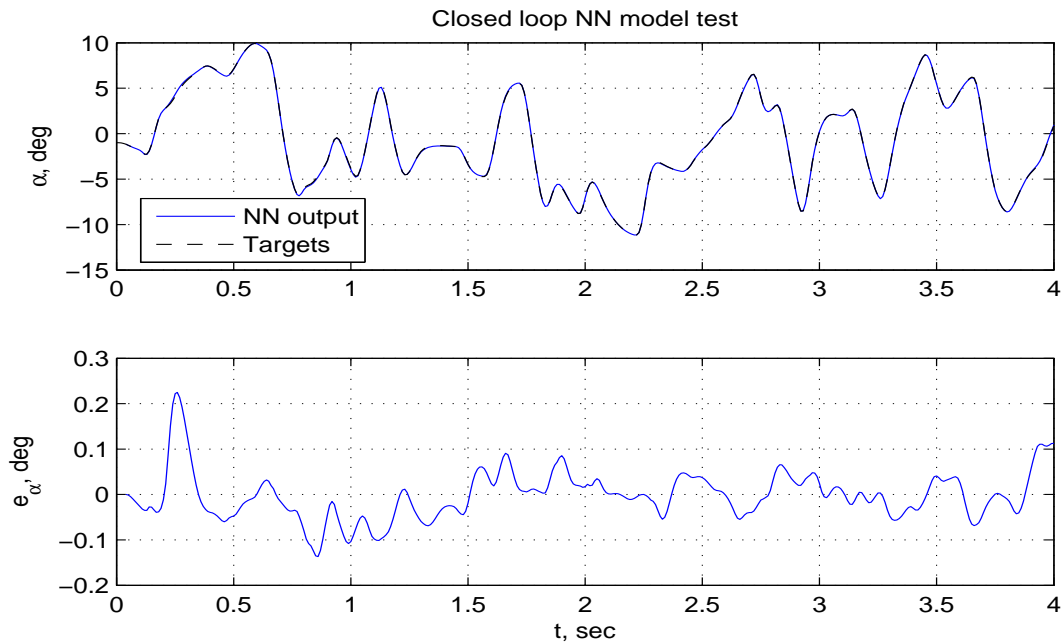
Наименование характеристики	Обозначение	Летательный аппарат	
		Микро-БПЛА «003»	Мини-БПЛА X-04
Полетная масса, кг	m	0.15	4.162
Момент инерции, кг·м ²	I_z	0.00157	0.629
Площадь крыла, м ²	S	0.135	0.45
САХ крыла, м	b_A	0.212	0.225
Угол атаки нулевой подъемной силы, рад	α_0	-0.097	-0.097
Коэффициент момента тангажа при $C_{ya} = 0$	m_{z0}	-0.07	0
Относительная координата центра масс	\bar{x}_T	0.15	0.3
Относительная координата фокуса	\bar{x}_F	0.23	0.42
Производная	C_{ya}^α	3.164	4.276
Производная	$m_z^{\delta_a}$	-1.218	-1.068
Производная	$m_z^{\sigma_z}$	-1.587	-8.3

можно большее разнообразие перепадов соседних по времени состояний (чтобы максимально достоверно отразить в НС-модели динамику моделируемой системы).

Поскольку целью управления в рассматриваемой задаче является максимально точное отслеживание требуемых значений угла атаки, то, соответственно, оценка точности формируемой модели осуществляется сопоставлением поведения по этой величине для исходного объекта управления, описываемого системой дифференциальных уравнений (3.13) или (3.14) и

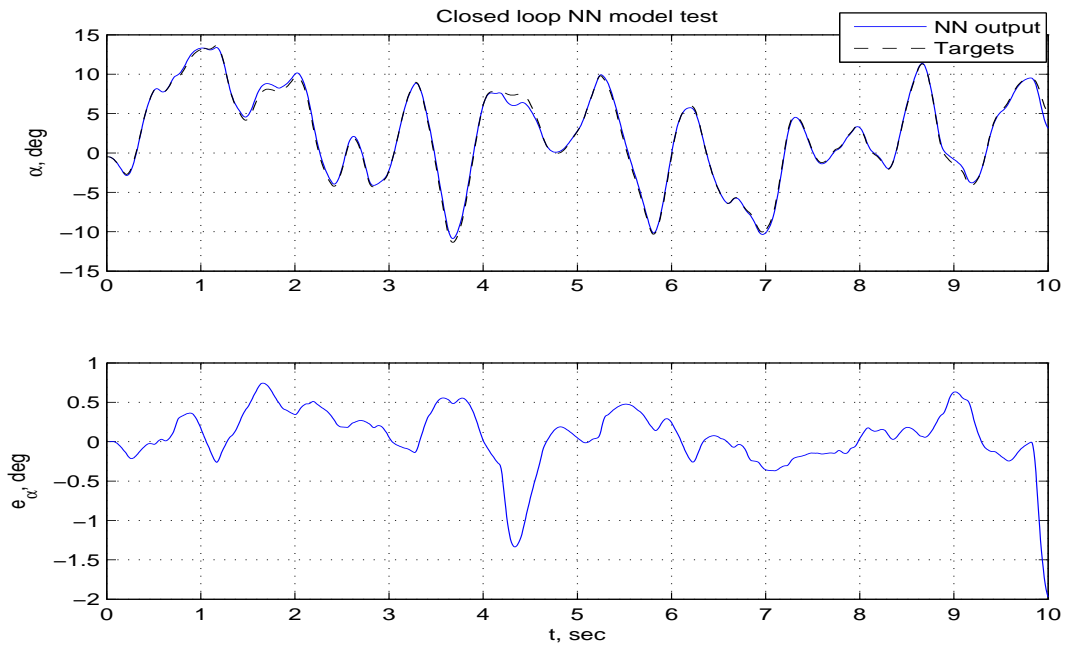


(a)

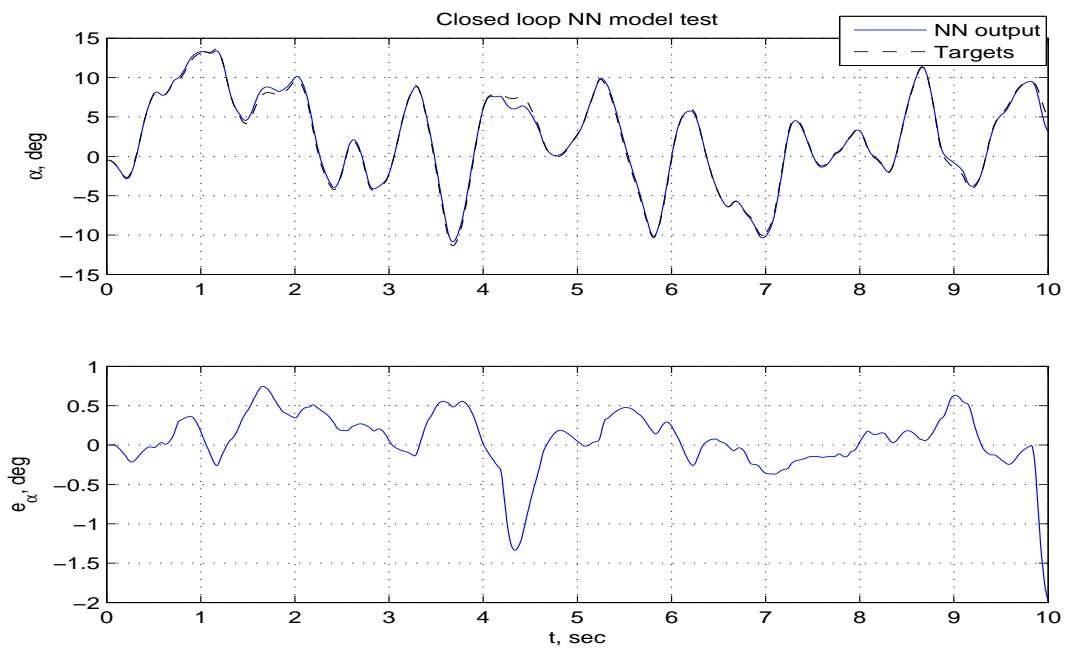


(b)

Рис. 3.6. Формирование нейросетевой модели микро-БПЛА «003» как объекта управления, режим полета с индикаторной скоростью км/ч: **(a)** — обучающая выборка для НС-модели; **(b)** — проверка работоспособности замкнутой НС-модели. Здесь α — угол атаки, град.; e_{α} — ошибка (расхождение углов атаки для объекта и НС-модели), град.; ϕ_{act} — командный сигнал привода органа управления, град.; t — время, с; Targets — выход объекта; NN output — выход НС-модели



(a)



(b)

Рис. 3.7. Формирование нейросетевой модели мини-БПЛА X-04 как объекта управления, режим полета с индикаторной скоростью км/ч: **(a)** — обучающая выборка для НС-модели; **(b)** — проверка работоспособности замкнутой НС-модели. Здесь α — угол атаки, град.; e_α — ошибка (расхождение углов атаки для объекта и НС-модели), град.; ϕ_{act} — командный сигнал привода органа управления, град.; t — время, с; Targets — выход объекта; NN output — выход НС-модели

сформированной НС-модели. Точность модели оценивается ошибкой e_α , вычисляемой как разность углов атаки для объекта управления и НС-модели в один и тот же момент времени.

Из приводимых примеров видно, что предлагаемый подход дает возможность строить довольно точные НС-модели (значение e_α в пределах $\pm(0.5 \div 0.7$ град.), однако в ряде случаев точность ухудшается, что впоследствии является причиной получения неудовлетворительных адаптационных свойств синтезированных нейроконтроллеров. Пути преодоления этих затруднений рассматриваются ниже, в разд. 5 и 6.

4 Нейросетевая реализация адаптивных динамических систем применительно к управлению полетом ЛА

Одним из важнейших классов динамических систем являются летательные аппараты различных видов. Управление движением современных и перспективных ЛА приходится обеспечивать, как уже отмечалось в разд. 1.1.2, в условиях значительных и разнообразных неопределенностей в значениях их параметров и характеристик, режимов полета, воздействий внешней среды. Кроме того, в ходе полета могут возникать разнообразные нештатные ситуации, в частности, отказы оборудования и повреждения конструкции, последствия которых в значительной части случаев можно парировать за счет реконфигурации системы управления ЛА.

Наличие значительных и разнообразных неопределенностей является одним из наиболее серьезных факторов, осложняющих решение всех трех задач (анализ, синтез, идентификация) для динамических систем, одним из важнейших видов которых являются летательные аппараты. Проблема здесь состоит в том, что текущая ситуация, в которой оказывается ДС, может меняться резко, значительно, а также, из-за наличия неопределенностей, непредсказуемо. Чтобы обеспечить нормальное функционирование системы в таких условиях, она должна уметь оперативно приспосабливаться к изменениям ситуации, т. е. она должна быть адаптивной. Как уже отмечалось в разд. 1.7, под адаптивной понимается такая система, которая может оперативно подстраиваться к меняющейся ситуации за счет изменения каких-то из своих элементов. При этом принимается, что такими элементами, как правило, являются законы управления, реализуемые ДС, а также модель ДС как объекта управления. Большей частью, изменения в этих элементах могут затрагивать как значения параметров, так и структуру законов управления и/или моделей.

Анализ возможностей алгоритмов адаптивного управления применительно к движению ЛА осуществляется в следующих подразделах применительно к таким основным видам адаптивных систем (АС), как АС с прогнозирующей моделью, АС с эталонной моделью, АС с инверсной моделью и АС на основе обратной задачи динамики.

Все эти схемы адаптивного управления требуют наличия модели объекта управления. Как было показано в разд. 1.7.4 и 1.7.5, принятая в качестве основной нейросетевая реализация данных схем, обладающая высокой вычислительной эффективностью, требует нейросетевой реализации и модели объекта управления. Вопросы формирования нейросетевой модели объекта управления, требуемой в проводимых исследованиях, подробно рассматривались в

разд. 2.

Схемы адаптивного управления часто основываются на использовании некоторой эталонной модели (ЭМ), задающей требуемый характер поведения рассматриваемой системы; именно этот вариант реализован в проведенных исследованиях. В системах адаптивного управления, реализующих такую схему, параметры регулятора $\theta_c(t)$ корректируются согласно алгоритму, реализуемому законом адаптации. Это вычисление основывается непосредственно на значении ошибки слежения $\varepsilon(t) = y_m(t) - y(t)$, где $y_m(t)$ — выход эталонной модели, $y(t)$ — выход объекта управления. Соображения, на основе которых выбиралась эталонная модель, требуемая для выполнения исследований, излагаемых в данной главе, приводятся в разд. 4.4.2.1.

Опыт использования упомянутых схем адаптивного управления показал, что выбор параметров ЭМ существенно влияет на характер получаемых результатов: некорректный выбор этих параметров может привести к тому, что система управления становится неработоспособной. В то же время, при адекватных значениях параметров ЭМ удастся получить, как это будет показано ниже, систему управления, которая хорошо справляется с поставленными перед ней задачами. Результаты анализа влияния параметров ЭМ на работоспособность синтезируемой системы управления представлены в разд. 4.4.2.1.

Изучаемые схемы адаптивного управления существенным образом основаны на использовании НС-модели объекта управления в качестве источника информации о поведении этого объекта. Поскольку, в силу приближенного характера НС-модели, реальные значения величин, описывающих движение объекта, неизбежно отличаются от получаемых в качестве выходов НС-модели, возникает ошибка, ухудшающая качество управления. Один из возможных подходов к компенсации этой ошибки предложен в разд. 4.1.3. Он состоит в том, чтобы неточность НС-модели трактовать как возмущающее воздействие на систему, приводящее к отклонению траектории реального объекта от эталонной траектории, и парировать это воздействие путем введения в систему компенсирующего контура.

4.1 Адаптивное управление с эталонной моделью

4.1.1 Общая схема адаптивного управления с эталонной моделью

В задаче управления с эталонной моделью (MRAC — Model Reference Adaptive Control) регулятор реализуется в нейросетевом виде (нейроконтроллер) с использованием НС типа

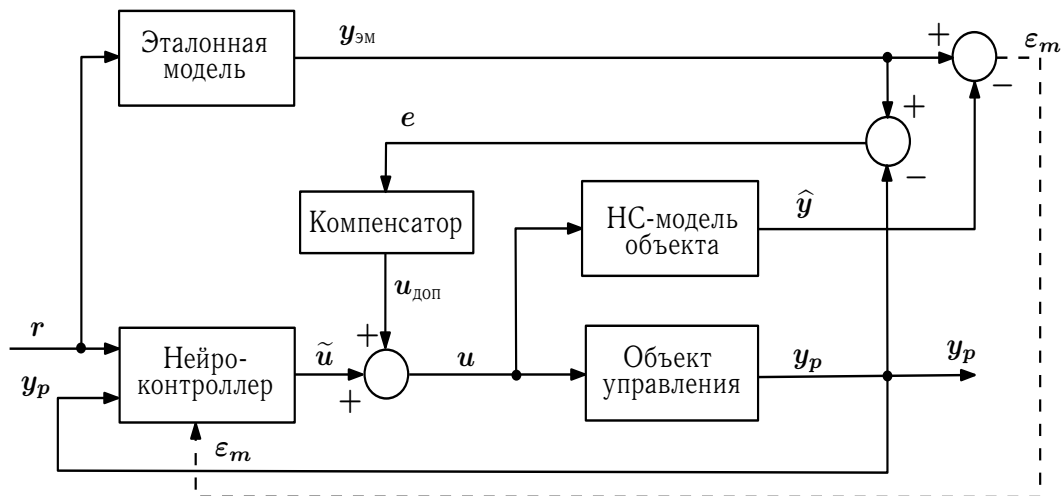


Рис. 4.1. Схема нейросетевого адаптивного управления с эталонной моделью. Здесь обозначено: \tilde{u} — управление на выходе нейроконтроллера, $u_{\text{доп}}$ — добавочное управление от компенсатора, u — результирующее управление, y_p — выход объекта управления, \hat{y} — выход нейросетевой модели объекта управления; $y_{\text{эм}}$ — выход эталонной модели; ε — расхождение между выходами объекта управления и эталонной модели; ε_m — расхождение между выходами объекта управления и НС-модели; r — задающее воздействие

NARX. Обучение его осуществляется таким образом, чтобы выход системы под его управлением максимально точно соответствовал выходу эталонной модели. Для реализации процесса обучения нейроконтроллера требуется нейросетевая модель объекта.

В нейросетевой реализации схемы адаптивного управления с эталонной моделью (рис. 4.1) используются два нейросетевых модуля: управляющая сеть (нейроконтроллер) и модель объекта управления (НС-модель). Вначале решается задача идентификации для рассматриваемого объекта управления, затем полученная НС-модель используется для обучения нейроконтроллера, который должен обеспечить максимально точное отслеживание выхода эталонной модели.

Нейроконтроллер представляет собой двухслойную сеть, на входы которой через линии задержки (TDL-элементы) подается задающий входной сигнал $r(t)$ и выход объекта управления $y_p(t)$, а также, в ряде случаев, и выход нейроконтроллера $\tilde{u}(t)$ (эта связь на схеме не показана).

НС-модель объекта управления, структура которой приведена на рис. 4.2, в качестве входов получает, также через линии задержки, управляющий сигнал от нейроконтроллера, а также выход объекта управления.

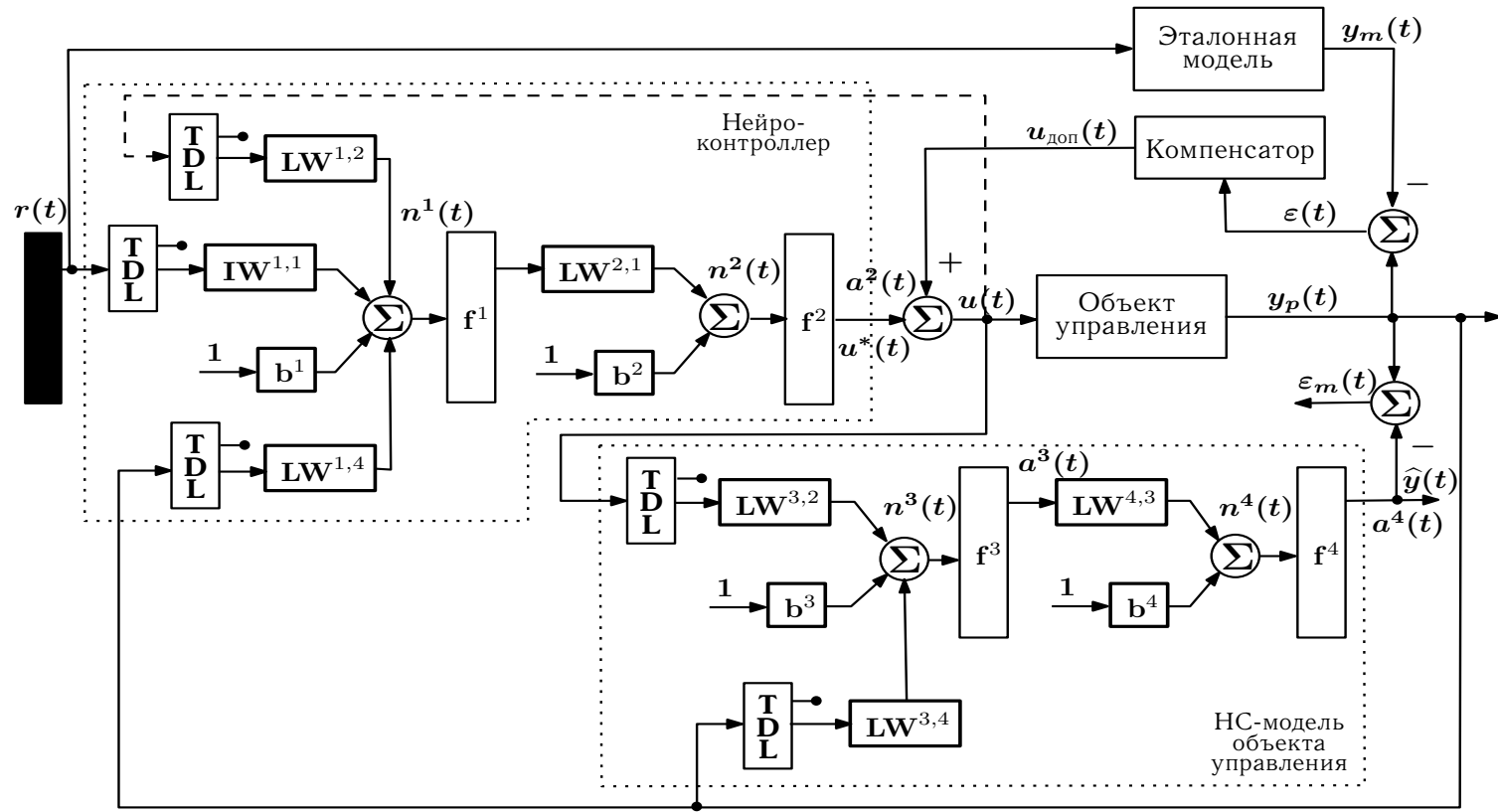


Рис. 4.2. Структурная схема нейросетевой реализации адаптивного управления с эталонной моделью: TDL — линия задержки; IW — матрица синаптических весов связей между входным и первым обрабатывающим слоем НС; LW — матрица синаптических весов связей между обрабатывающими слоями НС; b — набор смещений слоя НС; f — набор активационных функций слоя НС; Σ — набор сумматоров слоя НС; $n(t)$ — набор скалярных выходов сумматоров; $a(t)$ — набор скалярных выходов активационных функций; $r(t)$ — задающий сигнал; $y_p(t)$ — выход объекта управления; $\hat{y}(t)$ — выход НС-модели; $y_m(t)$ — выход эталонной модели; $u_*(t)$ — управление, вырабатываемое нейроконтроллером; $u_{\text{доп}}(t)$ — добавочное управление от компенсатора; $u(t)$ — управление, поступающее на вход объекта; $\varepsilon(t) = y_p(t) - y_m(t)$ — расхождение между выходами объекта и эталонной модели

4.1.2 Синтез нейроконтроллера для адаптивного управления с эталонной моделью

1. Уравнение нейроконтроллера (НК) имеет вид (статический контроллер):

$$u_k = f(r_k, r_{k-1}, \dots, r_{k-d}, y_k, y_{k-1}, \dots, y_{k-d}), \quad (4.1)$$

где y — выход объекта управления, r — задающий сигнал.

По аналогии со схемой управления с эталонной моделью для линейных систем уравнение нейроконтроллера должно выглядеть несколько иначе:

$$u_k = f(r_k, u_{k-1}, \dots, u_{k-d}, y_k, y_{k-1}, \dots, y_{k-d}), \quad (4.2)$$

Однако проведенное моделирование показывает, что работают эти варианты примерно одинаково, но первый обучается несколько быстрее. Поэтому в качестве основного был принят статический вариант (4.1) нейроконтроллера.

2. Использование схемы адаптивного управления с ЭМ требует, очевидно, определить тем или иным способом соответствующую эталонную модель, отражающую представления разработчика системы о том, что такое «хорошее» поведение данной системы, к которому нейроконтроллер должен стремиться приблизить поведение объекта управления.

Эталонная модель может быть определена различными способами. В данной работе ЭМ строится на основе колебательного звена с достаточно высоким демпфированием в сочетании с апериодическим звеном.

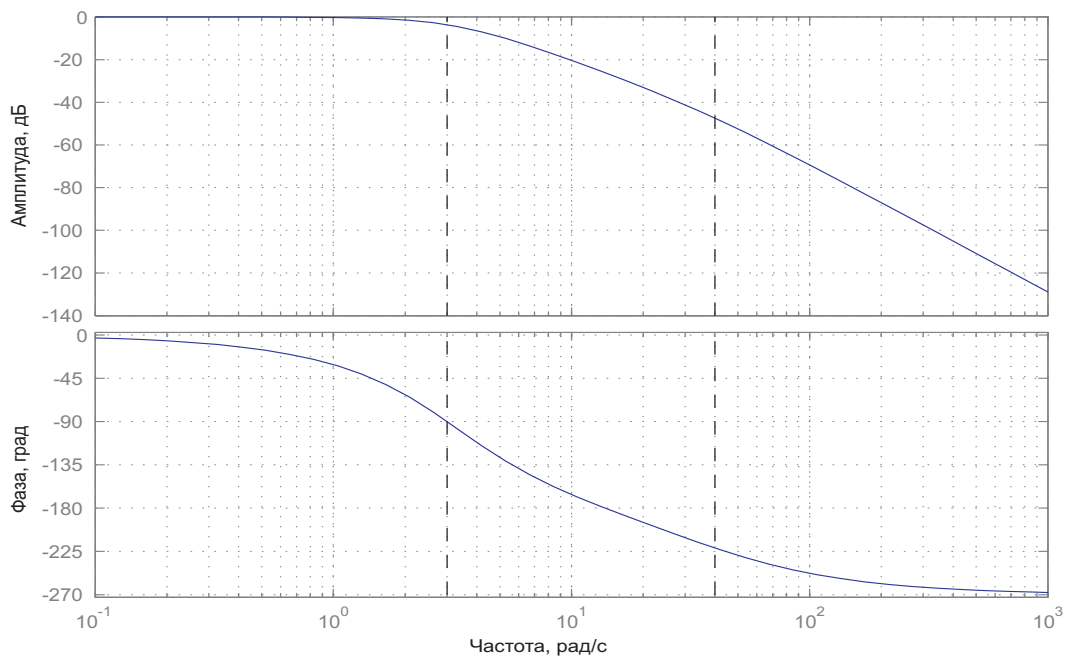
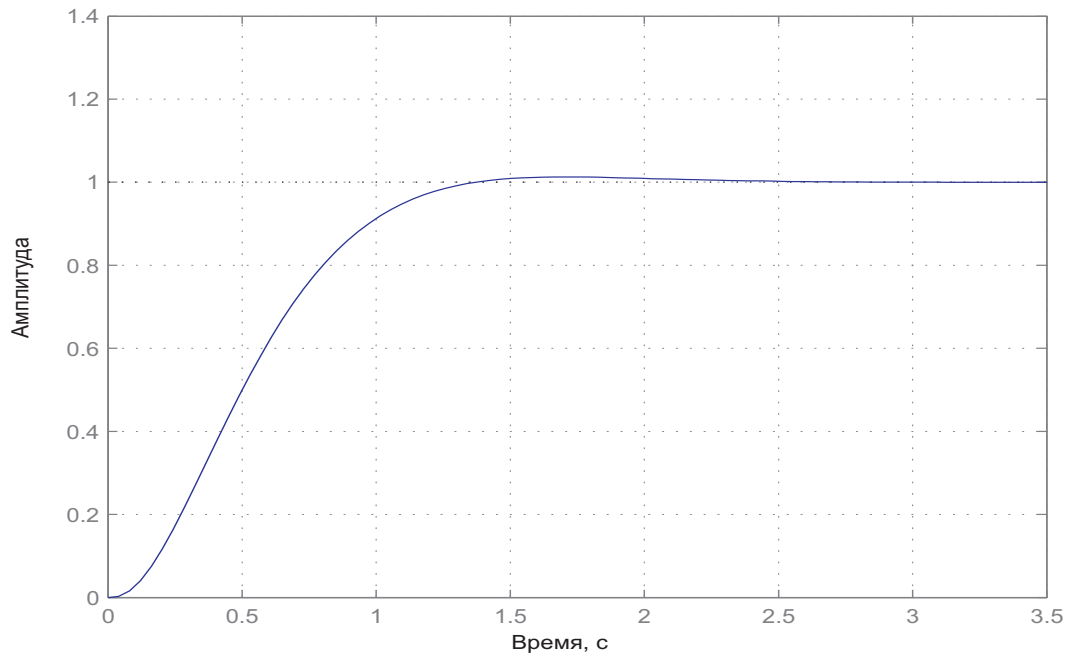
Для случая, когда движение ЛА описывается уравнениями (3.13), ЭМ задается следующим образом:

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= x_3, \\ \dot{x}_3 &= \omega_{act}(-x_3 - 2\omega_{эм}\zeta_{эм}x_2 + \omega_{эм}^2(r - x_1)). \end{aligned} \quad (4.3)$$

Здесь $\omega_{act} = 40$, $\omega_{эм} = 3$, $\zeta_{эм} = 0.8$, вектор фазовых переменных $x = [\alpha_{эм}, \dot{\alpha}_{эм}, \varphi_{act}]$.

Еще один рассмотренный вариант ЭМ, близкий к (4.3), также представляет собой линейную систему третьего порядка, которая задается передаточной функцией вида:

$$W_\alpha = \frac{\omega_{эм}^2}{((1/\omega_{act})p + 1)(p^2 + 2\omega_{эм}\zeta_{эм}p + \omega_{эм}^2)}. \quad (4.4)$$



(b)

Рис. 4.3. Характеристики эталонной модели вида (4.4): **(a)** переходная характеристика ЭМ; **(b)** частотные характеристики ЭМ

Представление о желаемом характере поведения объекта управления, задаваемом ЭМ (4.4), дает рис. 4.3, на котором приводятся характеристики данной ЭМ во временной и в частотной области. Поведение ЭМ вида (4.3), как показывает вычислительный эксперимент, очень близко к поведению ЭМ (4.4).

Если движение ЛА описывается уравнениями (3.14), тогда в дополнение к системе (4.3), описывающей канал угла атаки, добавляется еще ЭМ для канала тангенциальной (скоростной) перегрузки:

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -2\omega_{\text{ЭМ}}\zeta_{\text{ЭМ}}x_2 + \omega_{\text{ЭМ}}^2(r - x_1) \end{aligned} \quad (4.5)$$

или, в виде передаточной функции:

$$W_{n_x} = \frac{\omega_{\text{ЭМ}}^2}{(p^2 + 2\omega_{\text{ЭМ}}\zeta_{\text{ЭМ}}p + \omega_{\text{ЭМ}}^2)}. \quad (4.6)$$

В этой модели $\omega_{\text{ЭМ}} = 1$, $\zeta_{\text{ЭМ}} = 0.9$, вектор фазовых переменных $x = [n_{x_{a\text{ЭМ}}}, \dot{n}_{x_{a\text{ЭМ}}}]$, r — задающий сигнал.

3. Цель настройки нейроконтроллера — минимизация ошибки $y_{\text{ЭМ}} - \hat{y}$, т. е. приближение поведения контроллера с моделью объекта к поведению эталонной модели. При хорошей НС-модели это означает и минимизацию до определенного уровня «настоящей» ошибки $y_{\text{ЭМ}} - y$ (нейроконтроллер с реальным объектом).

Несмотря на то, что нейроконтроллер является статическим, он работает в составе динамической системы, поэтому настраивать его нужно как объединенную рекуррентную сеть. Эта настраиваемая сеть состоит из двух подсетей (самого нейроконтроллера и замкнутой модели объекта), замкнутых внешней обратной связью. В процессе настройки параметры подсети модели не меняются, т. е. НС-модель служит только для замыкания внешней обратной связи и представления всей системы в нейросетевом виде (чтобы оценить чувствительность выходов объекта управления к параметрам нейроконтроллера).

В пакетном режиме такую сеть можно обучать с помощью того же самого метода Левенберга-Марквардта (но с динамическими производными, для вычисления якобиана при этом применять либо метод обратного распространения ошибки с FR-модификацией, либо метод обратного распространения ошибки во времени). Рекуррентность доставляет дополнительные трудности в процессе обучения НК: чем больше выборка (чтобы пройти по всей области входных данных, она должна быть достаточно большой), тем больше шансов, что процесс

обучения застрянет в каком-то из локальных минимумов. Причем такие шансы с увеличением длины выборки растут с катастрофической скоростью. Из-за этого приходится делить всю выборку на сегменты.

Чтобы настройка параметров происходила однозначным образом, требуется чтобы замкнутая сеть с контроллером стартовала с эталонной траектории на каждом сегменте (так как на начальные условия нейроконтроллер повлиять не может).

При этом приходится учитывать следующие факторы:

1. На маленьких сегментах (меньше 500–1000 точек) учиться нельзя, поскольку из-за маленького диапазона входных данных сеть обучается только этому конкретному сегменту, забывая про все остальные.
2. На больших сегментах всегда попадаете плохой локальный минимум.
3. На средних по длине сегментах происходит то же самое, но с помощью ротации сегментов эту проблему можно до некоторой степени обойти.

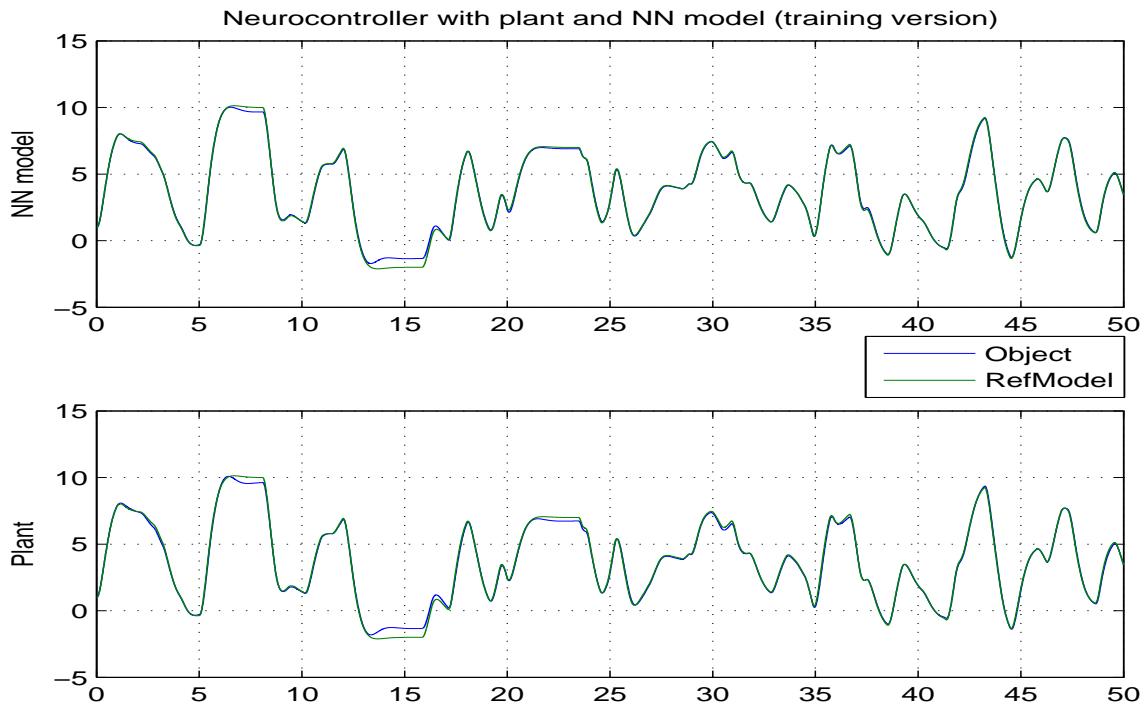
По указанным причинам приходится использовать сегменты средней длины, на каждом делать по 3–7 эпох, проходить по всему кругу несколько раз, а затем укрупнять сегменты, чтобы улучшить качество обучения. В итоге процесс обучения НК становится весьма трудоемким (до нескольких часов, в зависимости от рассматриваемого варианта).

С учетом сказанного выше, для обучения нейроконтроллера в пакетном режиме (т. е. для его преднастройки) оказывается выгоднее использовать последовательный режим обучения, упомянутый в предыдущем разделе, разница только в том, что требуется использовать динамическое обратное распространение для вычисления якобиана).

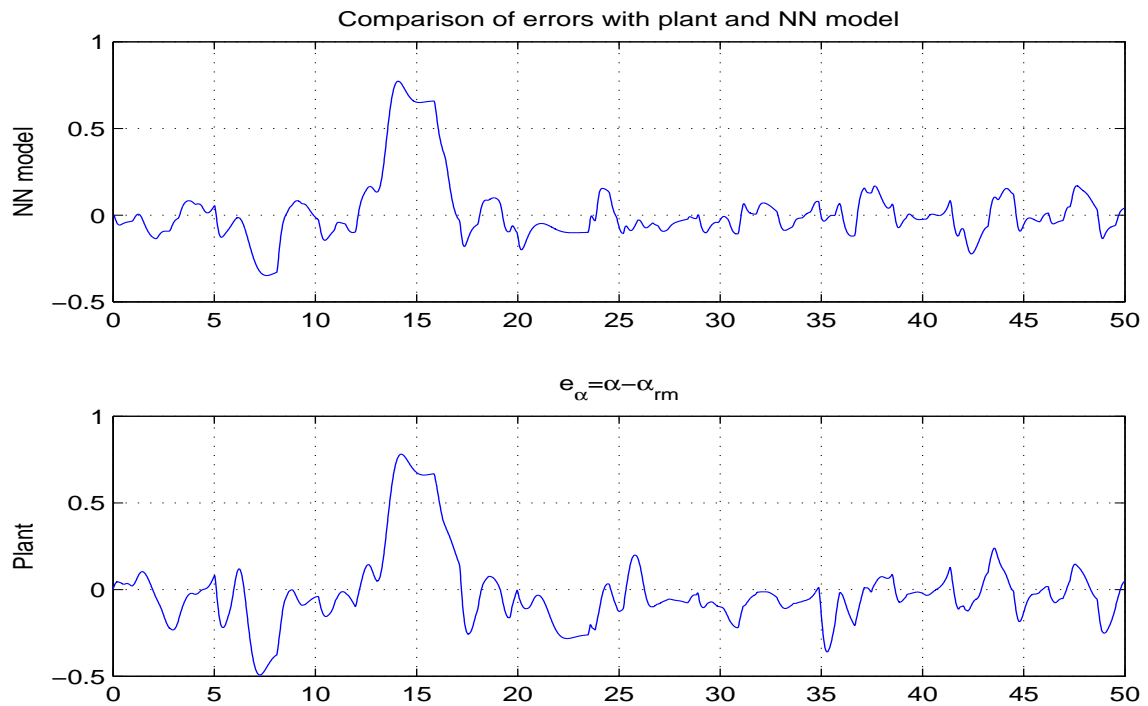
Фильтр Калмана в этом случае выполняет роль «сшивателя» отдельных сегментов в один массив данных. Причем сами сегменты можно выбрать маленькими (30–100 точек, что существенно экономит вычислительное время), главное, чтобы на этом интервале проявлялась динамика объекта управления. И хотя потенциально последовательные методы обладают меньшей точностью, здесь оказывается важнее обойти проблему локальных минимумов и сократить время обучения.

4. Таким образом порядок настройки нейроконтроллера будет следующим:

1. Установка начальных условий на эталонной траектории. Обычно под начальные условия просто отводятся первые несколько точек сегмента.



(a)



(b)

Рис. 4.4. Характеристики работы нейроконтроллера при использовании его с реальным объектом управления и с НС-моделью (ГЗЛА X-43, режим полета $M = 6$): **(a)** сравнение характеристик; **(b)** величины ошибок отслеживания задающего сигнала. Обозначения: Plant, Object — объект управления, NNModel — НС-модель, $e_\alpha = \alpha - \alpha_{rm}$ — ошибка отслеживания как разность между значениями угла атаки для объекта управления и эталонной модели

2. Просчет объединенной сети на этом сегменте (определение поведения объекта управления при текущих параметрах нейроконтроллера), определение ошибки следования ЭМ, вычисление якобиана этой ошибки по параметрам нейроконтроллера.
3. Корректировка параметров на текущем сегменте с использованием уравнений фильтра Калмана.

В режиме реального времени нейроконтроллер обучается по такой же схеме, но с некоторыми отличиями:

1. Последовательные сегменты представляют собой скользящий интервал (обычно 50 точек — 0.5 с). Однако параметры обновляются не на каждом шаге моделирования (0.01 с), а каждую 0.1 с.
2. Одновременно учится и НС-модель, а поэтому меняются параметры модельной подсети в объединенной системе.

Следует отметить, что нейроконтроллер учится управлять не самим объектом, а его моделью, поэтому, если НС-модель не обладает требуемой точностью, то качество управления будет неудовлетворительным.

Модель не может быть абсолютно точной, так как нейросетевой подход в своей основе дает приближенные решения. Следовательно, с помощью такого «чистого» подхода невозможно добиться точного управления (точного следования ЭМ).

Этот результат показан на рис. 4.4. Для сравнения на этом же рисунке показана работа нейроконтроллера с тем объектом, которому он обучался (НС-модель). Видно, что точность работы НК с реальным объектом несколько снижается, что свидетельствует о наличии уклонения поведения реального объекта от поведения его НС-модели. Способ повышения точности работы НК в данной ситуации рассматривается в следующем разделе.

4.1.3 Компенсирующий контур в схеме адаптивного управления с эталонной моделью

Ошибки и неточность нейросетевой модели можно интерпретировать как возмущения, приводящие к отклонению траектории реального объекта от эталонной траектории. Для уменьшения ошибки следования можно воспользоваться компенсатором (управление непосредственно по ошибке) — ему все равно какова природа возмущений и он очень хорошо встраивается в схему с эталонной моделью.

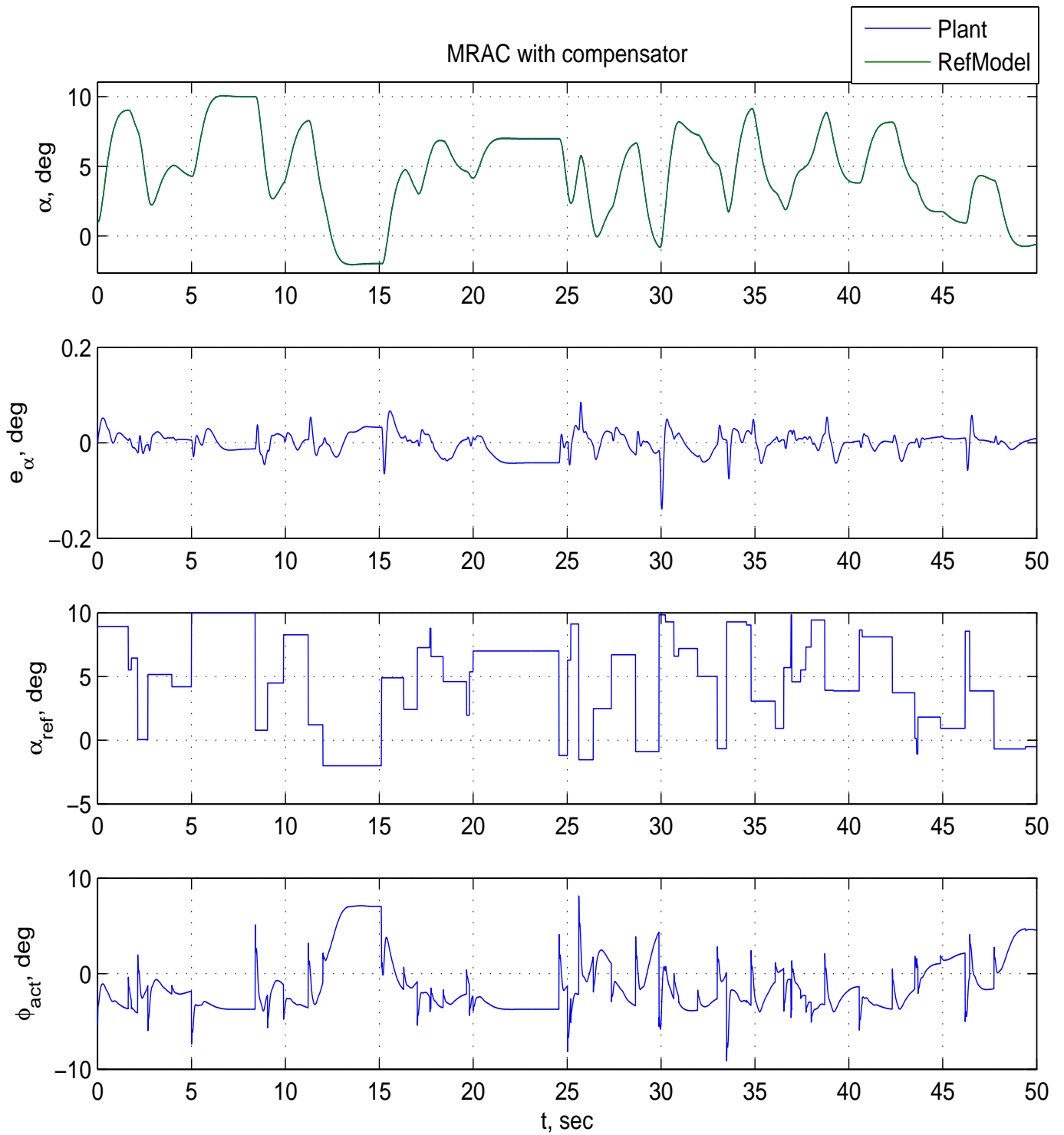


Рис. 4.5. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором (ГЗЛА Х-43, режим полета $M = 6$). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

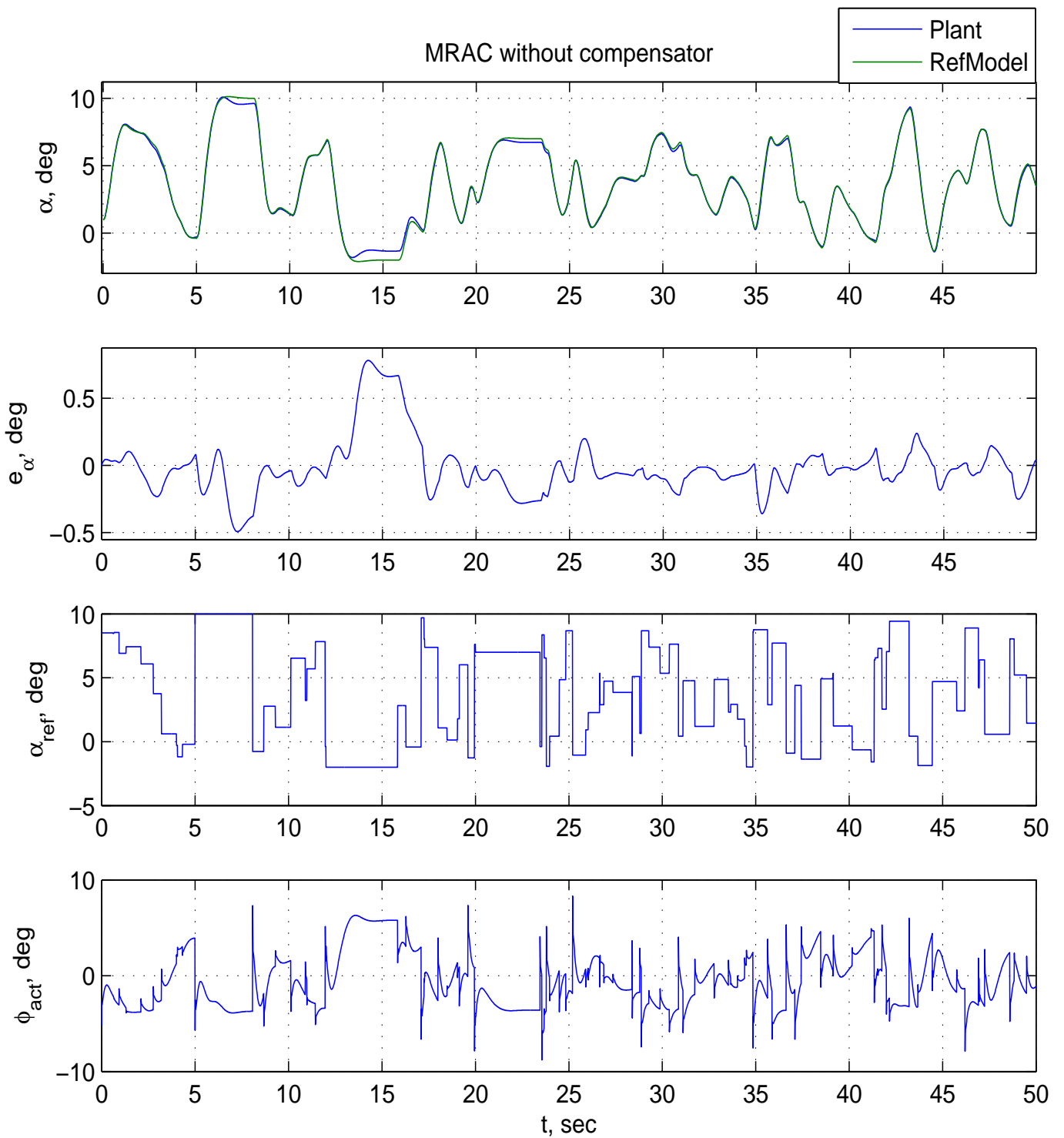


Рис. 4.6. Результаты вычислительного эксперимента для системы управления с эталонной моделью и без компенсатора (ГЗЛА Х-43, режим полета $M = 6$). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

В простейшем случае компенсатор (ПД-компенсатор) реализует через дополнительную обратную связь закон управления вида [61] (см. также рис. 4.1 и 4.8):

$$\varphi_{\text{доп}} = K_p e + K_d \dot{e}, \quad (4.7)$$

где $e = y_{\text{эм}} - y$ — ошибка следования ЭМ, $y_{\text{эм}}$. В системе управления компенсатор дискретизирован, \dot{e} вычисляется как конечная разность.

Несмотря на простоту, компенсирующий контур снижает ошибку следования примерно на порядок. Сопоставить эффект от применения компенсатора можно по данным, приведенным для случая ГЗЛА на рис. 4.5 (компенсатор используется) и на рис. 4.6 (компенсатор не используется). Более сложный случай иллюстрируется с помощью рис. А.3 и А.4, где показано влияние компенсатора при возникновении отказной ситуации, которая привела к смещению центровки на 5% назад в момент времени $t = 20$ с.

Можно использовать также интегральный компенсатор, представляющий собой фильтр вида [61]:

$$W_{\text{комп}} = \frac{W_{\text{эм}}^{-1}}{(\tau p + 1)^{n-m} - 1}, \quad (4.8)$$

где n — порядок числителя передаточной функции ЭМ, m — порядок ее знаменателя; τ — произвольная константа, настраиваемый вручную параметр контура компенсации (обратно пропорциональна коэффициенту усиления).

Применение интегрального компенсатора позволяет избавиться от установившейся ошибки, при этом будут полностью подавляться постоянные возмущения. Это видно, например, из рис. А.5 (см. Приложение А): когда возмущение, вызванное внешним воздействием, затухает, ошибка системы близка к нулю.

Однако в неустановившемся режиме интегральный компенсатор работает точно так же, как и ПД-компенсатор (см. рис. А.6 — без отказов, а также рис. А.7 — при возникновении отказной ситуации), а поскольку установившиеся режимы для систем рассматриваемого класса нехарактерны, то применительно к этим системам вполне можно обходиться более простым ПД-компенсатором.

4.1.4 Оценка влияния параметров эталонной модели на работоспособность синтезируемой системы управления

Выше отмечалось, что требуемый характер поведения рассматриваемой системы в проведенных исследованиях определялся с использованием эталонной модели (ЭМ). Как пока-

зывает опыт вычислительных экспериментов, целесообразно выбирать ЭМ таким образом, чтобы определяемый ею характер желаемого поведения был бы достаточно близким к поведению реального объекта управления. Если это условие не удовлетворяется, то контроллер, в стремлении максимально приблизить характер поведения системы к желаемому, будет выдавать слишком большие значения командных сигналов для приводов органов управления, что может привести к существенному ухудшению качества регулирования.

С учетом этих соображений, в разд. 4.4.2.1 для рассматриваемых схем адаптивного управления была введена эталонная модель, представляющая собой последовательное соединение аperiodического и колебательного звеньев:

$$W_{\text{ЭМ}} = \frac{\omega_{\text{ЭМ}}^2}{(T_{\text{пр}}p + 1)(p^2 + 2\zeta_{\text{ЭМ}}\omega_{\text{ЭМ}}p + \omega_{\text{ЭМ}}^2)}.$$

Параметрами данной эталонной модели являются собственная частота $\omega_{\text{ЭМ}}$, коэффициент относительного демпфирования $\zeta_{\text{ЭМ}}$ и постоянная времени «префильтра» $T_{\text{пр}}$. Формально аperiodическое звено в ЭМ не является префильтром, но выполняет такую его функцию, как сглаживание резких сигналов, поступающих на привод. Данная структура эталонной модели была выбрана исходя из следующих соображений:

1. Переходные процессы в угловом продольном движении являются колебательными (система нелинейных дифференциальных уравнений, описывающая продольное угловое движение самолета, имеет второй порядок), поэтому основу эталонной модели составляет колебательное звено.
2. Под объектом управления понимается самолет с приводом органа управления, поэтому чисто колебательная ЭМ недостижима для контроллера, так как следование ей требует очень больших сигналов, поступающих на привод. Чтобы этого избежать, в ЭМ введено аperiodическое звено, играющее роль префильтра.
3. Параметры ЭМ выбраны исходя из ее достижимости для объекта с контроллером, а также с учетом требуемого диапазона отклонения рулей.

Величина коэффициента относительного демпфирования $\zeta_{\text{ЭМ}}$ в ЭМ экспериментально была выбрана равной 0.8, чтобы добиться компромисса между отсутствием перерегулирования и затянутостью переходного процесса, реализуемого ЭМ. Постоянная времени префильтра $T_{\text{пр}}$ была принята равной 0.05 с.

Основным параметром ЭМ является частота $\omega_{\text{ЭМ}}$, которая определяет желаемое быстродействие системы в целом. С физической точки зрения увеличить скорость реакции можно

только с помощью большего отклонения рулей в переходном режиме.

Известно, что желаемое быстродействие системы выбирается на основе компромисса между скоростью реакции системы и скоростью отклонения рулей, что было продемонстрировано в серии вычислительных экспериментов применительно к рассматриваемой нелинейной системе.

Как уже отмечалось, желаемое качество управления в рассматриваемых схемах адаптивного управления задается с помощью ЭМ. Влияние желаемого быстродействия на расход рулей и на нагрузку на привод показано на примере схемы с эталонной моделью, результаты соответствующих вычислительных экспериментов представлены на рис. 4.7, а также на рис. А.8–А.11.

По результатам этих экспериментов видно, что чем больше собственная частота эталонной модели, тем больше ошибки слежения в переходном процессе: она возрастает от ± 0.1 град. при $\omega_{эм} = 1.5$ 1/с до ± 2.1 град. при $\omega_{эм} = 4$ 1/с. Также заметно увеличение расхода рулей при большом заданном быстродействии. Эксперименты, результаты которых приведены на рис. А.8–А.11, выполнялись для ГЗЛА. Эффективность элевонов в функции руля высоты у данного ЛА невелика, кроме того, значительные отклонения рулей расходуются на балансировку на разных углах атаки. В связи с этим представляется разумным ограничить частоту ЭМ значением $\omega_{эм} = 2$ 1/с, чтобы оставить некоторый запас по отклонению рулей.

Полученные результаты показывают также, что в некоторых случаях достигается установленное ограничение привода по скорости (± 60 град/с), что сказывается на переходном процессе (рис. А.11) и может приводить к неустойчивости в случае большого отклонения от желаемого движения. Для уменьшения используемой скорости привода можно увеличить постоянную времени $T_{пр}$ в эталонной модели, но это может привести к росту запаздывания в системе, при этом (при заданном приводе) чем больше собственная частота эталонной модели $\omega_{эм}$, тем на большей части времени переходного процесса будет сказываться влияние привода.

4.1.5 Адаптивное управление с эталонной моделью применительно к угловому движению ЛА

1. Для оценки свойств получаемой адаптивной системы с ЭМ была проведена серия вычислительных экспериментов. Часть результатов этих экспериментов была уже представлена выше. В частности, на рис. 4.4 применительно к ГЗЛА Х-43 (полет при $M = 6$) было показано влияние точности НС-модели на характеристики получаемого НК.

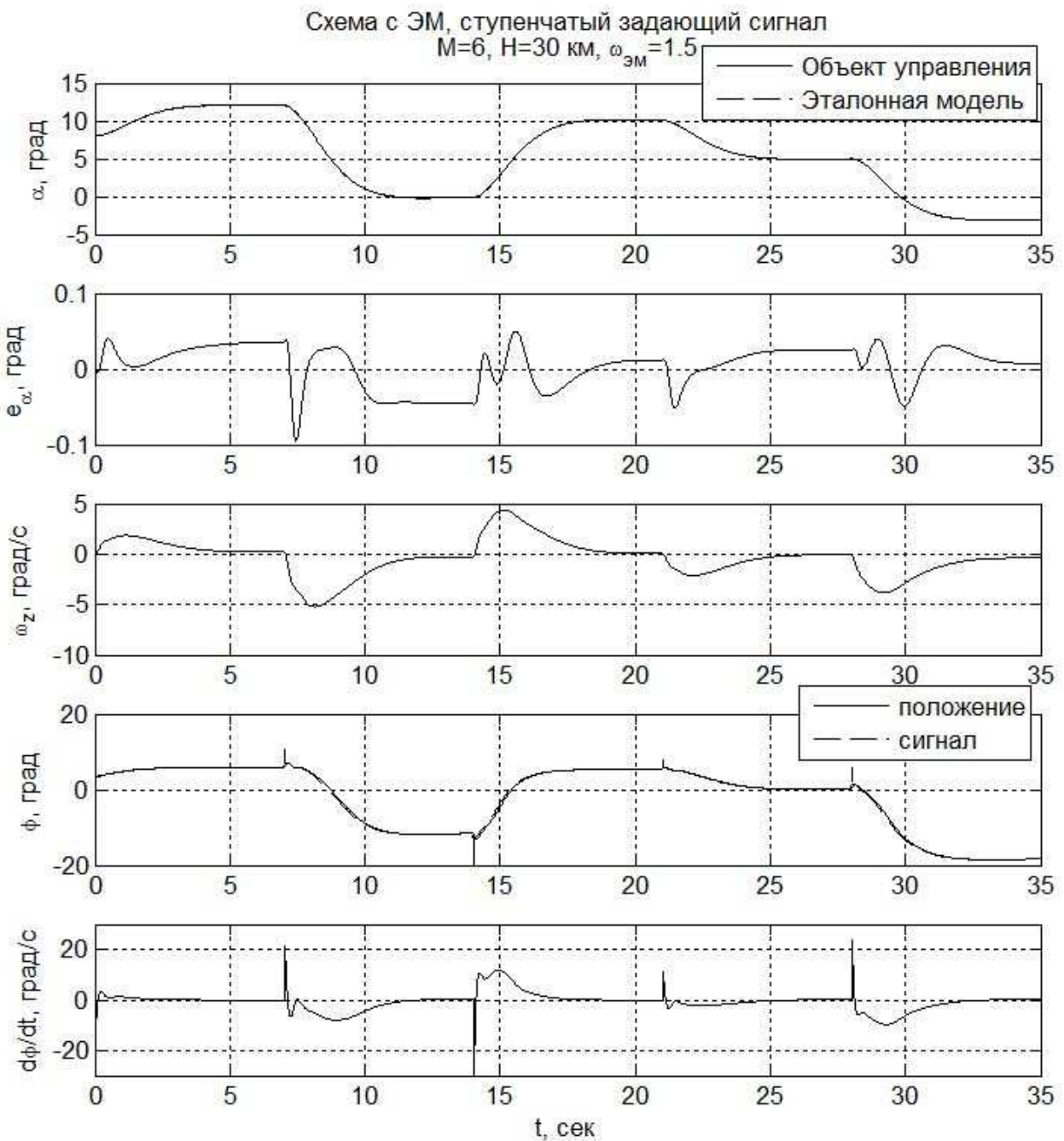


Рис. 4.7. Результаты вычислительного эксперимента для системы управления с эталонной моделью применительно к перспективному ГЗЛА для оценки влияния собственной частоты $\omega_{\text{ЭМ}}$ эталонной модели ($\omega_{\text{ЭМ}} = 1.5$, ступенчатый задающий сигнал по углу атаки, режим полета $M = 6, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

Результаты, представленные на рис. А.1–А.4 применительно к этому же ЛА, демонстрируют эффект от введения компенсирующего контура в состав адаптивной системы с ЭМ.

2. Дополнительные данные показаны на рис. А.12–А.16. В частности, рис. А.12 показывает, как влияет на характеристики системы управления с ЭМ и компенсатором недостаточная точность НС-модели. Видно, что несмотря на несовершенство используемой НС-модели, качество управления остается достаточно высоким (значения ошибки отслеживания лежат в диапазоне от -0.2 град до $+0.2$ град), хотя и менее высоким, чем при более точной НС-модели. В этом заслуга компенсатора, без которого в данном случае значения ошибки становились недопустимо большими.

3. Данные, представленные на рис. А.13, А.14 и А.15 демонстрируют работу системы управления с ЭМ и компенсатором под воздействием традиционно используемого ступенчатого воздействия по задающему сигналу, а на рис. А.16 показана работа такой системы для последовательности ступенчатых изменений задающего сигнала, следующих через промежутки времени, в течение которых предыдущее возмущение уже парировано, т. е. эти ступенчатые воздействия можно считать изолированными. Такого рода проверка динамических свойств управляемой системы остается важной и для систем рассматриваемого в данной работе класса, поскольку позволяет наглядно наблюдать и оценивать характер реакции системы управления при возникновении некоторого возмущающего воздействия.

Однако в современной практике тестирования нелинейных систем принято вместо изолированных ступенчатых возмущающих воздействий, характерных для линейных систем, использовать более сложные входные (задающие) сигналы, позволяющие выполнять тестирование в существенно более жестком режиме. А именно, задается входной сигнал, который часто и значительно изменяется по величине, поэтому системе управления приходится начинать реагировать на некоторое i -е возмущение еще до того, как завершился переходный процесс, связанный с реагированием на $(i - 1)$ -е возмущение (возможно также и на $(i - 2)$ -е, $(i - 3)$ -е, ...). Другими словами, при таком подходе к тестированию систему заставляют реагировать постоянно не на одно какое-то возмущение, а на их комбинацию («смесь»), компоненты которой находятся на разных стадиях завершения переходных процессов, порожденных ими, причем комбинация эта постоянно меняется случайным образом. Здесь системе управления приходится работать в значительно более сложных условиях, чем при традиционном ступенчатом воздействии, но данный подход лучше отвечает природе задач, возникающих при управлении нелинейными системами, действующими в условиях неопределенности. Напри-

мер, атмосферная турбулентность, воздействуя на ЛА, не будет «дожидаться», пока система управления отработает ее предыдущее воздействие.

По этой причине в примерах тестирования алгоритмов адаптивного управления, рассматриваемых ниже, в основном используется сложное задающее воздействие с входным сигналом, который меняется, как правило, часто и на значительную величину.

Задающий сигнал по углу атаки, отслеживание которого должна обеспечить рассматриваемая система управления, строился при этом по тем же правилам, что и командный сигнал привода органа управления при формировании обучающего набора для НС-модели (см. разд. 2.3). А именно, формировалась случайная последовательность ступенчато изменяемых отслеживаемых значений угла атаки, при частых и существенных перепадах между соседними значениями элементов данной последовательности. Такой подход призван обеспечить возможно большее разнообразие состояний моделируемой системы (чтобы покрыть по-возможности равномерно и плотно все пространство состояний системы), а также возможно большее разнообразие перепадов соседних по времени состояний (чтобы максимально достоверно отразить в алгоритме управления, реализуемом НК, динамику объекта).

4. На рис. А.17–А.21 показана работа системы управления с ЭМ и компенсатором для случая, когда в качестве объекта управления рассматривался самолет F-16 и последовательно возникали две отказные ситуации, приводившие к смещению центра масс самолета, а также к уменьшению эффективности его органа управления продольным движением.

На рис. А.22 показано, как адаптивная система управления с ЭМ и компенсатором справляется с влиянием двух последовательных отказов, существенно влияющих на динамику ГЗЛА X-43. Первый из них приводит к смещению центровки на 10% назад (при $t = 20$ с), второй – к уменьшению на 50% эффективности органа управления продольным движением (при $t = 50$ с). Видно, что использованная схема адаптации обеспечивает работу с незначительной ошибкой ($e_\alpha \approx \pm 0.05$ град) до момента возникновения первой отказной ситуации. Адаптация к изменению динамики объекта, вызванному этой ситуацией, происходит достаточно быстро (примерно за 1.2–1.5 с). Ошибка теперь (до момента возникновения второй отказной ситуации) стала больше, но она укладывается, в основном, в диапазон $e_\alpha \approx \pm 0.2$ град, устойчивость работы системы сохранена. После второго отказа устойчивость сохранена, но значения ошибки становятся довольно значительными ($e_\alpha \approx \pm 0.5$ град).

На рис. А.23–А.27 показаны результаты аналогичных вычислительных экспериментов для ГЗЛА NASP.

5. В рассмотренных выше вычислительных экспериментах использовалась модель движения ЛА вида (3.13) с единственной управляющей переменной ϕ_a . Связь между углом атаки α и тягой P в этой модели вводилась через значения коэффициента момента тангажа $m_z(\alpha, P)$ (см. рис. 3.3), дополнительные эффекты от влияния тяги на угол атаки и угла атаки на тягу не учитывались, управление тягой также не вводилось.

Для того, чтобы оценить значимость факторов, исключенных из рассмотрения в серии экспериментов, описанной выше, была выполнена еще одна серия вычислительных экспериментов, проведенная также применительно к ГЗЛА Х-43, который совершал крейсерский полет при числе Маха $M = 6$.

В этой серии модель движения имела вид (3.14), т.е. было учтено взаимовлияние угла атаки и тяги, а для парирования ошибок по углу атаки и тангенциальной (скоростной) перегрузке в дополнение к управлению по углу атаки (командный сигнал φ_{act}) было введено также и управление по тяге двигателя (командный сигнал δ_{rud}).

6. Был рассмотрен ряд вариантов комбинаций используемых задающих сигналов по обоим каналам, применение схем с компенсатором и без него в одном или обоих каналах. Все варианты рассматривались для схемы адаптивного управления с эталонной моделью. В число этих вариантов вошли следующие:

1. Постоянная заданная перегрузка $n_{x_a} = 0$, заданный угол атаки — случайный. Компенсаторы в обоих каналах.
2. Постоянная заданная перегрузка $n_{x_a} = 0$, заданный угол атаки — случайный, в канале перегрузки нет компенсатора.
3. Постоянный заданный угол атаки (2 град), случайная заданная перегрузка, компенсаторы в обоих каналах.
4. Постоянный заданный угол атаки (2 град), случайная заданная перегрузка, нет компенсатора в канале перегрузки.
5. Оба задающих сигнала — случайные ступенчатые. Компенсаторы в обоих каналах.
6. Оба задающих сигнала — случайные ступенчатые. Нет компенсатора в канале перегрузки.
7. Оба задающих сигнала — случайные ступенчатые. Во всех каналах отсутствуют компенсаторы.

Результаты вычислительных экспериментов для этих вариантов представлены на рис. А.28–А.41. Вследствие того, что по каждому из вариантов приводится 8 графиков, они разделены по две части по 4 графика для каждого из вариантов. На первом графике в каждой из пар показано поведение объекта, эталонной модели и управляющие сигналы, на втором графике — ошибки отслеживания и задающие сигналы.

7. Результаты, полученные в этой серии экспериментов, позволяют сделать следующие выводы. Точность отслеживания угла атаки при использовании расширенной модели движения (3.14) несколько снижается.

В случае, когда используется компенсатор в обоих каналах, величина ошибки лежит в пределах от $-(0.02 \div 0.04)$ до $+(0.01 \div 0.02)$ по n_{x_a} и от $-(0.05 \div 0.20)$ до $+(0.7 \div 1.1)$ по α . При этом наименьшие значения ошибок получаются при выдерживании режима с $\alpha = \text{const} = 2$ град., а наибольшие, когда задающие сигналы по обоим каналам — случайные, с резко, часто и существенно меняющейся амплитудой. Такой режим нехарактерен для ЛА рассматриваемого класса в его нормальном полете. Он скорее будет адекватен некоторой нештатной ситуации, в которой резко и часто меняются параметры полета ЛА. Но и в этой достаточно сложной ситуации система управления справилась со своими функциями вполне успешно.

В случае, когда компенсатор использовался только в канале угла атаки, а в канале перегрузки его не было, качество управления несколько ухудшилось. А именно, величина ошибки в этом случае лежит в пределах от $-(0.06 \div 0.10)$ до $+(0.05 \div 0.08)$ по n_{x_a} и от $-(0.12 \div 0.60)$ до $+(0.4 \div 1.2)$ по α . Видно, что в относительных единицах ошибка увеличилась, причем по перегрузке в большей степени, чем по углу атаки, однако в абсолютных единицах величины ошибок остаются вполне приемлемыми, т. е. алгоритм адаптации работает достаточно эффективно, несмотря на усложнившиеся условия.

Отсутствие компенсаторов в обоих каналах сказывается более существенным образом. Величина ошибки в этом случае лежит в пределах от -0.10 до $+0.08$ по n_{x_a} и от -1.2 до $+2.1$ по α . Таким образом, роль компенсатора в рассматриваемой схеме адаптивного управления довольно значительна, но все же не критична.

8. Влияние изменения угла атаки на продольное траекторное движение за счет связи между углом атаки и тягой двигателя можно оценить по динамике и границам изменения значений перегрузки n_{x_a} .

Из результатов, представленных на рис. А.28–А.41, видно, что перегрузка в случае исполь-

зования компенсаторов в обоих каналах лежит в пределах от $-(0.03 \div 0.15)$ до $+(0.01 \div 0.15)$, при отсутствии компенсатора в канале перегрузки — в пределах от $-(0.10 \div 0.18)$ до $+(0.10 \div 0.17)$, а при отсутствии компенсаторов в обоих каналах — от -0.18 до $+0.19$). Таким образом, положительное влияние компенсаторов в системе управления рассматриваемого класса видно и здесь, но общее влияние изменений по углу атаки на продольное траекторное движение можно признать несущественным. По этой причине в оценочных расчетах эффективности алгоритмов адаптации вполне допустимо пользоваться более простой одноканальной моделью (3.13) вместо двухканальной модели (3.14). Разумеется, для окончательной отработки этих алгоритмов необходимо пользоваться полной моделью движения ЛА.

9. На рис. А.42–А.45 приведены результаты вычислительных экспериментов еще для одного класса летательных аппаратов, а именно, для микро-БПЛА и мини-БПЛА. При этом рис. А.42 и А.43 демонстрируют работу адаптивной системы управления в штатных условиях функционирования, а рис. А.44 и А.45 показывают, как адаптивная система управления с ЭМ и компенсатором справляется с влиянием двух последовательных отказов, существенно влияющих на динамику объекта. Первый из них приводит к смещению центровки на 10% назад (при 5 с для микро-БПЛА «003» и 10 с для мини-БПЛА Х-04), второй — к уменьшению на 50% эффективности органа управления продольным движением (при 10 с для микро-БПЛА и 20 с для мини-БПЛА). Видно, что использованная схема адаптации обеспечивает работу с незначительной ошибкой (как правило, $e_\alpha \approx \pm 0.05^\circ$) до момента возникновения первой отказной ситуации. Адаптация к изменению динамики объекта, вызванному этой ситуацией, происходит достаточно быстро (примерно за 1,2–1,5 с). Ошибка теперь (до возникновения второй отказной ситуации) стала больше, но она укладывается, в основном, в диапазон $e_\alpha \approx \pm 0.2^\circ$, устойчивость работы системы сохранена. После второго отказа устойчивость сохранена, но значения ошибки становятся довольно значительными (большой частью, их значения лежат в диапазоне $e_\alpha \approx \pm 0.5^\circ$).

10. Результаты вычислительных экспериментов, представленные в данном разделе, убедительно показывают, что в подавляющем большинстве случаев адаптивная нейросетевая система управления, структура которой представлена на рис. 4.1, успешно справляется со своими задачами, в том числе с учетом взаимосвязей между углом атаки и тягой двигателя ЛА, а также и при возникновении отказных ситуаций, когда реконфигурации алгоритма управления движением позволяет оперативно парировать последствия отказов оборудования и поврежденной конструкции ЛА.

4.2 Адаптивное управление с прогнозирующей моделью

4.2.1 Общая схема адаптивного управления с прогнозирующей моделью

В задаче управления с прогнозирующей моделью (MPC — Model Predictive Control) используется модель объекта, с помощью которой прогнозируется его будущее поведение, а также оптимизационный алгоритм для выбора управляющего воздействия, обеспечивающего наилучшие значения прогнозируемых характеристик рассматриваемой системы.

Управление с прогнозирующей моделью основывается на методе скользящего горизонта, согласно которому НС-модель предсказывает выход объекта управления через заданный промежуток времени (горизонт прогноза). Полученные результаты прогноза используются алгоритмом численной оптимизации для того, чтобы найти значение управления u , которое минимизирует на заданном горизонте прогноза следующий критерий качества управления:

$$J = \sum_{j=N_1}^{N_2} (y_r(t+j) - y_m(t+j))^2 + \rho \sum_{j=1}^{N_u} (u'(t+j-1) - u'(t+j-2))^2.$$

Здесь N_1 , N_2 и N_u — числовые параметры, определяющие горизонт прогноза, в пределах которого оцениваются значения ошибки слежения и приращений управляющего сигнала. Величины (y_r и y_m — желаемый выход объекта управления и выход НС-модели, соответственно, u' — пробные управляющие воздействия, ρ — весовой коэффициент, задающий относительную долю вклада отклонений по управлениям в общее значение критерия эффективности J).

Помимо горизонта прогноза, вторым важным параметром в схеме с прогнозирующей моделью является значение горизонта управления, т. е. величины промежутков времени, через которые алгоритм оптимизации выдает управляющий сигнал, величина которого не меняется до момента достижения следующего горизонта управления.

В общем случае горизонты управления и прогноза не совпадают. Как показывает вычислительный эксперимент, соотношение этих горизонтов в значительной степени определяет устойчивость схемы управления с ПМ. Согласно полученным экспериментальным данным горизонт управления целесообразно выбирать значительно меньше горизонта прогноза. В частности, в экспериментах, результаты которых представлены ниже, горизонт управления был принят минимальным (равным одному временному шагу Δt) и далее в пределах горизонта прогноза управление считалось постоянным. За счет такого выбора упрощаются вычисления на каждом шаге выработки управления и улучшается устойчивость алгоритма оптимизации в схеме с ПМ.

Горизонт прогноза в описываемых ниже экспериментах был принят равным 30 временным шагам (0.3 с), исходя из следующих соображений. Алгоритм оптимизации в схеме с ПМ пытается минимизировать прогнозируемое отклонение от эталонной траектории. Будем считать, что при наличии начальных отклонений прогнозируемая траектория должна сходиться к эталонной к концу интервала прогнозирования. Это значит, что чем меньше горизонт прогноза, тем большее по величине управляющее воздействие придется приложить к объекту, чтобы свести к нулю ожидаемое отклонение от ЭМ. Таким образом, горизонт прогноза определяет значение эффективного коэффициента усиления по ошибке слежения в схеме с ПМ: чем меньше горизонт прогноза, тем больше этот коэффициент. По этой причине минимальный горизонт прогноза ограничивается соображениями устойчивости динамической системы, поскольку при превышении некоторого порогового значения этого коэффициента устойчивость данной системы теряется. С другой стороны, существуют ограничения и на увеличение горизонта прогноза, связанные с вычислительной сложностью выработки управляющего сигнала, точностью слежения и приближенностью самого прогноза, связанной с постоянством управления на горизонте прогноза. В ходе вычислительных экспериментов было найдено компромиссное решение, согласно которому в решаемой для перспективного ГЗЛА задаче горизонт прогноза должен составлять 30 временных шагов.

Общая схема управления с прогнозирующей моделью показана на рис. 4.8, а соответствующая структурная схема — на рис. 4.9.

4.2.2 Синтез нейроконтроллера для адаптивного управления с прогнозирующей моделью

В данной схеме адаптивного управления используется только одна многослойная нейронная сеть, представляющая собой НС-модель объекта. Контроллер здесь представлен алгоритмом оптимизации. Структурная схема работы системы с прогнозирующей моделью иллюстрируется рис. 4.8.

Критерием качества для данной системы является среднеквадратическая ошибка на интервале прогноза (5–7 шагов), т. е. не берется в расчет прирост управления:

$$\begin{aligned}
 E(u) &= \frac{1}{2} e^T e, \quad e(u) = y - \hat{y}(u), \\
 e &= [e_{k+2}, e_{k+3}, \dots, e_{k+T}]^T, \\
 u &= [u_{k+1}, u_{k+2}, \dots, u_{k+T-1}]^T,
 \end{aligned}
 \tag{4.9}$$

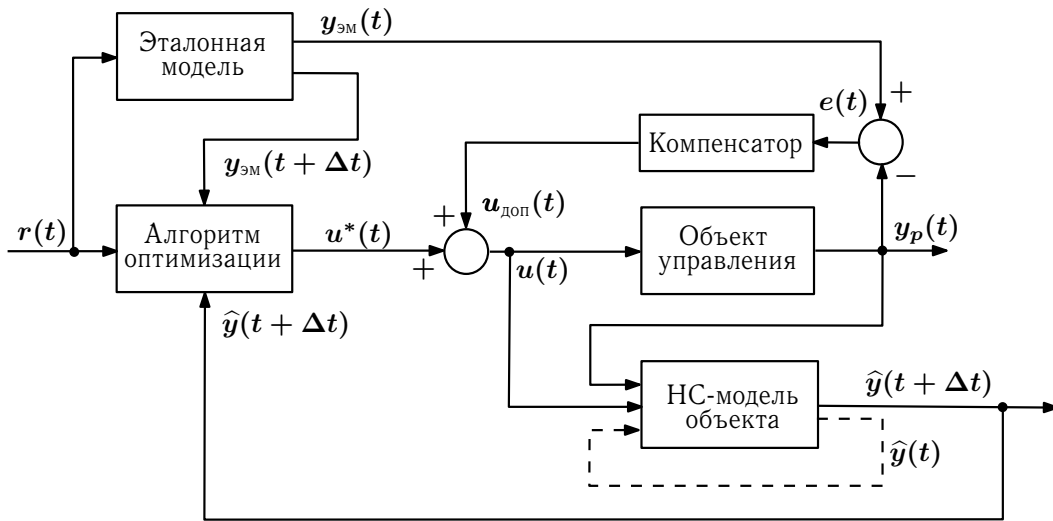


Рис. 4.8. Схема нейросетевого адаптивного прогнозирующего управления. Здесь обозначено: u^* — управление на выходе алгоритма оптимизации, $u_{\text{доп}}$ — добавочное управление от компенсатора, u — результирующее управление, y_p — выход объекта управления, \hat{y} — выход нейросетевой модели объекта управления; $y_{\text{эм}}$ — выход эталонной модели; ε — расхождение между выходами объекта управления и эталонной модели; r — задающее воздействие

где T — длина интервала прогноза.

Отсутствие в критерии слагаемого управления позволяет делать на каждом шаге всего одну итерацию методом Гаусса-Ньютона (благодаря тому, что в критерии качества ошибкой является отклонение траектории модели объекта от эталонной ($e = y - \hat{y}$), а не от заданного значения ($e = y - \hat{y}$)).

Таким образом, здесь выполняется минимизация отклонения поведения НС-модели от ЭМ не по параметрам нейроконтроллера (их тут нет вообще), а непосредственно по управлению на интервале прогноза. Чтобы применять эффективные методы оптимизации, необходимо вычислить динамический якобиан по управлению:

$$J_u = \begin{bmatrix} \frac{\partial e_{k+2}}{\partial u_{k+1}} & \frac{\partial e_{k+2}}{\partial u_{k+2}} & \cdots & \frac{\partial e_{k+2}}{\partial u_{k+T-1}} \\ \frac{\partial e_{k+3}}{\partial u_{k+1}} & \frac{\partial e_{k+3}}{\partial u_{k+2}} & \cdots & \frac{\partial e_{k+3}}{\partial u_{k+T-1}} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial e_{k+T}}{\partial u_{k+1}} & \frac{\partial e_{k+T}}{\partial u_{k+2}} & \cdots & \frac{\partial e_{k+T}}{\partial u_{k+T-1}} \end{bmatrix} \quad (4.10)$$

Это делается с помощью метода обратного распространения ошибки во времени (ВРТТ —

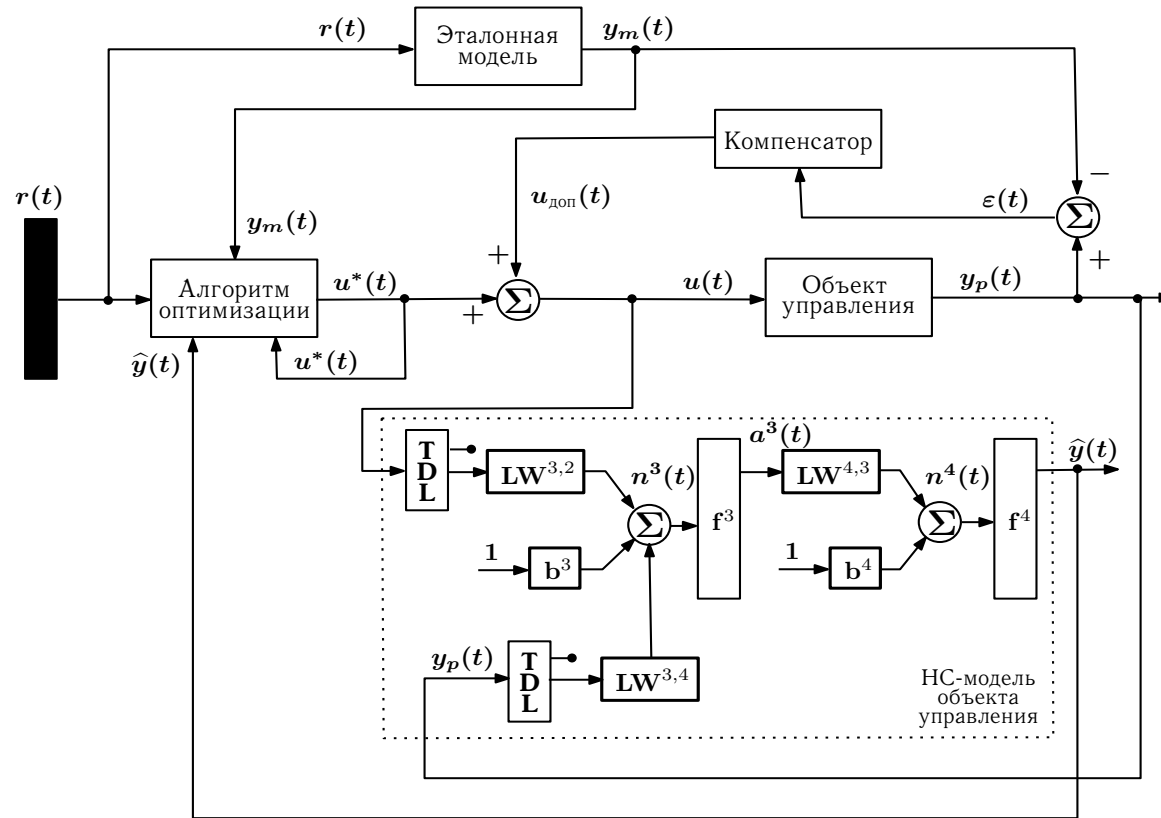


Рис. 4.9. Структурная схема нейросетевой реализации адаптивного управления с прогнозирующей моделью: TDL — линия задержки; LW — матрица синаптических весов связей между входным и первым обрабатывающим слоем НС; LW — матрица синаптических весов связей между обрабатывающими слоями НС; b — набор смещений слоя НС; f — набор активационных функций слоя НС; Σ — набор сумматоров слоя НС; $n(t)$ — набор скалярных выходов сумматоров; $a(t)$ — набор скалярных выходов активационных функций; $r(t)$ — задающий сигнал; $y_p(t)$ — выход объекта управления; $\hat{y}(t)$ — выход НС-модели; $y_m(t)$ — выход эталонной модели; $u_*(t)$ — управление, вырабатываемое алгоритмом оптимизации; $u_{\text{доп}}(t)$ — добавочное управление от компенсатора; $u(t)$ — управление, поступающее на вход объекта; $\varepsilon(t) = y_p(t) - y_m(t)$ — расхождение между выходами объекта и эталонной модели

Back Propagation Through Time) для замкнутой нейросетевой модели.

Метод Гаусса-Ньютона очень похож на метод Левенберга-Марквардта, отличается только тем, что коэффициент μ в нем не меняется, он выбирается экспериментально для решаемой задачи:

$$u_{n+1} = u_n + (J_u^T J_u + \lambda E)^{-1} J_u^T e. \quad (4.11)$$

С учетом сказанного, вычисление управления на каждом шаге интегрирования производится в следующей последовательности:

1. Построение желаемого поведения на интервале прогноза. Для этого ЭМ просчитывается на этом интервале при постоянном задающем сигнале (задающий сигнал $r_{k+1} \dots r_{k+T-1} = r_k$: за неимением более изощренных вариантов, предположим самое простое — последнее доступное значение этого сигнала).
2. Прогноз проведения объекта управления по его НС-модели на несколько шагов вперед. Начальными условиями для модели являются эталонная траектория и предыдущие значения управления, полученные с помощью такого контроллера.
3. Определение вектора ошибки следования, вычисление якобиана ошибки для каждого момента времени по управлению в каждый момент времени.
4. Корректировка вектора управления любым оптимизационным методом (в данном случае методом Гаусса-Ньютона). Пункты 2–4 представляют собой одну итерацию оптимизационной процедуры, выполняемой до разумного уменьшения ошибки прогноза.
5. В качестве управления (в реальный объект) на следующем шаге (процедура оптимизации занимает часть шага) берется первое управление на интервале прогноза (т. е. u_{k+1}). Этим же значением заполняется весь вектор начального приближения управления на следующем шаге.

4.2.3 Адаптивное управление с прогнозирующей моделью применительно к угловому движению ЛА

Поведение ЛА под управлением системы с прогнозирующей моделью в значительной степени подобно его поведению под управлением системы с эталонной моделью.

Для оценки эффективности алгоритма адаптивного управления с ПМ была проведено несколько серий вычислительных экспериментов. Вначале выполнялось тестирование алгоритма с ПМ для ступенчатого задающего сигнала по углу атаки. Требовалось синтезировать

закон управления продольным угловым движением перспективного ГЗЛА Х-43, который позволял бы обеспечить высокую точность стабилизации требуемого угла атаки, определяемого входным задающим сигналом при различных сочетаниях числа Маха и высоты полета, характерных для данного ЛА. На рис. 4.10 приведен пример результатов из данной серии, расширенный набор этих результатов дается на рис. А.46–А.54. Из полученных результатов видно, что законы управления, синтезированные с привлечением механизмов адаптации с ПМ, обеспечивают достаточно высокое качество управления. А именно, для всех исследованных режимов полета ошибка отслеживания заданного угла атаки при резком его изменении на величину до 12° не превышала $\pm 0.27^\circ$, а в ряде случаев падала до $\pm 0.08^\circ$. После того, как завершался переход к новому значению угла атаки, значение ошибки отслеживания снижалось практически во всех случаях до $\pm(0.01^\circ \div 0.02^\circ)$. Приведенные результаты показывают также характер работы органа продольного управления ГЗЛА (элевонов), требуемого для реализации синтезированного закона управления, путем сопоставления значений командного сигнала, поступающего на вход привода элевонов, с углом отклонения элевонов. Кроме того, анализируя данные по скорости отклонения элевонов, требуемой для реализации полученного закона управления, можно выявить требования к приводам элевонов. Из результатов вычислительных экспериментов следует, что требуемая скорость отклонения элевонов лежит в пределах $\pm 50^\circ/\text{с}$.

Эти данные можно сравнить с результатами для ГЗЛА Х-43 (полет с числом Маха $M = 6$), которые приведены на рис. А.55, где показана работа системы в условиях отсутствия отказов и повреждений при более сложном (случайном) характере задающего воздействия. На рис. А.56 демонстрируется работоспособность рассматриваемой системы в условиях двух последовательных отказов, влияющих на динамику ГЗЛА. Эти отказы вызвали смещение центровки на 5% назад в момент времени $t = 30$ с, а затем 30% уменьшение эффективности органа управления при $t = 60$ с.

Аналогичные вычислительные эксперименты применительно к адаптивной системе управления с прогнозирующей моделью были выполнены для маневренного самолета F-16 (результаты приведены на рис. А.57, А.58 и А.59), а также для БПЛА (работа системы управления с ПМ в штатном режиме — см. рис. А.60 и А.61, при возникновении отказов — см. рис. А.62).

Выводы, которые следуют из результатов вычислительных экспериментов, представленных в данном разделе, в целом аналогичны тем, что были сделаны для систем с эталонной моделью. А именно, в подавляющем большинстве случаев адаптивная нейросетевая систе-

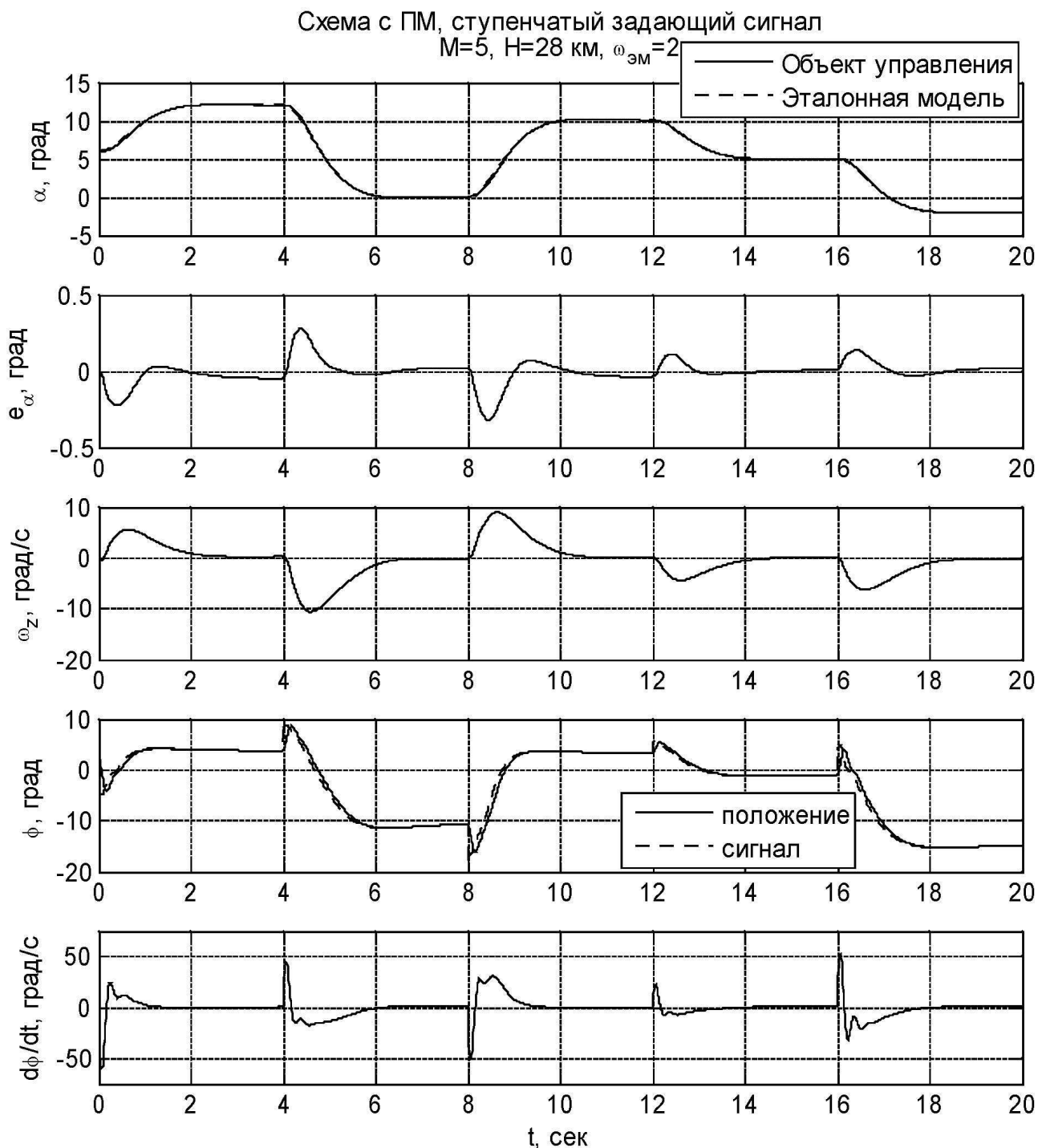


Рис. 4.10. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА при ступенчатом задающем сигнале по углу атаки (режим полета $M = 5, H = 28 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град;; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

ма управления, структура которой представлена на рис. 4.8, успешно справляется со своими задачами, в том числе и при возникновении отказных ситуаций.

Сопоставление схем адаптивного управления с эталонной моделью и с прогнозирующей моделью не позволяет отдать предпочтение ни одной из них. У каждой из них есть свои положительные и отрицательные свойства. Окончательное решение в пользу какой-либо одной из этих схем можно сделать лишь применительно к конкретной решаемой прикладной задаче, после проведения достаточно обширной серии вычислительных экспериментов.

4.3 Адаптивное управление с обратной динамикой и инверсной моделью

4.3.1 Адаптивное управления на основе метода обратной задачи динамики

4.3.1.1 Общая схема подхода к управлению на основе метода обратной задачи динамики

1. Результаты вычислительных экспериментов, осуществленных для адаптивных систем с эталонной моделью (ЭМ) и с прогнозирующей моделью (ПМ), представленные, соответственно, в разд. 4.1.5 и 4.2.3, показывают, что в ряде случаев они не вполне отвечают предъявляемым требованиям по качеству управления. В связи с этим, в рассмотрение была введена еще одна схема адаптивного управления, а именно, схема, построенная на базе метода обратной задачи динамики (ОД), которая в ряде случаев превосходит схемы управления с ЭМ и ПМ.

2. Обратными задачами динамики принято называть такие задачи, в которых для заданного объекта требуется найти закон управления, обеспечивающий реализацию некоторого предписанного движения этого объекта [67–70].

Концепция обратной задачи динамики управляемых систем первоначально возникла в рамках механики, а затем была распространена и на системы других классов.

Введем эту концепцию на примере динамической системы (ДС), движение которой описывается уравнением вида

$$m \ddot{x} + f(x, \dot{x}) = F(t), \quad (4.12)$$

где m — масса системы; переменные $x(t)$, $\dot{x}(t)$ характеризуют положение системы и скорость ее движения в каждый момент времени $t \geq 0$; $F(t)$ — сила, действующая на систему.

3. Для управляемой динамической системы можно сформулировать два класса задач. В задачах первого класса считается известной математическая модель ДС вида (4.12), а также

состояние ДС в начальный момент времени:

$$x(0) = x_0, \quad \dot{x}(0) = \dot{x}_0. \quad (4.13)$$

Сила $F(t)$ в (4.12) задана. Требуется найти траекторию движения системы, т.е. значения величин

$$x(t), \dot{x}(t), t \geq 0. \quad (4.14)$$

Решение сформулированной задачи сводится к интегрированию дифференциального уравнения (4.12) с начальными условиями (4.13).

В задачах второго класса, так же как и в задачах первого класса, математическая модель ДС (4.12), а также начальные условия (4.13) считаются известными. Кроме того, задается требуемая траектория движения рассматриваемой системы

$$x(t) = x_m(t), \dot{x}(t) = \dot{x}_m(t), t \geq 0. \quad (4.15)$$

Требуется найти силу $F(t)$, обеспечивающую реализацию движения (4.15).

4. Задачи сформулированных двух классов прямо противоположны по содержанию. Их принято именовать, соответственно, прямыми задачами динамики и обратными задачами динамики.

Если $F(t)$ в (4.12) представляет собой управляющее воздействие на ДС, то обратную задачу динамики можно трактовать как задачу синтеза такого алгоритма управления, при котором ДС обладает требуемыми динамическими свойствами.

Целью управления при этом является реализация такого движения ДС, которое воспроизводило бы с требуемой точностью некоторое желаемое движение $x_m(t), \dot{x}_m(t), t \geq 0$, задаваемое с помощью эталонной модели (ЭМ). Другими словами, должно выполняться соотношение

$$e = \|x(t) - x_m(t)\| + \lambda \|\dot{x}(t) - \dot{x}_m(t)\| \leq \varepsilon, \quad (4.16)$$

где e – ошибка воспроизведения требуемого движения, определяемая как взвешенная сумма норм разности между реализуемым движением $x(t), \dot{x}(t)$ и эталонным движением $x_m(t), \dot{x}_m(t)$; $\varepsilon > 0$ – требуемая точность воспроизведения эталонного движения. В идеальном варианте $e(t) = 0$, т.е.

$$\dot{x}(t) = \dot{x}_m(t). \quad (4.17)$$

5. Суть рассматриваемого метода можно пояснить на примере системы первого порядка с аддитивно входящим управлением:

$$\dot{x} = f(x) + g(x)u, \quad (4.18)$$

Согласно принципам обратной динамики, управление для системы (4.18) следует выбирать так, чтобы свести данную систему к виду $\dot{x}(t) = \dot{x}_m(t)$, или, в терминах ошибки слежения $e = x_m - x$, получаем соотношение $\dot{e} = 0$. Легко показать, что для решения данной задачи необходимо задать закон управления в виде

$$u(x, \dot{x}_m) = \frac{1}{g(x)}(\dot{x}_m - f(x)). \quad (4.19)$$

Справедливость этого утверждения можно проверить непосредственной подстановкой управления (4.19) в уравнение (4.18). Такая операция приводит к получению искомого соотношения (4.17) для идеального варианта управления системой.

6. Из приведенных рассуждений следует, что соотношение (4.17) будет выполняться, то есть управляемая система будет точно следовать заданной траектории, если удовлетворяются следующие условия:

- математическая модель (4.18) точно соответствует объекту управления;
- совпадают начальные условия для рассматриваемой ДС и для ЭМ, т. е. $x(0) = x_m(0)$;
- отсутствуют возмущения.

Перечисленные условия, однако, на практике обычно не выполняются, поэтому из-за неучтенных факторов, влияющих на поведение реальной системы, а также из-за приближенного знания функций $f(x)$ и $g(x)$ ошибка слежения будет расти со временем. Чтобы предотвратить такой рост ошибки, то есть парировать неопределенность и неточность априорной информации, необходимо скорректировать модель (4.17) таким образом, чтобы траектория системы (4.18) «притягивалась» бы к эталонной траектории, т.е. чтобы выполнялось соотношение (4.16). Это может быть сделано введением в соотношение (4.17) дополнительного члена, компенсирующего ошибку слежения:

$$\dot{x} = \dot{x}_m + K(x_m - x). \quad (4.20)$$

В этом случае

$$\dot{e} = -Ke. \quad (4.21)$$

Чтобы перейти от соотношения (4.17) к соотношению (4.20), необходимо соответственно изменить закон управления, придав ему следующую форму

$$u(x, x_m, \dot{x}_m) = \frac{1}{g(x)}(\dot{x}_m - f(x) + K(x_m - x)). \quad (4.22)$$

4.3.1.2 Структурная реализация подхода к управлению на основе метода обратной задачи динамики

1. Для высокоманевренного самолета (например, для самолета F-16), рассматриваемого как один из важных примеров объекта управления, характерна нелинейность аэродинамических характеристик во всем рабочем диапазоне, а также зависимость эффективности органов управления от ориентации ЛА относительно обтекающего его потока, а характеристик устойчивости данного летательного аппарата — от углов отклонения органов управления. В связи с этим, аддитивная форма математической модели (4.18), удобная для синтеза закона управления, не может быть использована и ее необходимо заменить моделью более общего вида:

$$\dot{x} = f(x, u). \quad (4.23)$$

Поскольку вместо соотношения (4.20) в случае системы (4.23) следует использовать соотношение вида

$$\dot{x} = f(x, u(x, x_m, \dot{x}_m)) = \dot{x}_m + K(x_m - x), \quad (4.24)$$

то выразить искомую управляющую функцию $u(x, x_m, \dot{x}_m)$ из данного уравнения аналитически и точно уже не представляется возможным. В связи с этим отыскивается приближенное решение, основанное на том, что функция $u(x, x_m, \dot{x}_m)$ аппроксимируется многослойной нейросетью прямого распространения (называемой нейроконтроллером) с помощью процедуры обучения, стандартной для сетей такого вида.

Слагаемое $K(x_m - x)$, как отмечалось выше, парирует действия возмущений и уменьшает ошибку слежения, т. е. оно является в определенном смысле стабилизирующим. Однако эффективность применения данного средства в значительной степени зависит от точности модели (4.18) и от качества приближения управления $u(x, x_m, \dot{x}_m)$ нейроконтроллером, что отрицательно сказывается на робастности системы управления.

2. Чтобы обеспечить требуемую робастность системы, можно для системы общего вида (4.23) вернуться к форме закона управления $u(x, \dot{x}_m)$, принятой для системы (4.18) с аддитивным управлением, но с компенсирующей составляющей, отделенной от неизвестных функций. Это возможно, если функция $g(x)$ знакопостоянна при всех допустимых x и диапазон ее изменения не слишком велик. Такая ситуация характерна для управления движением ЛА на докритических режимах. Управление в этом случае определяется в зависимости от предполагаемого вида функций $f(x)$ и $g(x)$, то есть от их оценок $\hat{f}(x)$ и $\hat{g}(x)$, следующим

образом

$$u(x, x_m, \dot{x}_m) = \frac{1}{\widehat{g}(x)} (\dot{x}_m - \widehat{f}(x)) + K(x_m - x), \quad (4.25)$$

а преобразованная система принимает вид:

$$\dot{x} = \frac{g(x)}{\widehat{g}(x)} \dot{x}_m + \left[f(x) - \frac{g(x)}{\widehat{g}(x)} \widehat{f}(x) \right] + g(x) K(x_m - x). \quad (4.26)$$

В таком случае даже при $\widehat{f}(x) = f(x)$ и $\widehat{g}(x) = g(x)$ уравнение ошибки выглядит как $\dot{e} = -g(x)Ke$, т.е. оно становится нелинейным. Потенциальная точность управления при этом несколько снижается, однако значительно возрастает робастность системы в целом: компенсатор $K(x_m - x)$ всегда работает на уменьшение ошибки слежения вне зависимости от точности оценок $\widehat{f}(x)$ и $\widehat{g}(x)$, а робастность самого компенсатора может быть обеспечена приданием ему достаточного запаса устойчивости.

Чтобы подчеркнуть различия между этими двумя формами компенсации, можно назвать вариант включения компенсирующего слагаемого в (4.22) «внутренней компенсацией», а вариант в выражении (4.25) — «внешней компенсацией».

Для объекта управления общего вида (4.23), если управляющая функция представлена нейроконтроллером, использование внешней компенсации в процессе адаптации и перенастройки позволяет добиться компромисса между робастными и адаптивными свойствами системы.

3. Структурная организация всей системы показана на рис. 4.11. Нейроконтроллер вырабатывает управляющий сигнал точного слежения за эталонной траекторией, а динамический компенсатор корректирует в сторону уменьшения отклонение e_m реальной траектории от эталонной. Желаемая траектория x_m, \dot{x}_m обычно вырабатывается эталонной моделью в зависимости от задающего сигнала $r(t)$.

Функция управления $u_{NN}(x, \dot{x}_m)$ для системы, показанной на рис. 4.11, удовлетворяет уравнению

$$\dot{x} = f(x, u_{NN}(x, \dot{x}_m)) = \dot{x}_m, \quad (4.27)$$

а на вход объекта поступает суммарное управление

$$u = u_{NN} + u_C, \quad (4.28)$$

где u_C — дополнительный управляющий сигнал, вырабатываемый в компенсирующем контуре.

Учет в законе управления сигналов x_m, \dot{x}_m дает возможность улучшить характеристики системы на различных частотах. При этом желаемая траектория x_m требуется для придания

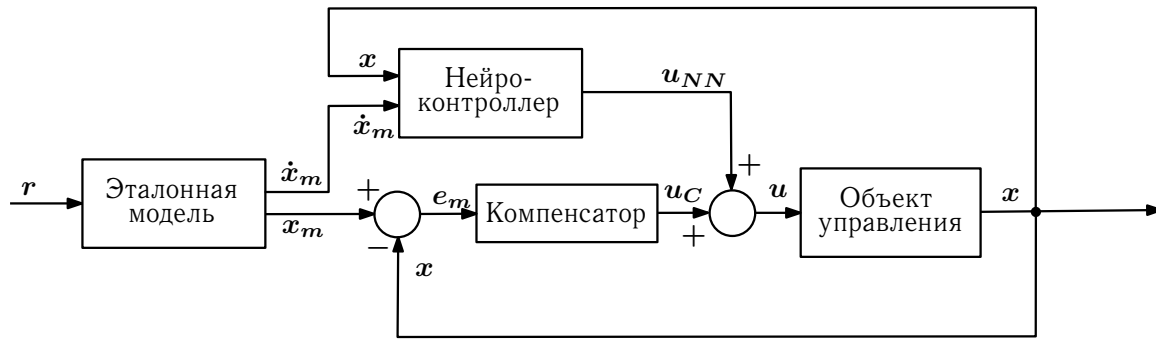


Рис. 4.11. Структурная схема системы управления по принципу обратной динамики: $r(t)$ — задающее воздействие; $x_m(t)$, $\dot{x}_m(t)$ — выходы эталонной модели; $x(t)$ — выход объекта управления; $\varepsilon_m(t)$ — расхождение между выходами объекта управления и эталонной модели; $u(t)$ — управление, подаваемое на вход ОУ; $u_{NN}(t)$ — управляющий сигнал от нейроконтроллера; $u_C(t)$ — управляющий сигнал от компенсатора

системе робастности и хорошего качества на малых частотах, поскольку компенсирующий контур представляет собой следящую систему с большим значением коэффициента усиления в обратной связи. Учет производной желаемой траектории \dot{x}_m при адекватной настройке нейроконтроллера позволяет устранить или значительно уменьшить ошибку слежения на средних и, в определенной степени, на больших частотах.

4. Метод обратной динамики обладает существенным преимуществом по сравнению с ранее рассмотренной схемой с эталонной моделью: все три составляющие, показанные на рис. 4.11 (эталонная модель, нейроконтроллер и компенсатор) являются в значительной степени самостоятельными элементами, их можно изменять независимо друг от друга. В противоположность этому, в схеме адаптивного управления с ЭМ эталонная модель и компенсатор взаимозависимы, их параметры варьировать можно только согласованно, что может существенно затруднить синтез требуемого закона управления.

4.3.2 Адаптивное управление с инверсной моделью

4.3.2.1 Схема адаптивного управления с инверсной моделью

1. Как отмечалось в предыдущем разделе, наряду с дальнейшим совершенствованием схем с ЭМ и ПМ, осуществляемой за счет привлечения более мощных средств НС-моделирования, представляется целесообразным также расширить перечень рассматриваемых схем адаптивного управления. Наряду с рассмотренной выше схемой на основе метода обратной задачи

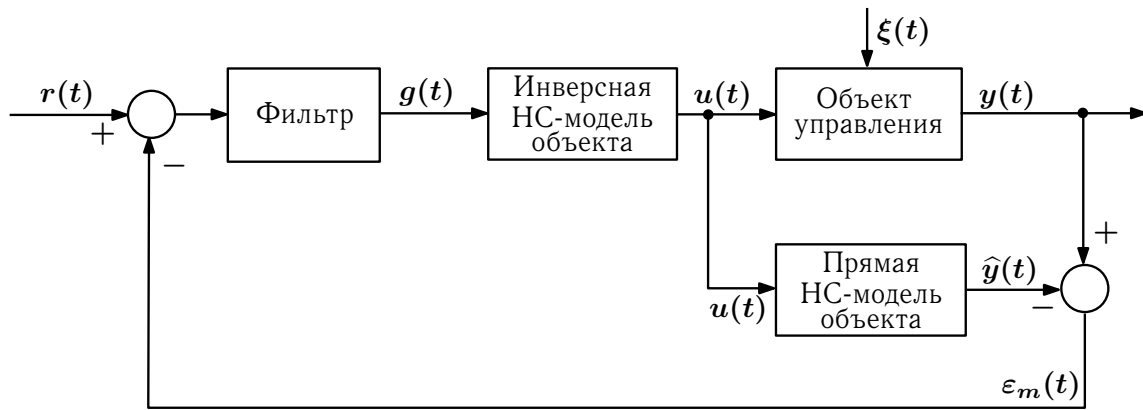


Рис. 4.12. Структурная схема системы управления с прямой и инверсной НС-моделями объекта управления: $r(t)$ — задающее воздействие; $\xi(t)$ — возмущающее воздействие; $g(t)$ — сигнал на выходе фильтра; $u(t)$ — управление; $y(t)$ — выход объекта управления; $\hat{y}(t)$ — выход нейросетевой модели объекта управления; $\varepsilon_m(t)$ — расхождение между выходами объекта управления и НС-модели

динамики, еще один из очевидных кандидатов на включение в такой расширенный перечень — это схема с инверсной моделью (ИМ) [53, 54, 82].

Схему управления с ИМ можно представить в двух вариантах: неадаптивном и адаптивном, точно так же, как это имеет место и для схем управления с ЭМ и ПМ.

2. Неадаптивная схема управления с ИМ (рис. 4.12), известная в теории управления как система с косвенным измерением возмущений, исследовалась на устойчивость и робастность при управлении нелинейными объектами и продемонстрировала возможность достижения высокого качества управления.

Помимо инверсной НС-модели, в состав систем управления с ИМ входит также и прямая НС-модель объекта управления (это точно такая же модель объекта, которая входила в состав систем управления с ЭМ и ПМ).

При соответствующем выборе операторов для элементов системы (т. е. фильтра, прямой и инверсной модели) выходная переменная $y(t)$ воспроизводит входной (задающий) сигнал $g(t)$ с желаемой динамикой и не зависит от возмущений $\xi(t)$, приведенных к выходу объекта управления.

Это возможно, если инверсная модель объекта управления реализует инверсный оператор A^{-1} , а прямая модель — прямой оператор A объекта управления. Для достижения желаемой динамики системы в ее состав вводится фильтр, реализующий оператор вида

$$A_F = \tilde{A}^{-1} \{ \hat{A} \},$$

где \tilde{A}^{-1} — реализация инверсной моделью точного инверсного оператора A^{-1} ; \tilde{A} — реализация прямой моделью точного прямого оператора A . В этом случае сигнал $r(t)$ воспроизводится на выходе объекта без искажений, т. е. $y(t) \equiv r(t)$ и не зависит от возмущения $\zeta(t)$, эквивалентного возмущению $\xi(t)$ и приведенного ко входу объекта.

Если прямая и инверсная модели объекта управления реализованы точно, т. е. если

$$\tilde{A}^{-1} = A^{-1} \quad \text{и} \quad \hat{A} = A,$$

тогда

$$y(t) = A_F \{r(t) - \varepsilon_m(t)\}.$$

Вход и выход системы в рассматриваемом случае связаны операторным соотношением

$$y(t) = L^{-1} \{A_F \{ \tilde{A}^{-1} \{ A \{ r(t) \} \} \} \},$$

$$L = 1 + A_F \{ \tilde{A}^{-1} \{ A \} \} - A_F \{ \tilde{A}^{-1} \{ \hat{A} \} \}$$

3. В системе управления с ИМ показанной на рис. 4.12, могут возникать проблемы устойчивости и робастности из-за недостаточно точной реализации прямой и инверсной моделей объекта управления. Если объект управления нелинейный и многосвязный по своей природе, как это и имеет место в случае ЛА, аналитическое решение проблемы едва ли возможно, остается лишь путь вычислительного моделирования. Как показывает имеющийся опыт [138–140], по-настоящему эффективную реализацию прямой и инверсной моделей, пригодных по уровню достижимой ими точности к использованию в составе систем управления с ИМ, обеспечивает лишь нейросетевое моделирование или же нейронечеткое моделирование, когда аппарат искусственных нейронных сетей используется совместно с аппаратом нечеткой логики [37].

Возможности системы управления со структурой, показанной на рис. 4.12, ограничиваются отсутствием обратных связей по воспроизводимой функции $r(t)$. Этого недостатка лишена схема, представленная на рис. 4.13.

Система управления, построенная по этой схеме, включает две динамически настраиваемые модели, прямую и инверсную. По указанным выше причинам обе эти модели наиболее целесообразно реализовывать в классе динамически обучаемых НС-моделей. В алгоритме обучения инверсной модели используются текущие измерения воспроизводимой (задающей) функции $r(t)$ и выхода прямой модели объекта $\hat{y}(t)$. Прямая модель выполняется также в виде динамической сети, обучаемой по прогнозу ошибки $\hat{\varepsilon}_m(t)$.

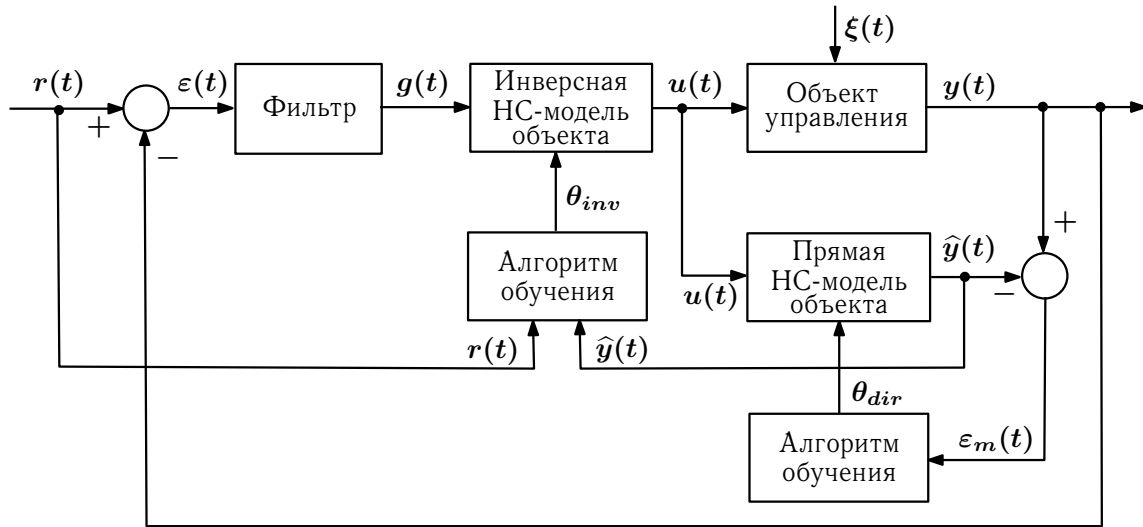


Рис. 4.13. Структурная схема адаптивной системы управления с прямой и инверсной НС-моделями объекта управления: $r(t)$ — задающее воздействие; $\xi(t)$ — возмущающее воздействие; $g(t)$ — сигнал на выходе фильтра; $u(t)$ — управление; $y(t)$ — выход объекта управления; $\hat{y}(t)$ — выход нейросетевой модели объекта управления; $\varepsilon_m(t)$ — расхождение между выходами объекта управления и НС-модели; $\varepsilon(t)$ — расхождение между выходами объекта управления и задающим воздействием; θ_{dir} и θ_{inv} — параметры прямой и инверсной модели объекта управления, соответственно

Предельные динамические свойства рассматриваемой системы вытекают из следующих рассуждений. Для обучения прямой модели ошибка определяется соотношением

$$\hat{\varepsilon}_m(t) = y(t) - \hat{y}(t) \rightarrow 0, \quad \text{при } t \geq 0,$$

так что $y(t) \rightarrow \hat{y}(t)$.

При настройке инверсной модели используется ошибка обучения

$$\hat{\varepsilon}_m(t) = g(t) - \hat{y}(t) \rightarrow 0, \quad \text{при } t \rightarrow \infty.$$

Следовательно,

$$y(t) \rightarrow g(t) \quad \text{и} \quad \hat{y}(t) \rightarrow g(t).$$

Если A — оператор объекта управления, A_c — оператор инверсной модели, а A_m — оператор прямой модели, то цепочка предельных переходов при обучении сетей

$$\hat{y}(t) \rightarrow y(t) \rightarrow g(t) \quad \text{при } t \rightarrow \infty$$

осуществляется тогда, когда

$$A_c = A_m^{-1} = A^{-1}.$$

В результате настройки, проведенной описываемым способом, получается инверсная модель объекта управления, которая выполняет в схеме изображенной на рис. 4.13, функцию адаптивного нейроконтроллера. Адаптивность этой системы проявляется в том, что для ее структурного синтеза требуется минимум априорной информации об объекте, кроме того, обе модели, прямая и инверсная (т. е. нейроконтроллер), подстраиваются непосредственно в процессе функционирования системы при неконтролируемых изменениях динамики объекта управления.

4. Схема адаптивного управления с инверсной моделью представляет безусловный интерес с точки зрения задач управления движением ЛА, если имеется возможность получить достаточно точные прямую и, особенно, инверсную модели объекта управления. При этом следует учитывать, что динамика объекта может меняться резко и непредсказуемо непосредственно в полете, в частности, из-за отказов оборудования и повреждений конструкции. Следовательно, необходимо располагать не только средствами построения достаточно точных ИМ и ПМ на этапе структурного синтеза системы управления, но и средствами оперативной корректировки этих моделей непосредственно в ходе полета.

4.3.2.2 Формирование инверсных моделей

Важнейшим элементом схем управления, рассматриваемых в данном разделе, является инверсная модель объекта управления.

В рамках проводившихся исследований было рассмотрено несколько вариантов инверсных моделей, в том числе:

1. Составная инверсная модель, которая получается из схемы с обратной динамикой размыканием обратной связи. Входами в этом случае являются значения величин, описываемых эталонной моделью, т. е. $\dot{x}_m(t)$ и $x_m(t)$.
2. Составная инверсная модель, которая получается из схемы с эталонной моделью размыканием обратной связи (этот вариант предложен в [37]). Входами здесь будут задающий сигнал $x_{ref}(t)$ и выход эталонной модели $x_m(t)$.
3. «Обыкновенный» вариант инверсной модели. На вход поступает только текущее состояние эталонной модели, и, в отличие от двух других вариантов, она не получается размыканием какого-либо контура. Входы в данном случае — это задержанный сигнал $x_m(t)$. Минимальное количество задержек — три, хотя этот вариант работал и с пятнадцатью

задержками. Преимуществом данного варианта является возможность получения инверсной модели в процессе статического обучения. Однако на практике возникает ряд проблем, описанных, например, в гл. 10 книги [63].

Сравнительный анализ этих вариантов, который осуществлялся средствами компьютерного моделирования, показал, что для случая продольного короткопериодического движения они дают практически одинаковые результаты. Вариант 3 несколько дольше обучается в сравнении с остальными двумя, однако превосходит их по точности из-за того, что при статическом обучении алгоритм работает сразу со всей обучающей выборкой. В качестве основного в дальнейшем анализе был принят первый вариант.

4.3.3 Адаптивное управление с обратной динамикой и инверсной моделью для случая углового движения ЛА

4.3.3.1 Система с обратной динамикой для управления продольным угловым движением ЛА

В данном разделе представлены результаты вычислительных экспериментов для системы с обратной динамикой, полученные для частного случая управления продольным угловым движением самолета. Случай полного (пространственного) движения будет рассмотрен ниже, в разд. 4.3.4.

Эксперименты проводились для различных режимов полета, характеризующихся скоростью и высотой полета. Высота полета была принята равной $H = 100$ м для всех рассматривавшихся случаев, скорость полета принимала значения $V = 300, 500, 700$ км/ч. Во всех случаях решалась задача отслеживания заданного значения угла атаки α . Соответствующие результаты моделирования, приведенные на рис. А.63–А.67, показывают, что качество управления, оцениваемое величиной ошибки отслеживания e_α заданного значения угла атаки α_{ref} , является вполне удовлетворительным при достаточно сложном характере изменения сигнала α_{ref} .

На рис. А.68 и А.69 показаны результаты аналогичных экспериментов для мини-БПЛА Х-04.

4.3.3.2 Система с инверсной моделью для управления продольным угловым движением ЛА

В данном разделе представлены результаты вычислительных экспериментов для системы с инверсной моделью, полученные для частного случая управления продольным короткопериодическим движением самолета.

Как и в случае системы с обратной динамикой, рассмотренном в предыдущем разделе, эксперименты проводились для различных режимов полета, характеризуемых скоростью и высотой полета. Высота полета была принята равной $H = 100$ м для всех рассматривавшихся случаев, скорость полета принимала значения $V = 300, 500, 700$ км/ч. Во всех случаях решалась задача отслеживания заданного значения угла атаки α . Соответствующие результаты моделирования, приведенные на рис. А.70–А.74, показывают, что качество управления, оцениваемое величиной ошибки отслеживания e_α заданного значения угла атаки α_{ref} , является вполне удовлетворительным при достаточно сложном характере изменения сигнала α_{ref} .

4.3.3.3 Сопоставление систем управления с обратной динамикой и с инверсной моделью для случая продольного углового движения ЛА

1. Результаты, представленные на рис. А.63–А.67 для системы с обратной динамикой (ОД) и на рис. А.70–А.74 для системы с инверсной моделью (ИМ), позволяют сделать следующие выводы.

1. Обе схемы требуют наличия компенсирующего контура.

(a) В случае с ОД система пытается подобрать управление так, чтобы скорость изменения состояния \dot{x} была приближенно равна эталонной \dot{x}_m . Из-за этой приближенности возникают ошибки, которые накапливаются во времени (интегрируются), если перейти от скорости \dot{x} к координате x .

(b) ИМ работает по тому же принципу, но управление происходит по разомкнутому циклу, поэтому стоило ожидать, что ошибка будет накапливаться быстрее.

2. На рис. А.66 и рис. А.73 сравнивается динамика ошибки для схем с ОД и с ИМ, хотя фактически можно говорить о двух вариантах одной и той же схемы, поскольку таким же переключением входа контроллера $x \rightarrow x_m$ получают инверсные модели и в других

схемах. В случае управления по разомкнутому циклу ухудшается точность в переходном режиме, но зато значительно уменьшается ошибка в установившемся режиме. Хотя нельзя явно записать уравнение ошибки для ИМ, можно качественно оценить причины наблюдаемого характера поведения ошибки слежения для схем с ИМ и ОД:

- (а) Управление без обратной связи неявно предполагает, что объект всегда находится на эталонной траектории, независимо от его текущего положения. Поэтому точность в переходном режиме несколько падает, но балансировка сохраняется и ошибка колеблется в определенных пределах. Однако это справедливо только для устойчивого объекта управления.
- (б) При построении обратной динамики по замкнутому циклу объект управления далеко не отходит от эталонной траектории, т.е. область пространства состояний, покрываемая при обучении, соответствует поведению генератора траекторий (эталонной модели). Поэтому чем дальше отходит объект от заданной траектории, тем меньше точность управления. Это дополнительная причина увеличения ошибки при отсутствии компенсирующего контура (КК).

3. Точность при включении компенсирующего контура одинакова для обоих вариантов.
4. Для неустойчивых объектов управления характер поведения системы с обратной динамикой практически не меняется, но в схеме с инверсной моделью динамика ошибки слежения становится неустойчивой, поэтому ИМ следует применять только вместе с КК.

4.3.4 Адаптивное управление с обратной динамикой и инверсной моделью для случая пространственного движения ЛА

4.3.4.1 Управление по трем каналам без использования декомпозиции

В предыдущем разделе было рассмотрено построение систем с обратной динамикой и с инверсной моделью для частного случая управления продольным движением самолета. В данном разделе полученные результаты распространяются также и на случай управления пространственным движением ЛА, когда требуется формировать не только канал тангажа, но и каналы рысканья и крена. Вначале предпринимается попытка решить данную задачу, не

прибегая к декомпозиции, т. е. к разделению системы на отдельные подсистемы-каналы (см. рис. А.75–А.77 и рис. А.78–А.80 для случаев с ОД и ИМ, соответственно).

4.3.4.2 Управление по трем каналам с использованием декомпозиции

Результаты вычислительных экспериментов, полученные для случая управления пространственным движением без использования декомпозиции показывают, что очень трудно, а иногда и невозможно получить удовлетворительные результаты при таком подходе.

В связи с этим, полная система была разделена на три отдельных канала, взаимодействующих между собой, как при решении задачи идентификации (см. рис. 4.14), так и при решении задачи собственно управления (см. рис. 4.15).

Результаты вычислительных экспериментов по обучению нейросетевой модели с использованием декомпозиции приведены на рис. А.81–А.83.

Для оценки работоспособности трехканальной системы управления было проведено несколько серий вычислительных экспериментов. Результаты, приводимые на рис. А.84–А.86, показывают, как система справляется с отслеживанием заданных изменений по углу атаки совместно с требованием поддерживать нулевое значение угла скольжения. Если имеют место последовательные перекладки руля направления, приводящие к изменению угла скольжения в диапазоне $\pm 5^{\circ}$, то получаемое поведение рассматриваемой системы будет таким, как показано на рис. А.87–А.89. Кроме того, рис. А.90–А.92 и рис. А.93–А.95 демонстрируют результаты моделирования, полученные при последовательной перекладке элеронов, приводящей изменению угловой скорости крена ± 80 град/с и ± 120 град/с, соответственно.

Между каналами рассматриваемой системы существуют перекрестные связи. Результаты, приводимые на рис. А.96–А.98, показывают, как данная система справляется с компенсацией этих связей.

Полученные результаты позволяют сделать заключение о том, что в целом рассматриваемая система обладает удовлетворительной работоспособностью. Более подробный разбор этих результатов и анализ опыта, полученного в ходе проводившихся экспериментов, проводится в следующем разделе.

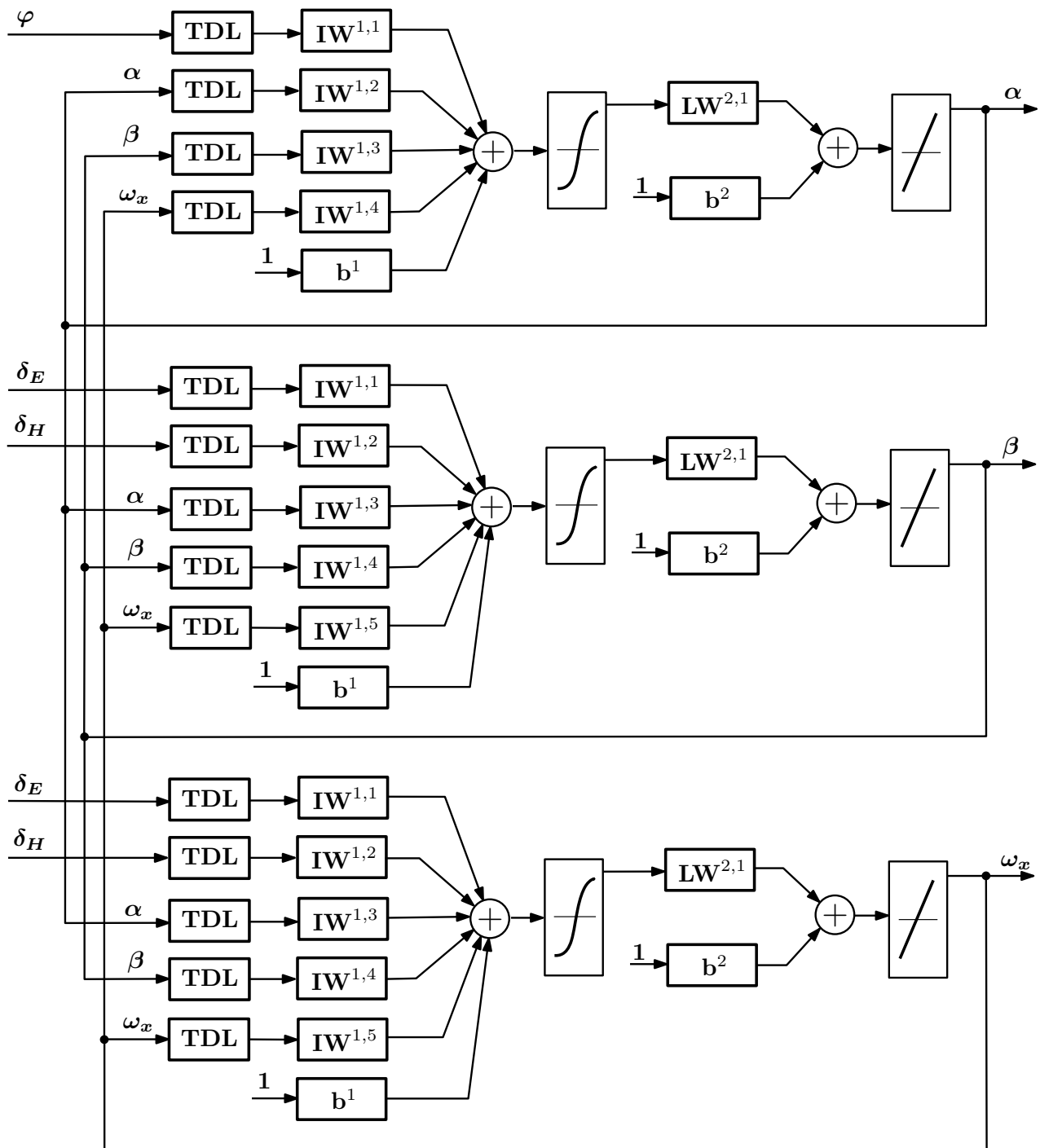


Рис. 4.14. Структурная схема нейросетевой идентификации модели движения летательного аппарата, построенной с использованием декомпозиции. Обозначения: α — угол атаки; β — угол скольжения; ω_x — угловая скорость крена; φ — командный сигнал для привода управляемого стабилизатора, град; δ_r^c — командный сигнал для привода руля направления, град; δ_a^c — командный сигнал для привода элеронов, град; TDL — элементы задержки; IW, LW — матрицы синаптических весов; b — векторы смещений

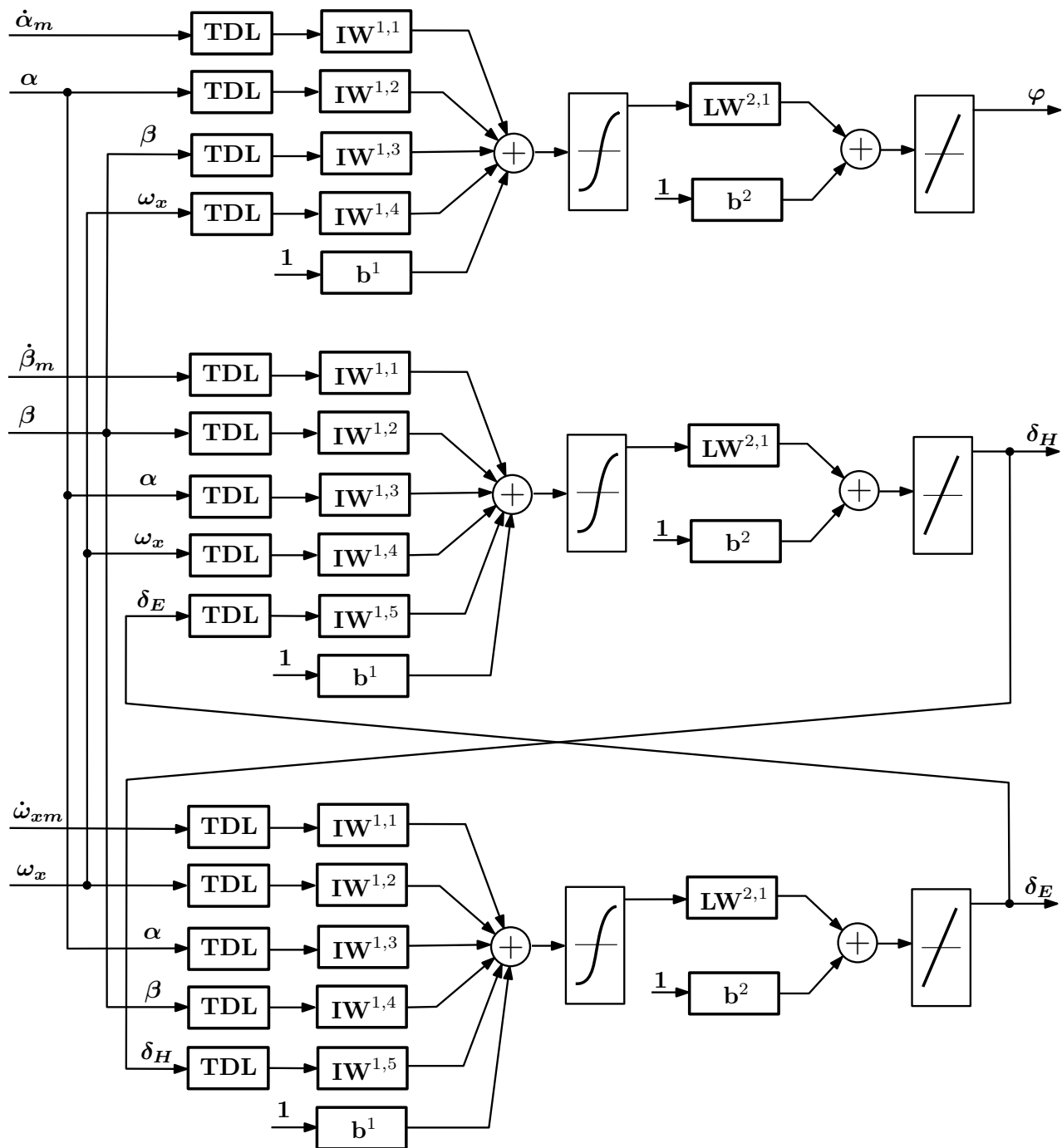


Рис. 4.15. Структурная схема нейросетевой системы управления пространственным движением летательного аппарата, построенной с использованием декомпозиции. Обозначения: α — угол атаки; β — угол скольжения; ω_x — угловая скорость крена; $\dot{\alpha}_m, \dot{\beta}_m, \dot{\omega}_{xm}$ — производные углов атаки и скольжения, угловой скорости крена, соответственно, получаемые от эталонной модели; φ — командный сигнал для привода управляемого стабилизатора, град; δ_r^c — командный сигнал для привода руля направления, град; δ_a^c — командный сигнал для привода элеронов, град; TDL — элементы задержки; IW, LW — матрицы синаптических весов; b — векторы смещений

4.3.4.3 Анализ результатов экспериментов с управлением по трем каналам

1. Ход и результаты экспериментов с построением НС-модели и контроллера пространственного движения позволяют сделать следующие выводы.

В полной модели, когда используется НС с одним скрытым слоем (3 выхода), не удается хорошо воспроизвести канал управления углом скольжения β , хотя, по-видимому, можно было бы найти такие начальные условия, при которых обучение будет успешным. При декомпозиции НС-модели на две подсети (взаимосвязанные продольный и боковой канал) и совместном обучении результат получается даже хуже, чем у простой двухслойной сети. Поскольку потенциал у обеих моделей примерно одинаковый, ухудшение точности в структурированной модели может быть обусловлено следующими факторами:

- недостаточным количеством эпох обучения, так как то же самое число эпох требует гораздо больше времени для структурированной НС-модели, чем для общей;
- свойствами алгоритма — хотя выходы подсетей зависят от своих параметров, метод Левенберга-Марквардта корректирует все параметры сразу (и, видимо, через коэффициент μ в этом алгоритме идет настройка весов по «чужим» выходам, на которые эти веса влиять не должны), однако непонятно, насколько велик этот эффект.

Альтернативный вариант декомпозиции — на три подсети вместо двух, что должно, исходя из общего смысла решаемой задачи, дать хороший эффект. Действительно, за 300 итераций процесс обучения сошелся довольно хорошо, лучше, чем при декомпозиции на две подсети и примерно также, как и без декомпозиции (следует отметить, что для сходимости β (сеть с одним выходом) нужно примерно 600–1000 итераций, которые в этом случае выполняются достаточно быстро).

Результаты экспериментов показывают, что общая (неразделенная) сеть обучается хуже по количеству необходимых эпох (затраты времени в 3–4 раза больше), чем все три подсети по отдельности. По-видимому, здесь играют роль свойства алгоритма, порождающие сильную взаимозависимость весов и выходов сети, отмеченную выше, кроме того у обученной неразделенной сети хуже обобщающие свойства, медленнее «точная» сходимость, чем у связанных вместе трех отдельных сетей.

2. В ходе проводившихся экспериментов удалось построить полную модель бокового канала (выходами которой были β и ω_x). Успешной, однако, для определенного набора данных оказывается только одна попытка из 7–10 предпринимаемых. Здесь сказываются два фактора:

наряду с начальными условиями большую роль играют особенности конкретного обучающего набора данных. Но, если обучение происходит на одном и том же наборе, то определяющим является выбор начальных условий.

Разделенная на две подсети модель бокового движения обучается отлично, т. е. разложение каждого выхода по своему базису дает огромный эффект. Процесс обучения в этом случае нормально сходится уже за 150–300 эпох.

3. Модель в пространстве состояний $(a, b, \omega_x, \omega_y, \omega_z)$ с 20 нейронами и одним скрытым слоем обучалась 2 часа. Качество настройки получилось хорошим с первой попытки и на одном наборе данных.

Обычно для модели в пространстве состояний требуется длительное обучение, но зато получается хорошее качество с одним скрытым слоем в используемой сети.

Процессы обучения для моделей по угловым скоростям ω_z, ω_y сходятся медленно и всегда проходят через плато в 300–500 эпох.

4. Процесс обучения полного контроллера (с тремя выходами — $\varphi, \delta_H, \delta_E$) в замкнутом виде не сходится. В разомкнутом виде контроллер настроить удастся, но отчетливо проявляется различная точность настройки в разных каналах — проявляется разная значимость ошибок в разных каналах, качественно можно рассматривать обучение по суммарной ошибке (по одному критерию) как обучение по нескольким критериям, свернутым в один. И хотя решение существует, такая свертка порождает побочные локальные минимумы в функции ошибок, которые не удастся преодолеть последовательному алгоритму (при обучении модели с несколькими выходами и пакетный алгоритм попадал в локальный минимум).

Частичное решение состоит в том, чтобы назначать разные уровни шумов разным каналам в фильтре Калмана в соответствии с диапазоном значений выхода канала так, чтобы относительная степень неопределенности была бы примерно одинаковой. Улучшение в таком варианте наблюдается, но до конца процесс обучения все равно не сходится.

5. Ряд проблем возник при формировании контроллера для канала крена. Не удалось добиться сходимости процесса обучения применительно к контроллеру для ω_x ни в одном из следующих двух вариантов:

- управление по состоянию,
- управление по выходу (с задержками).

Возможные причины этого заключаются в непредставительном обучающем наборе или в

недостаточной сходимости последовательного процесса обучения.

Поскольку в модели ω_x по состоянию только одна задержка на входе, стало возможным попытаться применить метод Левенберга-Марквардта на всей выборке сразу. Результат получился очень хорошим, при этом тестирование проводилось в замкнутом виде с НС-моделью.

Следовательно можно сделать вывод, что причиной недостаточной сходимости обучения в реальном времени является то, что при обучении в последовательном режиме нельзя добиться такой же высокой точности, как в пакетном, в данном случае даже при использовании динамических производных.

Однако, сравнение этих двух вариантов показывает, что вырабатываемое управление практически совпадает в обоих вариантах, а значительно бóльшие ошибки при обучении с фильтром Калмана возникают из-за характера объекта управления, в частности, очень быстрой реакции крена (ω_x) на отклонение элеронов, поэтому даже небольшие ошибки в управлении приводят к большим отклонениям от эталонной траектории. Поэтому при использовании компенсатора точность будет примерно одинакова.

Все вышесказанное относилось к управлению по состоянию. При управлении по выходу качество обучения снижается, появляются «выпавшие» интервалы с очень низкой точностью (ошибка по углу скольжения порядка 9–10 град).

Судя по всему, эти «выпавшие» интервалы вызваны особенностями НС-модели, например, плохим обобщением в некоторых регионах (причиной чего, в свою очередь, является недостаточно представительный набор данных для модели).

Дополнительные тесты показали, что причинами такой неравномерной сходимости являются:

- более сложная функция ошибки, чем при управлении по состоянию;
- ограничения, присущие последовательным методам обучения;
- большие ошибки, характерные для НС-модели в ограниченных областях.

Тесты, проведенные для случая с инверсной моделью, показывают, что получаемой системе присущи те же самые особенности (выпавшие сегменты), что и замкнутой системе с обратной динамикой. Это может говорить о недостаточной «гладкости» НС-модели по выходу.

6. Анализ результатов, полученных в ходе вычислительных экспериментов, описанных выше, позволяет сделать следующие выводы:

1. При анализе возможностей компенсации перекрестных связей по углу скольжения видно, что движение рыскания является неуправляемым и возбуждается перекрестными связями. Видно, что точность компенсации не везде высокая, что обусловлено двумя причинами:

- (a) Угол скольжения β выходит из области, в которой обучались контроллеры.
- (b) В ряде случаев не хватает рулей для точного следования эталонной траектории (например, на больших углах атаки и при значительных углах скольжения момент m_x^β требует почти полного отклонения элеронов для его парирования).

2. В ряде случаев неудовлетворительным является качество управления по углу скольжения. Задающий сигнал по углу скольжения нулевой $\beta_{ref} = 0$, но качество управления иногда недостаточное. Здесь сказываются несколько причин:

- (a) Недостаточная эффективность руля направления для компенсации всех перекрестных связей (видно, что они усиливаются на больших положительных и отрицательных углах атаки).
- (b) Попытка заставить работать систему в более широкой области пространства состояний, чем она обучалась, либо просто недостаточное покрытие пространства состояний.
- (c) Слишком большой коэффициент усиления в компенсаторе для β , что приводит к релейному эффекту из-за ограниченного диапазона углов для руля направления (недостаточная устойчивость в большом).
- (d) Для разомкнутого НС-управления (инверсная модель) может проявляться взаимодействие различных каналов компенсатора при ограничении руля по одному из каналов и на определенных частотах задающего сигнала.
- (e) Для замкнутого НС-управления (обратная динамика) существенной, в основном, является недостаточная эффективность руля. Если руль направления становится на упоры при частых переключках $\omega_{x ref}$, то величина ошибки по β становится большой. Ошибки по β и по ω_x с замкнутым контроллером и компенсатором оказываются почти не взаимосвязаны.

Плохое качество работы контроллера проявляется близко к границе диапазона обучения. Скорее всего некоторые области пространства состояний остались незаполненными при

обучении контроллера или модели, либо плотность данных недостаточна для хорошего обобщения. В частности, выбросов по β значительно больше, когда $\beta_{ref} = 0$, чем при случайном задающем сигнале по β , потому что обучение специально при $\beta = 0$ не проводилось, а при $\beta = 0$ ошибки обобщения становятся более заметными.

Главными причинами не всегда достаточного качества управления в канале рыскания являются:

1. Недостаточная эффективность руля направления в сравнении с моментами от перекрестных связей. Некоторые перекрестные связи слишком велики, чтобы их устранить.
2. Недостаточные обобщающие свойства нейроконтроллера в некоторых областях пространства состояний (проявляется в значительно меньшей степени, чем п.1), поскольку управляемая система является многомерной.

В процессе обучения многомерной системы (модели или системы управления) на передний план выступает проблема данных (обучающих наборов), то есть становится проблематично вместить все вариации поведения системы в один фиксированный набор данных. Для одномерных систем эта проблема не является настолько критичной, как это следует из эксперимента.

Кроме того, в проводившихся экспериментах обучение НС-модели происходило в основном в пакетном режиме после генерации данных. Помимо недостаточного объема данных играет роль забывание в многослойном персептроне. В данном случае уже реализовано (применительно к используемой архитектуре сетей) обучение нейросетевой модели и контроллера в режиме реального времени, с использованием новых поступающих данных для адаптации к изменению динамических свойств объекта управления.

Обучение в последовательном режиме имеет гораздо менее жесткие ограничения на объем обучающих данных (если этот обучающий набор уже есть и фиксирован). Данные генерируются случайным образом (сигнал на рули или заданное движение), поэтому покрытие пространства состояний может быть неравномерным и неполным (особенно в многомерных системах) внутри одной выборки. Можно использовать несколько (достаточно много) разных наборов данных, последовательно используя их для обучения (обучение с фильтром Калмана позволяет это сделать).

Самый простой вариант — использовать обучение в реальном времени (для модели и контроллера), запуская на исполнение математическую модель с включенной адаптацией до тех

пор, пока не будет достигнута приемлемая точность во всем пространстве состояний. То есть имеет смысл последовательно уточнять НС-модель и закон управления именно в тех областях пространства состояний, где фактически работает система.

Однако, в силу недостатков алгоритмов адаптации, необходимо наличие предобученной НС-модели и контроллера. Это связано с тем, что для адаптации нелинейной системы в данном случае используются линейризованные алгоритмы (фильтр Калмана), которые хорошо сходятся только в окрестности минимума функции ошибки (когда имеет право на существование сама линейризация), то есть в данном случае адаптация ограничена дообучением, а обучение с нуля в режиме адаптации дает неприемлемые результаты, особенно для НС-модели.

4.4 Адаптивное управление угловым движением ЛА в условиях неопределенности

4.4.1 Влияние атмосферной турбулентности на работоспособность адаптивной системы управления продольным движением самолета

Один из вопросов, традиционно сложных для адаптивных систем управления, состоит в их способности противостоять внешним возмущающим воздействиям случайного характера.

В данном разделе предпринимается попытка оценить, вначале применительно к адаптивным системам с эталонной моделью, насколько хорошо синтезированная система способна справиться с возмущениями, в том числе и при возникновении отказных ситуаций.

В качестве модели атмосферной турбулентности использовалась известная модель Драйдена, в том виде, как она описана в стандарте MIL-F-8785C и реализована в наборе Simulink-модулей Aerospace Toolbox для системы Matlab. Воздействие турбулентности проявляется через добавочные компоненты для вертикальной скорости V_y и угловой скорости тангажа ω_z .

Все результаты были получены для маневренного самолета F-16, для режима полета $H = 100$ м и $V = 600$ км/ч. Они представлены на рис. А.99–А.110.

При этом рассмотрены два случая:

1. Возмущающее воздействие $\Delta V_{y,turb}$ в диапазоне ± 10 м/с, $\Delta \omega_{z,turb}$ в диапазоне ± 0.2 град/с (рис. А.99–А.104).
2. Возмущающее воздействие: $\Delta V_{y,turb}$ в диапазоне ± 20 м/с, $\Delta \omega_{z,turb}$ в диапазоне ± 2 град/с (рис. А.105–А.110).

Для того, чтобы оценить влияние атмосферной турбулентности на характеристики управляемой системы, для каждого из рассматриваемых случаев просчитывалось два варианта: с

воздействием турбулентности и без нее.

Во всех вариантах рассматривалось поведение системы при последовательном появлении двух отказных ситуаций, введенных таким же образом, как это было сделано в разд. 4.1.5. Первая из них ($t = 20$ с) проявляется в смещении центровки на +10%, вторая ($t = 50$ с) — в уменьшении на 50% эффективности органа продольного управления самолетом.

4.4.2 Адаптация к неопределенности в исходных данных

1. Наличие неопределенностей в исходных данных приводит к тому, что нейросетевая модель и нейроконтроллер будут настроены неточно и эта неточность оказывает отрицательное влияние на качество управления.

В проведенных вычислительных экспериментах неточное знание динамики имитировалось тем, что на начало моделирования задавались НС-модель или НК, настроенные на другой режим полета. Таким образом, синтез закона управления осуществлялся применительно к одному режиму полета (число Маха и высота полета), а затем закон управления начинал работать совсем в других условиях. Соответствие режима полета, для которого в проведенных вычислительных экспериментах осуществлялся синтез закона управления и режима, в котором этот закон управления тестировался, задается табл. 1.

Далее рассматривались два варианта действий.

Согласно первому варианту, включался механизм адаптации (его основные особенности были рассмотрены в разд. 1.7), который позволял осуществить корректировку закона управления применительно к тем условиям работы, в которых он оказался.

Второй подход состоял в том, чтобы оценить значимость введенных механизмов адаптации путем выявления их вклада в общую задачу обеспечения требуемого качества управления при меняющихся условиях функционирования. Для реализации этого подхода механизмы адаптации отключались, то есть корректировка закона управления не осуществлялась, а задача обеспечения качества управления системой целиком ложилась на механизмы обеспечения робастности, включающие компенсирующий контур, введенный в изучаемые системы на основе соображений, изложенных выше, в разд. 4.1.3.

Результаты вычислительных экспериментов в рамках первого из вариантов, перечисленных выше, представлены в следующих двух разделах (4.4.2.1 и 4.4.2.2). Второй подход анализируется в экспериментах, результаты которых приводятся ниже, в разд. 4.4.3.

При рассмотрении первого из двух анализируемых вариантов фактически требуется решить для объекта управления задачу идентификации в реальном масштабе времени с тем, чтобы располагать достоверной информацией о характере влияния отклонения органов управления на поведение объекта управления.

Для сходимости процесса идентификации в реальном времени требуется наличие на входе системы тестового (нецелевого) сигнала в течение некоторого времени (в рассматриваемом случае это время адаптации системы). Для настройки НС-модели таким сигналом может быть некоторый дополнительный сигнал на привод. Однако для нелинейных систем условием хорошей настройки является также учет как можно большего числа состояний из области в пространстве состояний, в которой функционирует система. Поэтому тестовый сигнал для НС-модели вводится не по отклонению руля, а по задающему воздействию, определенному именно в пространстве состояний, пропущенному через систему управления. В нейроконтроллере входным сигналом является непосредственно задающее воздействие, то есть для настройки контроллера в рассматриваемом случае тестовый сигнал вводится как изменяющийся задающий сигнал по углу атаки.

Таким образом, отдельные тестовые сигналы в виде отклонения органов управления (элевонов в рассматриваемом случае перспективного ГЗЛА) в реализуемом подходе не требуются. Это, однако, не избавляет от необходимости во внешних возмущающих воздействиях на объект управления, с помощью которых выявляется информация о реакциях данного объекта на управляющие воздействия. На этой информации основывается адаптационная корректировка используемых законов управления. Один из видов таких воздействий — это задающий сигнал по углу атаки, используемый в проводимых исследованиях. Как это было показано в разд. 2.3, данный сигнал должен быть таким, чтобы возможно полнее покрыть пространство состояний исследуемой системы, а также возможно полнее учесть ее динамику путем разнообразных по темпу переходов между отдельными состояниями. На большом объеме вычислительных экспериментов была показана работоспособность такого подхода.

Во всех вычислительных экспериментах, результаты которых приведены в разд. 4.4.2.1 (для схемы с ЭМ) и в разд. 4.4.2.1 (для схемы с ПМ) работа системы управления задавалась одним и тем же образом. Первые 20 с на вход системы управления подавался возмущающий задающий сигнал, который необходим для решения задачи идентификации, затем следует участок тестирования системы (его продолжительность также составляет 20 с) последовательностью ступенчатых воздействий, удаленных по времени одно от другого в степени,

ТАБЛИЦА 4.1. Таблица соответствия режимов полета, использованных при синтезе и тестировании нейроконтроллера для перспективного ГЗЛА

Режим полета, для которого проводилось тестирование НК	Режим полета, для которого проводился синтез НК
$M = 5, H = 32$ км	$M = 7, H = 30$ км
$M = 5, H = 30$ км	$M = 7, H = 28$ км
$M = 6, H = 30$ км	$M = 5, H = 28$ км
$M = 7, H = 30$ км	$M = 5, H = 32$ км
$M = 7, H = 28$ км	$M = 6, H = 32$ км
$M = 7, H = 32$ км	$M = 5, H = 28$ км
$M = 6, H = 32$ км	$M = 7, H = 28$ км
$M = 6, H = 28$ км	$M = 7, H = 32$ км
$M = 5, H = 28$ км	$M = 7, H = 30$ км

достаточной для затухания переходного процесса, вызванного ступенчатым воздействием до момента, когда будет подано следующее воздействие.

4.4.2.1 Адаптивное управление с эталонной моделью

В данном подразделе на рис. А.111–А.119 приведены данные вычислительного эксперимента по оценке влияния точности исходных данных на характеристики управления применительно к системе с эталонной моделью.

Как уже отмечалось выше, результаты вычислительных экспериментов, представленные на рис. А.111–А.119, демонстрируют работу системы управления на двух участках продолжительностью по 20 с каждый: на первом из них осуществляется адаптация закона управления путем подачи на вход системы управления возмущающего воздействия (активно изменяющегося задающего сигнала по углу атаки); на втором участке с такой же продолжительностью выполняется тестирование системы с помощью последовательности ступенчатых входных сигналов, разнесенных по времени так, чтобы возмущенное движение, вызванное поданным сигналом, успело бы затухнуть до момента подачи следующего сигнала. Если это условие вы-

полняется, то можно считать, что система в одном сеансе тестируется набором независимых друг от друга ступенчатых входных сигналов, различающихся по величине.

Величина собственной частоты ЭМ в данных экспериментах была выбрана, по причинам, изложенным в разд. , равной 2 1/с .

Анализ данных, приводимых на рис. А.111–А.119, позволяет сделать следующие выводы. Качество регулирования, оцениваемое по величине ошибки отслеживания заданного угла атаки, достигает в большинстве случаев приемлемых значений уже в течение 3–5 с процесса подстройки закона управления под изменившиеся условия полета. Таким образом, участок адаптации системы управления протяженностью в 20 с явно избыточен, его можно существенно сократить, как правило, до нескольких секунд.

Рассматриваемые механизмы адаптации обеспечивают достаточно высокую точность решения задачи управления: ошибка отслеживания заданного угла атаки не превышает, как правило, значений в диапазоне от ± 0.25 град. до ± 0.45 град., причем это касается как схемы с ЭМ, так и схемы с ПМ. В установившихся режимах при тестировании (когда затухало возмущение от ступенчатого воздействия) ошибка отслеживания угла атаки становилась практически нулевой. При этом начальная ошибка (в момент включения механизма адаптации) в ряде случаев доходила до ± 1 град. Как видно из табл. 4.1, разрыв между режимами полета, при которых синтезировался и тестировался закон управления, был достаточно велик: до 2 единиц по числу Маха, до 4 км по высоте полета. Тем не менее, механизмы адаптации (как в схеме с ПМ, так и в схеме с ЭМ) практически во всех случаях успешно справлялись с задачей подстройки закона управления под изменившиеся условия полета, восстанавливая точность отслеживания задаваемого угла атаки.

4.4.2.2 Адаптивное управление с прогнозирующей моделью

Результаты вычислительных экспериментов по оценке влияния неточности в исходной модели применительно к системе адаптивного управления с прогнозирующей моделью приведены на рис. А.120–А.128.

4.4.3 Оценка значимости механизмов адаптации в задаче управления угловым движением ЛА

1. Нейросетевая система управления в вариантах, исследуемых в данной работе, состоит из двух частей: собственно нейросетевой и дополнительного компенсирующего контура. Эти две части являются самостоятельными, каждая из них выполняет свои определенные функции. Нейросетевая часть является адаптивной, ее задачей является выявление динамических свойств объекта для их «сокращения» (в том числе и нелинейностей) и введение в систему желаемой динамики. Компенсирующая часть уменьшает ошибки, возникающие при неточном «сокращении» и придает системе свойство робастности, парируя отклонения от эталонной траектории независимо от того, чем они вызваны.

Как было показано в разд. 4.1.3, наличие компенсирующей части принципиально важно для схем адаптивного управления как с ЭМ, так и с ПМ. В этих схемах основная идея состоит в том, что в качестве источника информации о поведении объекта управления при формировании значений параметров регулятора используется нейросетевая модель (НС-модель) этого объекта. В силу приближенного характера НС-модели, результаты, получаемые с ее помощью, неизбежно отличаются от реальных значений величин, описывающих движение объекта. Подход к компенсации возникающей по этой причине ошибки был предложен в разд. 4.1.3. Он состоит в том, чтобы неточность НС-модели трактовать как возмущающее воздействие на систему, приводящее к отклонению траектории реального объекта от эталонной траектории, парируя это воздействие путем введения в систему компенсирующего контура (компенсатора). Как было показано в разд. 4.1.3, введение компенсирующего контура в состав адаптивной системы повышает ее робастность, без которой алгоритм адаптации может не уложиться с корректировкой закона управления в то временное окно, в пределах которого ситуация не успевает перерасти из аварийной в катастрофическую. В ряде случаев, когда изменения динамики объекта управления и/или условий его функционирования несущественны, может оказаться вполне достаточно только робастности системы без привлечения адаптационных механизмов. В связи с этим возникает вопрос о значимости механизмов адаптации, что они дают в сравнении с вариантом системы без адаптации, но с робастностью, улучшенной за счет использования компенсирующего контура.

2. Для оценки вклада адаптивности и робастности в поведение системы были проведены эксперименты, в которых отключались механизмы адаптации. Как и в предыдущем случае

(анализ влияния неопределенностей в исходных данных на свойства синтезируемой системы), система была изначально настроена неправильно. В предыдущей серии экспериментов, описанных в разд. 4.4.2, система управления адаптировалась к изменившимся условиям функционирования объекта управления, именно для этого в экспериментах, результаты которых показаны на рис. А.111–А.119 (схема с ЭМ) и рис. А.120–А.128 (схема с ПМ), до 20 с идет возмущающий задающий сигнал, требуемый для подстройки закона управления, а затем уже тестовый ступенчатый сигнал для оценки качества подстройки. В серии экспериментов, результаты которых представлены ниже, система управления продолжала функционировать с той настройкой («неправильной»), которая была получена первоначально, без осуществления адаптационной подстройки, только за счет свойств робастности (см. рис. А.129–А.131 для системы с ПМ и рис. А.132–А.134 для системы с ЭМ).

3. В проведенных экспериментах, результаты которых представлены на рис. А.129–А.134, время функционирования системы делилось, как и в предыдущем случае, на два отрезка продолжительностью по 20 с каждый. Характер задающего воздействия по углу атаки такой же, как и в предыдущем случае, т. е. первые 20 с задающий сигнал представляет собой случайную последовательность часто изменяющихся значений угла атаки. При таком сигнале возмущение от поданного входного сигнала не успевает затухнуть, как на объект действует уже следующий сигнал. Такого рода сложный сигнал позволяет оценить кумулятивное воздействие нескольких возмущений, реакции от воздействия которых пересекаются по времени, т. е. позволяет осуществить тестирование системы управления в достаточно жестких условиях. В отличие от предыдущего случая, корректировка закона управления не проводилась, т. е. адаптационный механизм был отключен.

На втором временном отрезке система управления тестируется с помощью более традиционного входного сигнала, позволяющего оценить характер поведения системы при изолированных ступенчатых воздействиях.

Результаты показывают, что неточность настройки не является критической для устойчивости системы, но вызывает значительные отклонения от эталонной траектории (по рассматриваемому каналу). В установившемся режиме эти отклонения могут быть сведены к нулю с помощью интегрального компенсатора, однако в переходных режимах (при работе на больших частотах) с этим справиться не удастся и единственная возможность повысить точность — включить механизмы адаптации вплоть до уменьшения ошибки ниже определенного уровня.

Как видно из результатов, представленных на рис. А.129–А.134, отсутствие механизмов адаптации приводит к резкому ухудшению качества управления. А именно, ошибка отслеживания задающего сигнала по углу атаки лежит теперь в пределах $\pm 2^\circ$, а в ряде случаев — даже и в пределах $\pm 4^\circ$.

Таким образом, механизмы адаптации позволяют снизить ошибку слежения в значительно большей области пространства состояний (в случае нелинейной системы) и расширить полосу частот, в которой ошибка слежения не превышает некоторой заданной величины. То есть наделение системы адаптивными свойствами позволяет ей бороться с гораздо более широким классом параметрических неопределенностей объектов управления.

4. Общий вывод, который позволяют сделать результаты, полученные в разд. 4.4, состоит в том, что методы адаптивно-робастного моделирования и управления в вариантах с эталонной и прогнозирующей моделью, а также основанные на использовании метода обратной динамики и инверсной модели объекта управления, являются мощным и перспективным инструментом, позволяющим решать поставленные задачи отказоустойчивого управления движением ЛА в условиях неопределенности.

5 Нейросетевые полуэмпирические модели управляемых динамических систем

В разд. 3 и 4 было показано, что эмпирические НС-модели, т. е. НС-модели традиционного типа, реализующие концепцию «черный ящик», имеют серьезные ограничения по уровню сложности решаемых задач, порожденные размерностью таких моделей. В связи с этим возникает проблема снижения размерности создаваемой НС-модели таким образом, чтобы не ухудшить ее гибкость. Эту проблему предлагается решать в классе модульных полуэмпирических динамических моделей, объединяющих возможности теоретического и нейросетевого моделирования. Рассматривается подход к формированию таких моделей, а также процесс их поэтапной корректировки с целью получения требуемых характеристик точности моделирования. Дается сопоставление этих характеристик для полуэмпирических и эмпирических моделей, подтверждающее перспективность предлагаемого класса моделей.

5.1 Соотношение между эмпирическими и полуэмпирическими моделями динамических систем

В разд. 3 было рассмотрено формирование эмпирических моделей (моделей типа «черный ящик»), основываясь только на экспериментальных данных наблюдения за поведением моделируемой системы. Этот подход типичен для традиционного нейросетевого моделирования. Он может оказаться весьма полезным, а в ряде случаев и единственно возможным, если отсутствует какое-либо априорное знание о природе моделируемой системы, механизмах ее функционирования, особенностях поведения. Однако нередкой является ситуация, когда такого рода знания имеются в той или иной форме. В частности, имеются многочисленные модели движения для объектов различного вида (летательные аппараты, корабли, автомобили и т. п.), основанные на законах механики, а в ряде случаев — и на законах из других областей науки, например, моделирование движения летательных аппаратов с большими сверхзвуковыми скоростями, когда существенную роль начинают играть тепловые явления, заставляют включать в модель движения не только соотношения, основанные на законах механики, но и соотношения из области термодинамики и теплопередачи.

Модели такого рода, особенно полученные непосредственно из фундаментальных законов природы («из первых принципов»), играют важнейшую роль во всех областях науки и техники. Формирование таких моделей, однако, сопряжено с серьезными трудностями, для

их получения требуется располагать значительным объемом знаний о моделируемом объекте, что далеко не всегда возможно. Кроме того, если даже такая модель существует, например, модель движения летательного аппарата, совсем не обязательно, что она окажется пригодной для решения некоторой конкретной поставленной задачи. Во-первых, эта модель может содержать величины и зависимости, в значениях которых имеются существенные неопределенности, что, соответственно, препятствует получению точного и надежного решения. Во-вторых, даже если модель сформирована полностью и факторы неопределенности в ней отсутствуют, она может оказаться непригодной для решения реальных прикладных задач. Например, если требуется моделировать движение некоторого объекта в реальном масштабе времени с высокой точностью, то традиционная модель движения в виде системы дифференциальных уравнений (обыкновенных или в частных производных), решение которой осуществляется с помощью соответствующих методов численного интегрирования, может потребовать недопустимо большого времени на получение решения.

Чтобы преодолеть указанные затруднения, формируются, как было показано выше, приближенные эмпирические модели, одним из наиболее эффективных способов получения которых является нейросетевой подход.

Модели, называемые *теоретическими* («белый ящик»), прямо противоположны по принципам своего формирования чисто эмпирическим моделям («черный ящик»). Эмпирические данные вовлекаются в процесс получения теоретической модели лишь опосредованно как источник сведений о системе, характере ее поведения, что дает возможность выбрать соответствующий класс соотношений, описывающих это поведение, но непосредственно эти эмпирические данные при формировании этих соотношений не используются. В противоположность этому, *эмпирические* модели основываются исключительно на экспериментальных данных, они формируются таким образом, чтобы отвечать этим данным наилучшим образом, т. е. воспроизводить их с наименьшей ошибкой.

Эмпирические модели, позволяя преодолеть затруднения, связанные с теоретическими моделями, порождают, в свою очередь, новые затруднения, которых не было в моделях теоретического типа. В частности, обучение (настройка) этих моделей требует наличия соответствующих обучающих наборов данных, получение которых может представлять собой очень непростую задачу. В случае, когда о моделируемом объекте нет никаких данных, кроме экспериментальных, характеризующих его поведение, ничего не остается, кроме как пытаться получить для него эмпирическую модель. Но в случае, когда помимо эмпирических дан-

ных имеются еще и какие-то знания о нем, например, в виде уравнений движения, пусть и с неопределенными факторами в них, необходимо пытаться использовать теоретические данные как дополняющие имеющиеся экспериментальные данные. Такого рода комбинированный, компромиссный, подход можно назвать *полуэмпирическим* моделированием («серый ящик») [76, 149–162]. В сравнении с чисто теоретическим подходом его применение позволяет повысить точность моделирования за счет того, что негативное влияние элементов теоретической модели, не поддающихся адекватному описанию из-за недостатка соответствующих знаний, можно компенсировать переводом этой модели в полуэмпирическую форму и уточнением ее обучением на доступных экспериментальных данных. Применительно к чисто эмпирическим моделям учет имеющихся теоретических знаний путем перехода к полуэмпирическим моделям позволяет упростить процесс формирования моделей, удовлетворяющих заданным требованиям, а также, что весьма немаловажно, сократить объем экспериментальных данных, требуемых для обучения модели. При этом, чем больше объем привлекаемых теоретических знаний, тем меньше будет объем требуемых экспериментальных данных.

5.2 Общая схема процесса формирования полуэмпирических НС-моделей

Предлагаемый подход состоит в сочетании теоретических знаний о моделируемой динамической системе со структурными преобразованиями и обучением теоретической модели, преобразованной в нейросетевую форму, для ее улучшения.

Учитываются теоретические знания двух видов: об объекте моделирования («физика объекта») и о вычислительных методах анализа моделей. Уточнение модели выполняется средствами нейросетевого обучения. В итоге формируется динамическая модель нейросетевого типа (НС-модель), в архитектуре которой учитываются имеющиеся знания об объекте моделирования. Традиционные нейросетевые модели, в частности, рассмотренные в разд. ??, имеют чисто эмпирический характер («черный ящик»), они основываются только на экспериментальных данных о поведении системы [144]. Рассматриваемые далее динамические модульные сети, отражающие как экспериментальные данные, так имеющиеся теоретические знания, можно классифицировать как полуэмпирические модели (модели типа «серый ящик») [76, 160].

Формирование динамических сетей с модульной архитектурой в виде полуэмпирических НС-моделей состоит из следующих этапов [186, 204]:

- 1) формирование теоретической модели с непрерывным временем для исследуемой ди-

намической системы, сбор доступных экспериментальных данных о поведении этой системы;

- 2) оценка точности теоретической модели динамической системы на доступных данных, в случае недостаточной ее точности выдвижение гипотез о причинах этого и возможных путях их устранения;
- 3) преобразование исходной модели с непрерывным временем в модель с дискретным временем;
- 4) формирование нейросетевого представления для полученной модели с дискретным временем;
- 5) обучение нейросетевой модели;
- 6) оценка точности обученной нейросетевой модели;
- 7) корректировка, в случае недостаточной точности, нейросетевой модели путем внесения в нее структурных изменений.

5.3 Основные элементы процесса формирования полуэмпирических НС-моделей

5.3.1 Преобразование исходной теоретической модели в конечно-разностную форму

Первый этап процесса формирования полуэмпирической модели, как уже отмечалось, состоит в дискретизации исходной теоретической модели, которая обычно представляет собой систему дифференциальных уравнений, т. е. модель с непрерывным временем. Эта операция необходима для получения динамической системы с дискретным временем, используется для построения соответствующей рекуррентной нейронной сети. Выбор метода дискретизации [38, 167–169] играет важную роль, поскольку последствия такого выбора сказываются на устойчивости получаемой модели с дискретным временем.

5.3.1.1 Явные и неявные схемы дискретизации

Рассмотрим обыкновенное дифференциальное уравнение первого порядка вида:

$$\frac{dx(t)}{dt} = f[x(t)]. \quad (5.1)$$

Использование явной схемы дискретизации дает следующий результат:

$$x[(k + 1)T] = \varphi[x(kT), T]. \quad (5.2)$$

Здесь T — шаг дискретизации (интегрирования), который обычно совпадает с периодом выборки экспериментальных данных; k — положительное целое число; функция φ определяется конкретным используемым методом дискретизации.

Неявная схема дискретизации для того же самого уравнения (5.1) приводит к соотношению вида:

$$x[(k+1)T] = \psi[x[(k+1)T], x(kT), T]. \quad (5.3)$$

Основное различие между (5.2) и (5.3) состоит в том, что величина $x[(k+1)T]$ при использовании явной схемы появляется только в левой части соотношения (5.2), тогда как для неявной схемы — в обеих частях соотношения (5.3). Отсюда следует, что если формируется одношаговый предиктор для состояния x , вычисление $x[(k+1)T]$ по известному значению $x(kT)$ является тривиальной задачей при использовании явной схемы, но в случае неявной схемы требует решения нелинейного уравнения.

Рассмотрим более общий случай, когда исследуемая система описывается обыкновенным дифференциальным уравнением в векторной форме:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t)]. \quad (5.4)$$

При использовании явной схемы дискретизации получим:

$$K[\mathbf{x}(kT)]\mathbf{x}[(k+1)T] + \Psi[\mathbf{x}(kT), \mathbf{u}(kT), T] = 0, \quad (5.5)$$

где K — матрица, а Ψ — вектор-функция. Если же применить неявную схему дискретизации, результат будет следующим:

$$K[\mathbf{x}[(k+1)T]]\mathbf{x}[(k+1)T] + \Psi[\mathbf{x}[(k+1)T], \mathbf{x}(kT), \mathbf{u}[(k+1)T], T] = 0. \quad (5.6)$$

В данном случае опять, как и в скалярном варианте, получение $\mathbf{x}[(k+1)T]$ по значению $\mathbf{x}(kT)$ — тривиальная задача, если предположить, что матрица K имеет обратную:

$$\mathbf{x}[(k+1)T] = -K^{-1}[\mathbf{x}(kT)]\Psi[\mathbf{x}(kT), \mathbf{u}(kT), T], \quad (5.7)$$

тогда как при использовании неявной схемы необходимо будет решать систему нелинейных уравнений.

Рассмотрим опять обыкновенное дифференциальное уравнение первого порядка вида (5.1). Конкретизируем для него в качестве примера приведенные выше соотношения применительно к схеме Эйлера в явной и неявной форме.

Схема Эйлера в явной форме основана на предположении, что функция $f(\cdot)$ постоянна и ее значение равно $f[x(kT)]$ между моментами времени kT и $(k+1)T$, что приводит к следующему соотношению:

$$x[(k+1)T] = x(kT) + Tf[x(kT)]. \quad (5.8)$$

Схема Эйлера в неявной форме предполагает, что функция $f(\cdot)$ остается постоянной, а ее значение между моментами времени kT и $(k+1)T$ равно $f[x((k+1)T)]$, что дает соотношение вида:

$$x[(k+1)T] = x(kT) + Tf[x((k+1)T)]. \quad (5.9)$$

5.3.1.2 Влияние схемы дискретизации на устойчивость модели

Как видно из предыдущего раздела, использование явной схемы дискретизации значительно упрощает процесс получения полуэмпирической модели с дискретным временем по сравнению с неявной схемой. В то же время, неявные схемы часто оказываются более предпочтительными, чем явные схемы с точки зрения устойчивости получаемой модели.

Проиллюстрировать данное положение можно на простом примере. Пусть исследуемый объект описывается обыкновенным дифференциальным уравнением первого порядка следующего вида:

$$\frac{du(t)}{dt} = -\alpha u(t), \quad \alpha > 0. \quad (5.10)$$

Явный метод Эйлера дает дискретизированную модель в форме

$$\frac{u[(k+1)T] - u(kT)}{T} = -\alpha u(kT), \quad (5.11)$$

или, что то же самое,

$$u[(k+1)T] = (1 - \alpha T)u(kT). \quad (5.12)$$

Из (5.12) видно, что $u[(k+1)T]$ вычисляется рекурсивно, исходя из заданного стартового значения $u(0)$, причем эта рекурсия сходится, только если величина $(1 - \alpha T)$ имеет значение меньше 1, или, что то же самое, $T < 2/\alpha$. Время, требуемое для численного интегрирования данного уравнения, пропорционально величине $1/\alpha$. Если значение этой величины достаточно велико, численное интегрирование может оказаться невозможным, так как потребуется нереализуемо малое значение шага интегрирования T .

Рассмотрим теперь для того же самого исходного уравнения (5.10) применение неявной схемы Эйлера. В результате получим:

$$\frac{u[(k+1)T] - u(kT)}{T} = -\alpha u[(k+1)T], \quad (5.13)$$

или, что то же самое,

$$u[(k+1)T] = \frac{1}{(1 + \alpha T)} u(kT). \quad (5.14)$$

В силу того, что знаменатель в правой части соотношения (5.14) всегда будет иметь значение больше 1, процесс вычисления величины $u[(k+1)T]$ сойдется вне зависимости от значения, принимаемого параметром α .

Такое обеспечение сходимости вычислительного процесса, однако, приводит к необходимости решать нелинейное уравнение или систему таких уравнений, что не требуется при использовании явной схемы дискретизации. Из сказанного следует, что принятие той или иной схемы дискретизации для перехода от исходной теоретической модели к модели с дискретным временем оказывает существенное влияние на вид получаемой в итоге нейросетевой модели.

5.3.1.3 Влияние схемы дискретизации на обучение модели

Дискретизация с помощью явных схем для исходных теоретических моделей приводит к моделям, которые непосредственно пригодны к представлению их в нейросетевой форме. А именно, дискретная динамическая система, отвечающая исходной системе (теоретической модели) с непрерывным временем (5.4), имеет вид (5.7):

$$\mathbf{x}[(k+1)T] = -K^{-1}[\mathbf{x}(kT)]\Psi[\mathbf{x}(kT), \mathbf{u}(kT), T].$$

Данная система соответствует канонической форме рекуррентной нейронной сети, рассматриваемой в следующем разделе. Для такой формы НС-модели ядром является слоистая сеть прямого распространения, которая аппроксимирует функцию $-K^{-1}\Psi$. Обучение моделей такого вида не представляет принципиальной сложности и осуществляется средствами, рассмотренными в разд. .

В случае, когда по соображениям устойчивости получаемой модели приходится использовать неявную схему дискретизации, процесс формирования модели несколько усложняется, но такой процесс по-прежнему остается реализуемым (процедуры такой реализации можно найти в работе [160, 195].

5.3.2 Нейросетевое представление модели динамической системы с дискретным временем

1. Как было показано выше, если какие-либо априорные знания об объекте моделирования отсутствуют и доступны только экспериментальные данные, характеризующие поведение рассматриваемого объекта, можно сформировать только модель эмпирического типа (модель типа «черный ящик»). Однако если, наряду с экспериментальными данными, все же имеются еще и теоретические знания об объекте, можно их использовать при формировании модели, которая будет в этом случае полуэмпирической (модель типа «серый ящик»). Один из важнейших вопросов, возникающих при этом, состоит в следующем: каким образом организовать обучение таких моделей? Надо ли для каждой из полуэмпирических моделей строить свой собственный алгоритм обучения, учитывающий ее структурную специфику, или же можно каким-либо образом унифицировать и упростить решение этой проблемы?

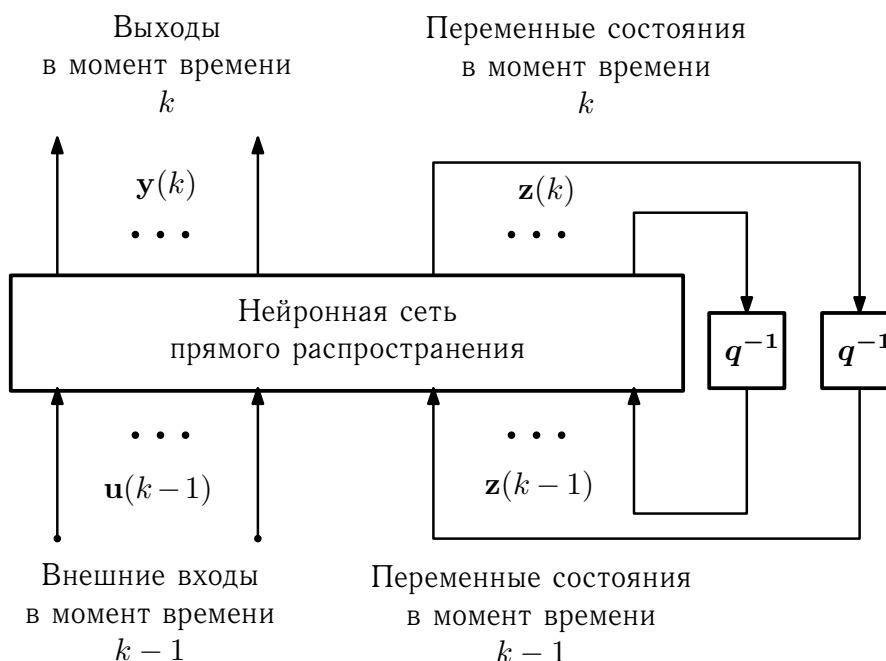


Рис. 5.1. Каноническая форма рекуррентной нейронной сети

Ответ на вопрос о возможности унификации процедур обучения эмпирических и полуэмпирических динамических моделей положителен, как это показано в работе [76]. Это обстоятельство существенно упрощает процесс формирования нейросетевых рекуррентных моделей рассматриваемого класса при решении реальных прикладных задач.

Подход к решению проблемы обучения рекуррентных нейросетевых моделей, предложенный в [76], основывается на сведении исходной модели к некоторой «канонической» форме.

Эта задача может быть решена для любой рекуррентной нейронной сети, при этом итоговая каноническая форма ее будет минимальной по сложности моделью в пространстве состояний.

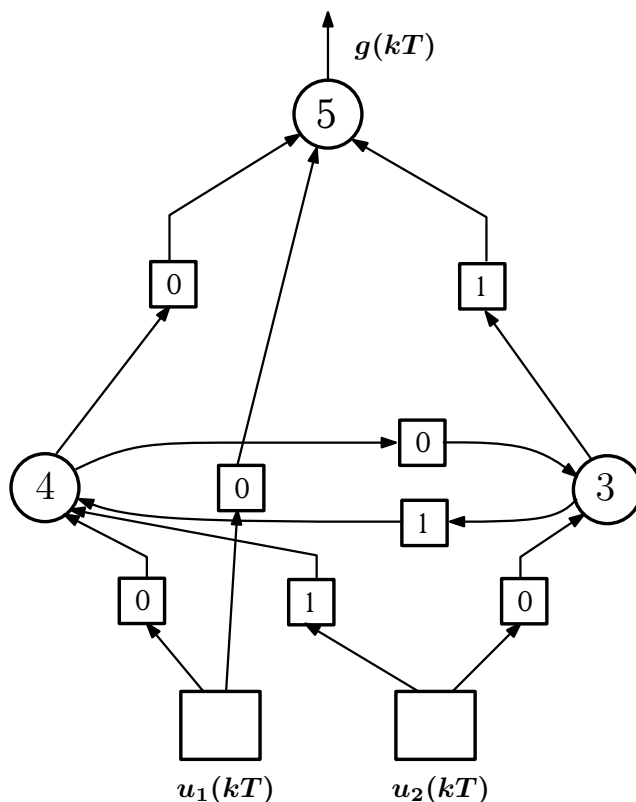


Рис. 5.2. Пример рекуррентной нейронной сети

2. Каноническая форма рекуррентной нейронной сети показана на рис. 5.1, из которого видно что ядром такого представления является нейронная сеть прямого распространения, все обратные связи, имеющиеся в модели, являются внешними по отношению к ядру этой модели и содержат только единичные задержки.

Процедура получения канонической формы для произвольной рекуррентной нейронной сети рассматривается в [76], где дается достаточно общий метод, позволяющий автоматизировать процесс перехода к каноническому представлению модели.

Характер преобразования, которому подвергается исходная рекуррентная сеть при переходе к канонической форме, можно оценить на примере, если сравнить между собой исходную сеть, показанную на рис. 5.2, с сетью в канонической форме, представленной на рис. 5.3.

Здесь приняты следующие соглашения и условные обозначения. Сеть имеет два входа, цифрами в элементах квадратной формы обозначены целочисленные значения задержек сиг-

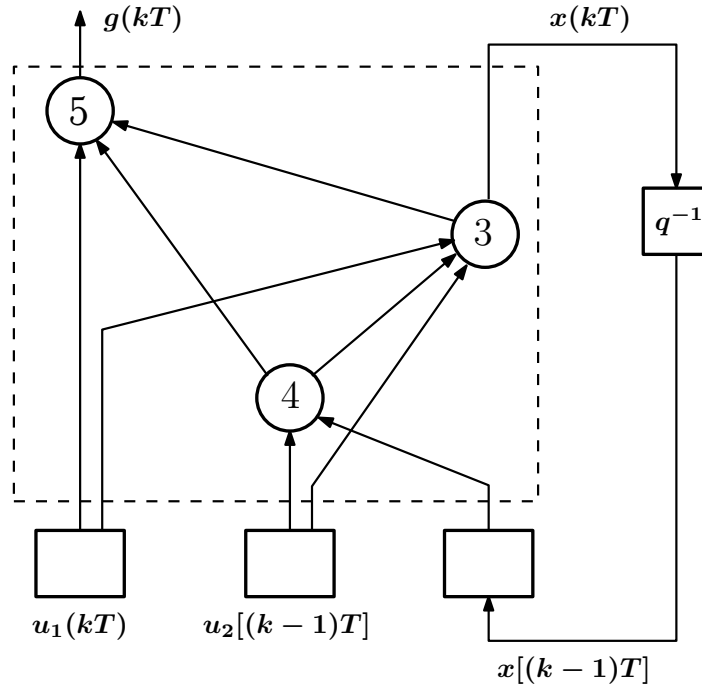


Рис. 5.3. Каноническая форма для рекуррентной нейронной сети из примера на рис. 5.2

нала для соответствующих элементов сети. Сеть содержит в себе цикл, начинающийся и завершающийся в элементе с номером 3 и проходящий через элемент с номером 4. На рис. 5.3 пунктиром выделено ядро полученной модели, представляющее собой слоистую сеть прямого распространения.

3. Проиллюстрируем характер перехода от исходной модели к ее канонической форме еще на одном примере, несколько более сложном [76]. Пусть исследуемый объект описывается следующей системой уравнений:

$$\begin{aligned}
 \ddot{x}_1 &= \phi_1(x_1, x_2, x_3, u), \\
 \dot{x}_2 &= \phi_2(x_1, x_3), \\
 \ddot{x}_3 &= \phi_3(x_1, \dot{x}_2), \\
 y &= x_3.
 \end{aligned}
 \tag{5.15}$$

Представление системы (5.15) в форме с дискретным временем, основанное на использовании явного метода Эйлера, имеет вид:

$$\begin{aligned}
 x_1(k+1) &= \psi_1[x_1(k), x_1(k-1), x_2(k-1), x_3(k-1), u(k-1)], \\
 x_2(k+1) &= \psi_2[x_1(k+1), x_3(k+1)], \\
 x_3(k+1) &= \psi_3[x_3(k), x_3(k-1), x_1(k-1), x_2(k), x_2(k-1)], \\
 y(k+1) &= x_3(k+1).
 \end{aligned}
 \tag{5.16}$$

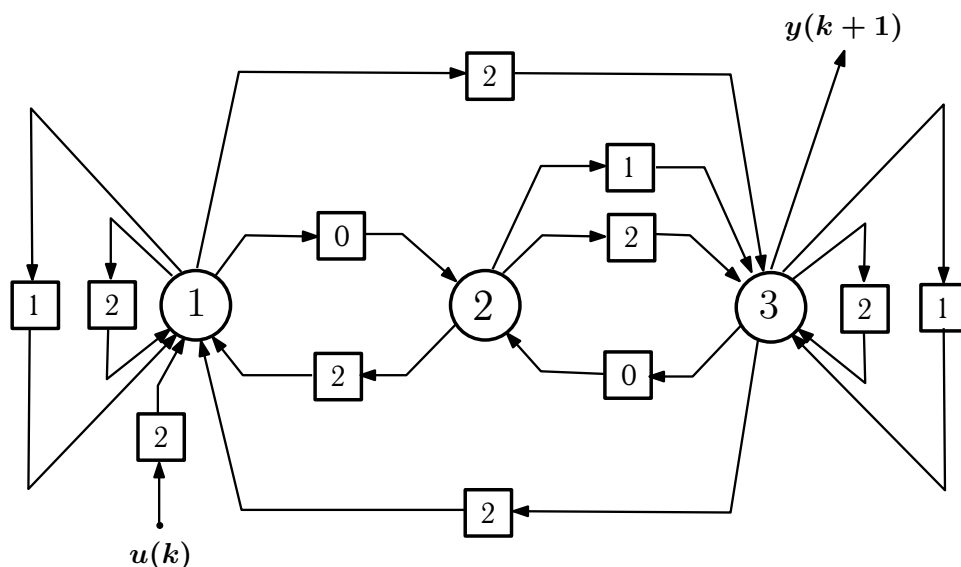


Рис. 5.4. Пример рекуррентной нейронной сети

Явный метод Эйлера, использованный для получения (5.16) из (5.15), основан на аппроксимации следующего вида для производной по времени функции $f(\cdot)$ в момент времени kT :

$$\{f[(k+1)T] - f(kT)\}/T,$$

где T — период дискретизации (шаг интегрирования) для исходной системы (5.16).

Очевидно, соотношения (5.16), полученные выше, представлены не в канонической форме. Рекуррентная нейронная сеть, отвечающая уравнениям (5.16), показана на рис. 5.4. Здесь элементы с номерами 1, 2 и 3 вычисляют функции ψ_1 , ψ_2 и ψ_3 , соответственно. Цифры в элементах квадратной формы представляют собой целочисленные значения задержек для соответствующих связей в сети.

Чтобы перейти к канонической форме, следует ввести новый вектор переменных состояния, который может быть выбран следующим образом:

$$\mathbf{z}(k) = [x_1(k), x_2(k-1), x_3(k), x_3(k-1)]^T.$$

Каноническая форма рекуррентной нейронной сети при таком выборе вектора состояния имеет вид, показанный на рис. 5.5. Полученная модель представляет собой слоистую сеть прямого распространения с элементами единичных задержек в ней и с обратными связями, внешними по отношению к этой сети.

Сеть, показанная на рис. 5.5, полностью эквивалентна сети, представленной на рис. 5.4. Важное следствие проведенной реструктуризации состоит в том, что для модели в канониче-

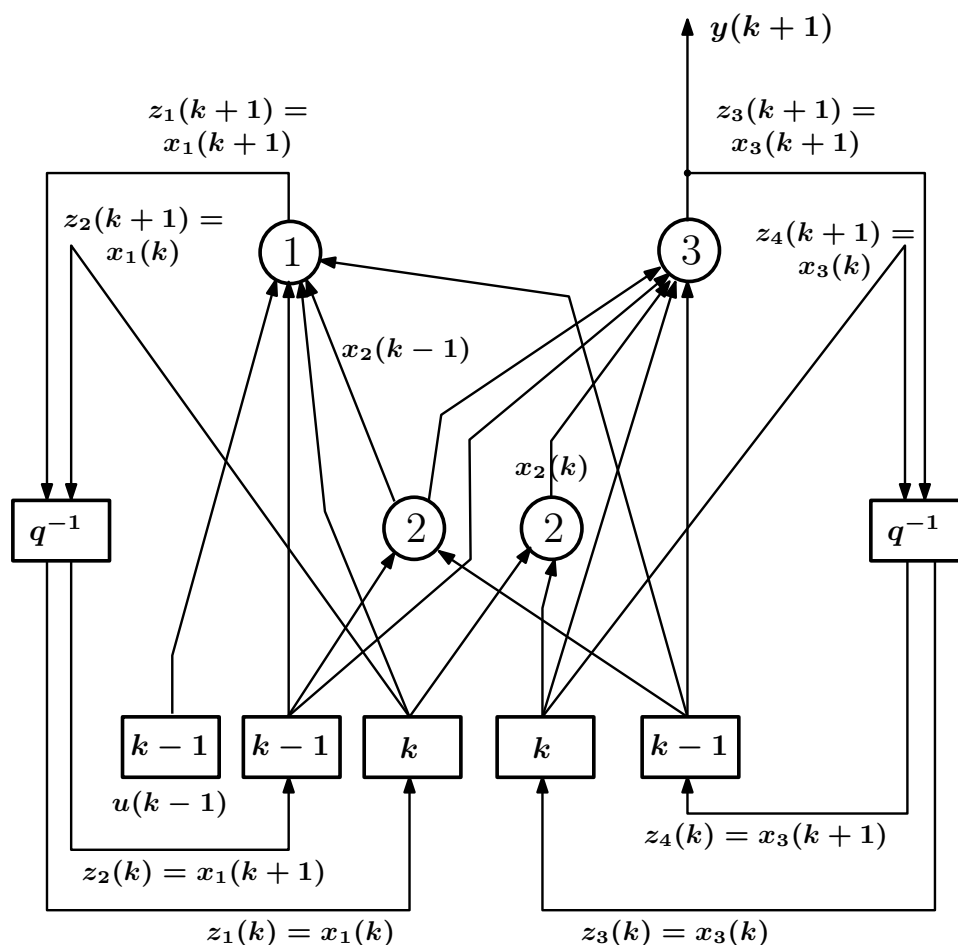


Рис. 5.5. Каноническая форма для рекуррентной нейронной сети из примера на рис. 5.4

ской форме можно использовать традиционные методы обучения сетей с такой архитектурой (см., например, [74]).

5.3.3 Структурная корректировка полуэмпирической НС-модели

Если эмпирические модели формируются исключительно на основе данных о функционировании моделируемого объекта (последовательности значений входных и выходных сигналов для них), то для того, чтобы построить полуэмпирическую модель, требуется располагать априорными знаниями о рассматриваемом объекте. Эти знания чаще всего имеют форму системы уравнений (в общем случае нелинейных) — дифференциальных (обыкновенных или в частных производных) и/или алгебраических. С точки зрения распространенности в приложениях наиболее важным случаем является вариант, когда модель объекта представлена системой нелинейных обыкновенных дифференциальных уравнений. Будем использовать

стандартную запись этих уравнений в пространстве состояний системы:

$$\begin{aligned}\frac{dx}{dt} &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \\ \mathbf{y}(t) &= \mathbf{g}(\mathbf{x}(t)),\end{aligned}\tag{5.17}$$

где \mathbf{x} — вектор переменных состояния, \mathbf{y} — вектор выходов, \mathbf{u} — вектор управляющих переменных, \mathbf{f} и \mathbf{g} — известные вектор-функции.

Данная модель по ряду причин может описывать поведение рассматриваемого объекта не вполне удовлетворительно. Такая ситуация бывает обусловлена, чаще всего, недостаточной детализацией модели, т. е. в ней учтены не все значимые факторы вследствие недостаточного уровня теоретического знания о них. Вторая возможная причина состоит в том, что функции \mathbf{f} и \mathbf{g} (или же некоторые их составные элементы) в (5.17) могут содержать различные величины, значения которых известны недостаточно точно, например, коэффициенты аэродинамических сил и моментов, если (5.17) представляет собой модель движения ЛА.

В том числе и по этой причине необходимо располагать экспериментальными данными о поведении моделируемого объекта, используя которые можно скорректировать модель путем ее дополнительной настройки (обучения). Если только за счет обучения не удастся получить модель с требуемыми свойствами, это будет означать, что модель требует еще и структурной корректировки. Чтобы можно было ее осуществить, требуется сформулировать ряд гипотез, каждая из которых представляет собой попытку представить, что именно в модели препятствует ее нормальной работе и каким образом соответствующее препятствие можно устранить. Соответствующая структурная корректировка осуществляется на модульной основе: объектом ее является некоторая часть модели, заменяемая на другой ее вариант, отвечающий одной из сформулированных гипотез. Эта часть модели рассматривается как достаточно автономный модуль, взаимодействующий с остальными частями модели через соответствующие связи. Такой модуль может реализовываться способом, наиболее пригодным с точки зрения выполняемых им функций. То, как реализуется данный модуль, не влияет, в общем случае, на оставшуюся часть НС-модели.

5.4 Пример формирования полуэмпирической НС-модели

1. В качестве объекта моделирования возьмем управляемую динамическую систему с непрерывным временем с теоретической моделью для нее в виде системы обыкновенных дифференциальных уравнений. Необходимы также экспериментальные данные о поведении

системы, чтобы можно было судить о точности предлагаемых моделей. Эти же данные потребуются для подстройки (обучения) модели, обеспечивающей повышение ее точности. Исходная теоретическая модель может описывать поведение объекта не вполне удовлетворительно из-за того, например, что некоторые ее элементы известны с недостаточной точностью.

Формирование полуэмпирической модели рассмотрим на примере динамической системы, описываемой следующими уравнениями

$$\dot{x}_1(t) = -(x_1(t) + 2x_2(t))^2 + u(t), \quad (5.18)$$

$$\dot{x}_2(t) = 8.322109 \sin(x_1(t)) + 1.135x_2(t). \quad (5.19)$$

Формулировка данной задачи в качестве прототипа использует пример, приведенный в [76].

Неудовлетворительная точность принятой теоретической модели обычно обусловлена тем, что в ней учтены не все значимые факторы из-за недостаточного уровня теоретического знания о них. Экспериментальные данные о поведении объекта позволяют скорректировать такую модель путем ее настройки (обучения). Если обучение не позволяет получить модель с требуемыми свойствами, это означает, что необходима еще и ее структурная корректировка. Для этого следует сформулировать ряд гипотез о том, что именно в модели препятствует ее нормальной работе. Объектом структурной корректировки является некоторая часть модели, заменяемая на другой ее вариант. Эта часть может рассматриваться как модуль, взаимодействующий с остальными частями модели через соответствующие связи.

Примем, что уравнение (5.18) выражает точное знание об объекте и возьмем его в неизменном виде. Уравнение (5.19) запишем вначале в упрощенной форме, чтобы имитировать неполное знание об объекте:

$$\dot{x}_2(t) = 8.32x_1(t). \quad (5.20)$$

Пара уравнений (5.18), (5.20) — это исходная теоретическая модель, являющаяся объектом корректировки с целью повышения ее точности.

2. Корректировку исходной модели можно выполнить, опираясь на экспериментальные данные о его поведении. Средства для этого имеются в НС-моделировании в форме методов обучения. Чтобы ими воспользоваться, теоретическую модель надо преобразовать в НС-форму. Первый шаг на пути к получению НС-модели состоит в переходе от исходной модели с непрерывным временем, т. е. от дифференциальных уравнений, к модели с дискретным временем, т. е. к разностным уравнениям.

Алгоритмической базой для дискретизации моделей с непрерывным времени являются численные методы решения обыкновенных дифференциальных уравнений в сочетании с опытом решения различных видов задач такого рода [38, 167–169]. Для перехода к дискретному времени в задаче (5.18), (5.20) воспользуемся двумя явными разностными схемами, а именно, схемой Эйлера 1-го порядка и схемой Адамса 4-го порядка:

- разностная схема Эйлера

$$\mathbf{x}(k+1) = \mathbf{x}(k) + \Delta t \cdot \mathbf{f}(k); \quad (5.21)$$

- разностная схема Адамса

$$\mathbf{x}(k+1) = \mathbf{x}(k) + \frac{1}{24}\Delta t [55 \cdot \mathbf{f}(k) - 59 \cdot \mathbf{f}(k-1) + 37 \cdot \mathbf{f}(k-2) - 9 \cdot \mathbf{f}(k-3)]. \quad (5.22)$$

В (5.21) и (5.22) использованы следующие обозначения:

$$\begin{aligned} \mathbf{x}(k) &= \mathbf{x}(t_k), \quad \mathbf{f}(k) = \mathbf{f}(t_k, \mathbf{x}(k)), \\ t &= t_0, t_1, \dots, t_k, \dots, t_N, \quad \Delta t = t_k - t_{k-1}. \end{aligned}$$

Примем, что экспериментальные данные о входах и выходах моделируемого объекта измеряются с временным шагом T . Вполне логично использовать эту величину и в качестве шага дискретизации для уравнений (5.18), (5.20), т. е. будем считать, что $\Delta t = T$.

Использование разностной схемы Эйлера приводит к получению для (5.18), (5.20) следующей модели с дискретным временем:

$$\begin{aligned} x_1[(k+1)T] &= x_1(kT) + T[-(x_1(kT) + 2x_2(kT))^2 + u(kT)], \\ x_2[(k+1)T] &= x_2(kT) + T(8.32x_1(kT)) \end{aligned} \quad (5.23)$$

или, если заменить конкретное значение 8.32 числового параметра во втором уравнении на его символическое представление w :

$$\begin{aligned} x_1[(k+1)T] &= x_1(kT) + T[-(x_1(kT) + 2x_2(kT))^2 + u(kT)], \\ x_2[(k+1)T] &= x_2(kT) + T(wx_1(kT)). \end{aligned} \quad (5.24)$$

Аналогичным образом для модели (5.18), (5.20) можно получить вариант с дискретным временем в случае использования схемы Адамса.

3. Величины и связи полученных разностных моделей можно интерпретировать в терминах элементов НС-моделей [76], что приводит к формированию рекуррентной нейронной сети. Чтобы не подстраивать каждый раз алгоритм обучения под специфику полученной НС-модели, сети приводятся к единообразному виду. Такое преобразование позволяет

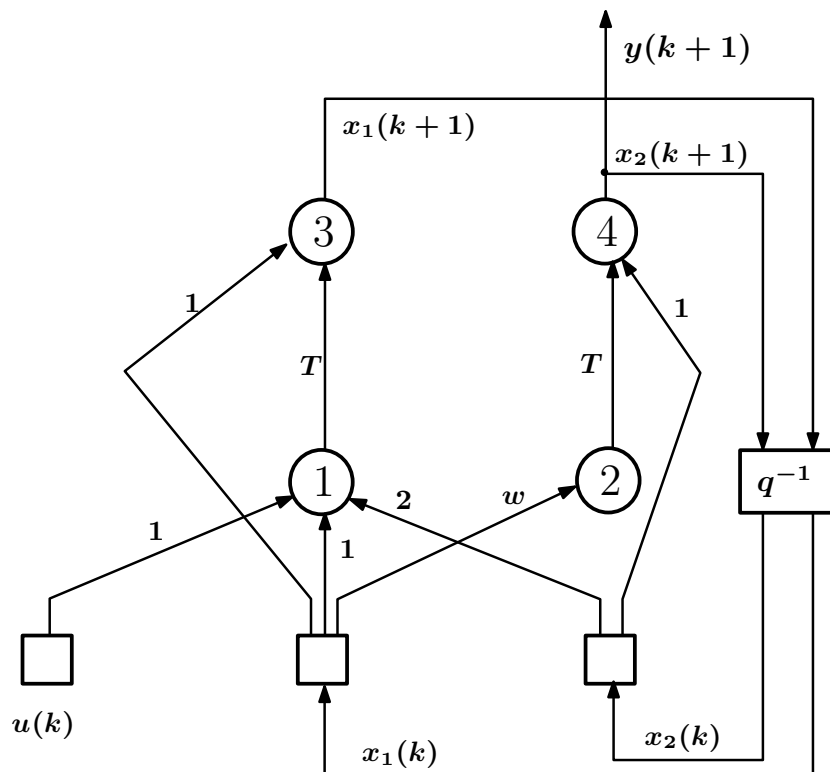


Рис. 5.6. Каноническая форма исходной теоретической модели (5.18), (5.20), дискретизированной с использованием явного метода Эйлера

получить НС-модель канонического вида [164–166] как слоистую сеть прямого распространения, замкнутую внешними обратными связями с задержками от выходов к входам этой сети (рис. 5.1).

В задаче (5.18), (5.20) принятые схемы дискретизации позволяют получить каноническое представление сети либо сразу (для явной схемы Эйлера (5.21), см. рис. 5.6), либо после незначительной корректировки исходного неканонического варианта (для явной схемы Адамса (5.22), см. рис. 5.7).

Алгоритм перехода от разностного к сетевому представлению позволяет сохранить в полученной НС-модели в явном виде локализацию выполняемых функций, которая имела место в исходной модели. Это позволяет при корректировке не подвергать изменениям фрагменты модели, не вызывающие сомнений в рамках решаемой задачи. Другие части модели, потенциально являющиеся причиной ее недостаточной точности, могут корректироваться. Таким образом, в процессе формирования полуэмпирической НС-модели целенаправленной параметрической и структурной корректировке подвергается только часть модели.

Обучающая выборка, требуемая для выполнения корректировки НС-модели, формируется как последовательность наблюдаемых выходов $\{y_i^*\}$, $i = 1, \dots, N$ для заданного управления

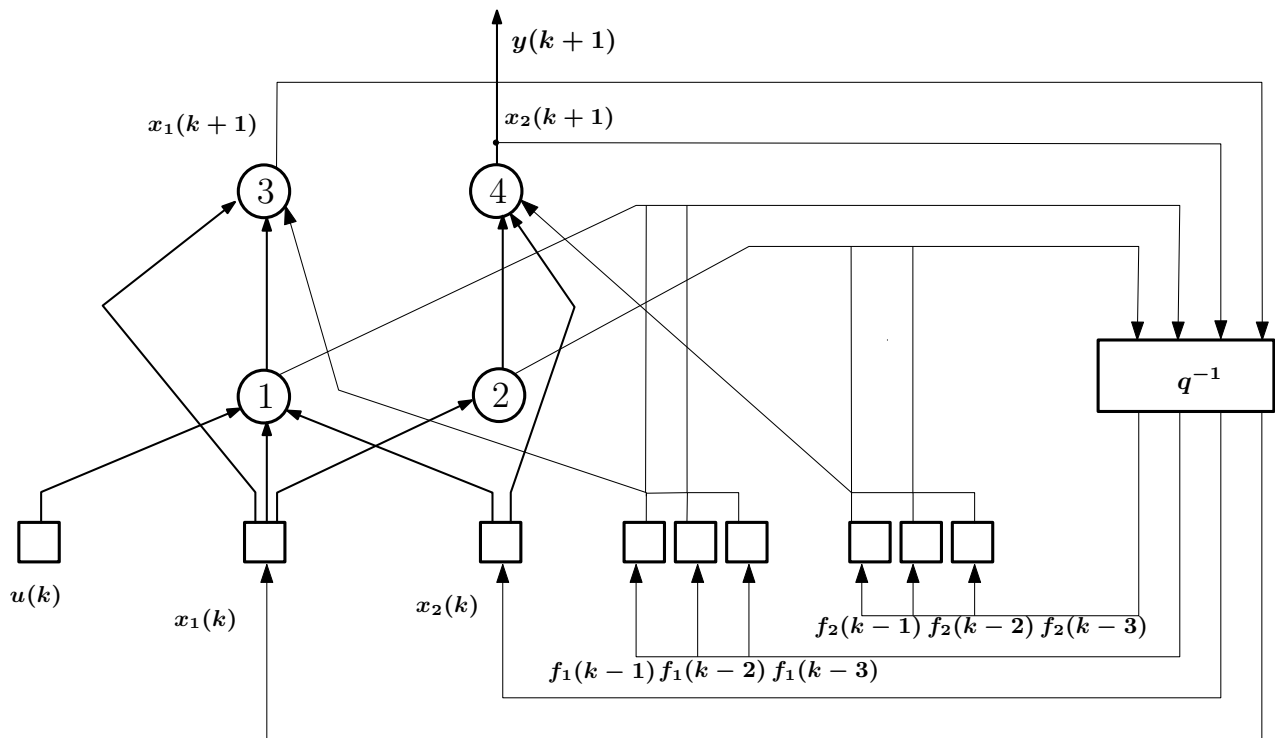


Рис. 5.7. Каноническая форма исходной теоретической модели (5.18), (5.20), дискретизированной с использованием явного метода Адамса

$u(t)$ и начальных условий x_0 . При этом используется случайный входной управляющий сигнал $u(t)$ специфического вида (пример такого сигнала показан на рис. 2.32а), определяющий, наряду с начальными условиями x_0 , поведение моделируемого объекта и его наблюдаемые выходы. Такая форма сигнала $u(t)$ обеспечивает представительность обучающей выборки, так как позволяет обеспечить требуемое разнообразие состояний, в которых может находиться объект моделирования, а также разнообразие скоростей перехода между состояниями. Для временного интервала $t \in [0; 100]$ и шага дискретизации $\Delta t = 0.025$ с в рассматриваемой демонстрационной задаче количество примеров в обучающей выборке будет $N = 1000$.

4. Если при фиксированной структуре за счет обучения не удастся добиться от НС-модели требуемой точности, это означает, что в ней необходимы структурные изменения, которые осуществляются на основе предварительно сформулированных гипотез относительно возможных причин неудовлетворительного поведения модели. Применительно к рассматриваемой задаче можно предположить, например, что требует замены линейная связь между величинами на нелинейную в уравнении (5.20), либо надо дополнительно учесть в нем влияние величины x_2 .

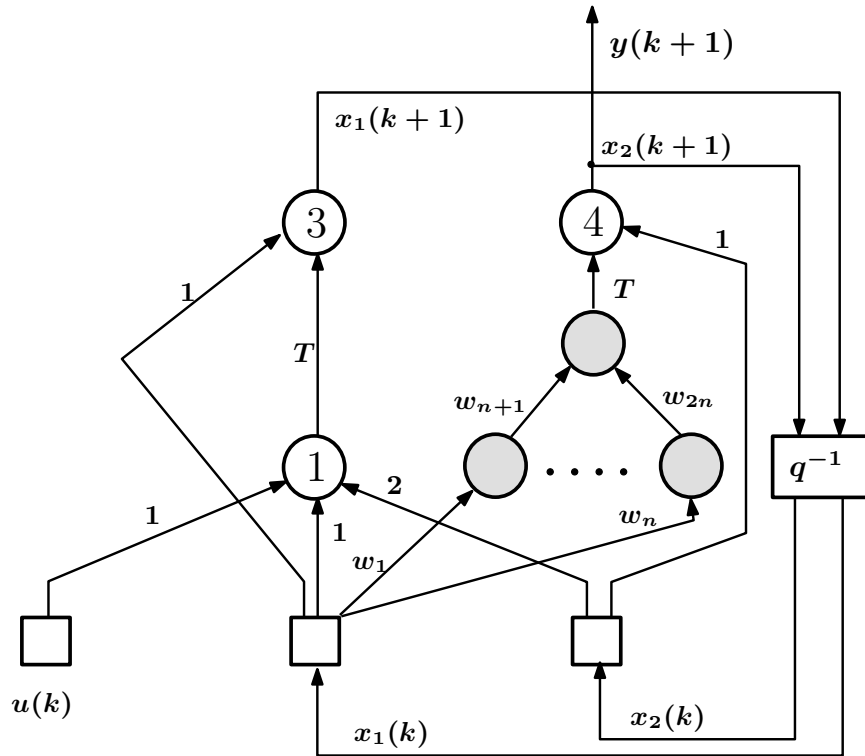


Рис. 5.8. Каноническая форма полуэмпирической модели (метод Эйлера), уточненная путем введения нелинейности в уравнение (5.20)

5.5 Экспериментальная оценка полуэмпирической модели

1. Будем рассматривать систему (5.18), (5.20) на временном интервале $t \in [0; 100]$ с шагом дискретизации $\Delta t = 0.025$ с и начальными условиями $x_1(0) = x_2(0) = 0$. Вектор состояния является частично наблюдаемым: $y(t) = x_2(t)$, с аддитивным белым шумом со среднеквадратическим отклонением (СКО) $\sigma = 0.01$, воздействующим на выход системы $y(t)$.

В идеальном варианте, когда НС-модель абсолютно точно воспроизводит исходную систему дифференциальных уравнений (5.18), (5.19), ошибка моделирования будет полностью определяться шумом, воздействующим на выход системы. Из этого следует, что сопоставление ошибки моделирования с СКО шума позволяет судить о степени успешности решения задачи моделирования рассматриваемой системы, а заданное значение СКО шума можно принять за целевое значение ошибки моделирования.

Обучение проводилось средствами системы Matlab для сетей в форме LDDN (Layered Digital Dynamic Networks), по общему для данного класса сетей алгоритму [77]. Для оптимизации параметров сети используется алгоритм Левенберга-Марквардта, вычисление мат-

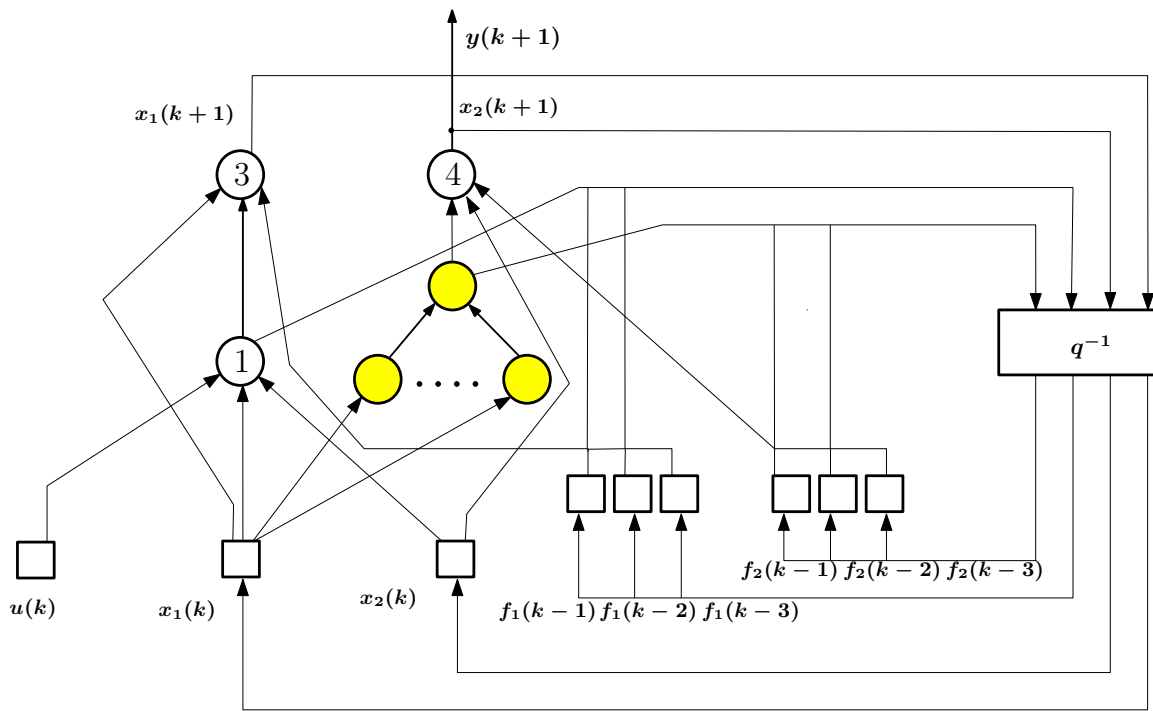


Рис. 5.9. Каноническая форма полуэмпирической модели (метод Аламса), уточненная путем введения нелинейности в уравнение (5.20)

рицы Якоби осуществляется по алгоритму RTRL (Real-Time Recurrent Learning) [74]. В качестве критерия оптимизации используется среднеквадратичная ошибка предсказания модели на обучающей выборке:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (d_i - y_i)^2}, \quad (5.25)$$

где N — количество элементов обучающей выборки, $d = \{d_1, \dots, d_N\}$ — вектор целевых значений наблюдаемой переменной (согласно модели (5.18), (5.19), а $y = \{y_1, \dots, y_N\}$ — вектор выходов нейросетевой модели.

2. Пусть на первом шаге проводимого вычислительного эксперимента теоретическая модель, отвечающая объекту (5.18), (5.19), точно не известна. Мы знаем точно только вид первого уравнения (5.18) из этой модели, вид второго уравнения (5.19) нам не известен. Примем в качестве рабочей гипотезы, что требуемое второе уравнение можно записать в виде (5.20). Если полученную таким способом модель использовать непосредственно для анализа поведения рассматриваемой системы, привлекая методы численного интегрирования систем обыкновенных дифференциальных уравнений, то результат оказывается неудовлетворительным. Как показывает вычислительный эксперимент, среднеквадратичная ошибка, получаемая при таком моделировании, равняется 0.1347 при использовании метода Эйлера и 0.0434 при использо-

Таблица 5.1. Ошибка моделирования для системы (5.18), (5.19)

Метод	ОДУ	НС-1	НС-2	НС-3	Opt
Эйлер	0.1347	0.1278	0.1252	0.0141	–
Адамс	0.0434	0.0390	0.0375	0.0119	–
NARX	–	–	–	–	0.0258

вании метода Адамса, что многократно превышает целевое значение 0.01. Результаты моделирования для этого и всех остальных рассматриваемых случаев представлены в табл. 5.1, где сопоставляются данные, полученные с использованием методов Эйлера и Адамса (ОДУ — результаты, полученные непосредственным интегрированием исходной системы дифференциальных уравнений (5.18), (5.20); НС-1, НС-2, НС-3 — результаты для НС-моделей, полученных после первого, второго и третьего шагов структурной корректировки, соответственно, а также с помощью чисто эмпирической НС-модели типа NARX (четвертый столбец)).

3. Если попытаться проанализировать причины указанной неудачи, то можно сформулировать следующие утверждения и предположения. Как уже отмечалось, первое из уравнений (5.18) исходной теоретической модели точно описывает характер изменения величины $x_1(t)$. Но относительно второго уравнения, имеющего вид (5.20), такой уверенности нет, следовательно, именно данное уравнение не позволяет получить требуемую точность описания объекта. Возможные причины такого положения дел состоят в следующем:

- значение $w = 8.32$ числового параметра в этом уравнении может быть неточным, т. е. следует попытаться подобрать другое значение данного параметра;
- вызывает сомнения линейный характер зависимости от x_1 , используемый в правой части данного уравнения, т. е. если корректировка значения числового параметра w из предыдущего пункта не позволит получить решение с требуемой точностью, необходимо попытаться использовать какое-либо нелинейное соотношение по x_1 в правой части уравнения (5.20);
- отсутствие переменной x_2 в правой части второго уравнения также может послужить причиной недостаточной точности модели, т. е. если меры, предложенные в двух предыдущих пунктах, не дадут требуемого результата, следует попытаться переформулировать правую часть уравнения (5.20) с привлечением x_2 .

Полуэмпирическая форма представления модели дает возможность внести в нее подобного рода изменения. Это осуществляется путем введения модуля-подсети, реализующего

Таблица 5.2. Ошибка моделирования для системы (5.18), (5.26)

Метод	ОДУ	НС-1	НС-2	НС-3	Opt
Эйлер	0.0644	0.0588	0.0582	0.0129	–
Адамс	0.0285	0.0258	0.0253	0.0119	–
NARX	–	–	–	–	0.0302

Таблица 5.3. Ошибка моделирования для системы (5.18), (5.27)

Метод	ОДУ	НС-1	НС-2	НС-3	Opt
Эйлер	0.0696	0.0622	0.0605	0.0117	–
Адамс	0.0458	0.0421	0.0413	0.0112	–
NARX	–	–	–	–	0.0258

необходимую нелинейность, в НС-модель, полученную дискретизацией из соответствующей теоретической модели. При этом, если изменения вносить в том порядке, в котором они перечислены выше, сложность модели будет последовательно возрастать и можно ожидать, что точность ее также будет возрастать.

Как видно из второго столбца табл. 5.1, в сравнении его с первым столбцом, подстройка значения коэффициента в уравнении (5.20) позволила несколько улучшить точность модели, но совершенно недостаточно: ошибка моделирования снизилась с 0.1347 до 0.1278 для метода Эйлера и с 0.0434 до 0.0390 для метода Адамса при целевом значении 0.01. По этой причине необходимо использовать второе предложение из списка, приведенного выше, направленное на повышение точности модели, а именно, следует заменить уравнение (5.20) с линейной зависимостью в его правой части от величины x_1 на уравнение с нелинейной зависимостью. Эта зависимость вводится в нейросетевую модель путем замены в ней фрагмента для упомянутой линейной зависимости (нейрон, отвечающий правой части уравнения (5.20), на нелинейную зависимость, которая задается однослойной сетью с сигмоидальной активационной функцией и 10 нейронами в слое (число нейронов подобрано в вычислительном эксперименте). Структурная схема НС-модели, отвечающая данному варианту применительно к схеме дискретизации Эйлера показана на рис. 5.8. Смещения нейронов скрытого слоя здесь не изображены, чтобы не загромождать схему. Изменения в НС-модели, основанной на схеме Адамса, представлены на рис. 5.9.

Введение нелинейности по x_1 в уравнение (5.20), как следует из результатов, представлен-

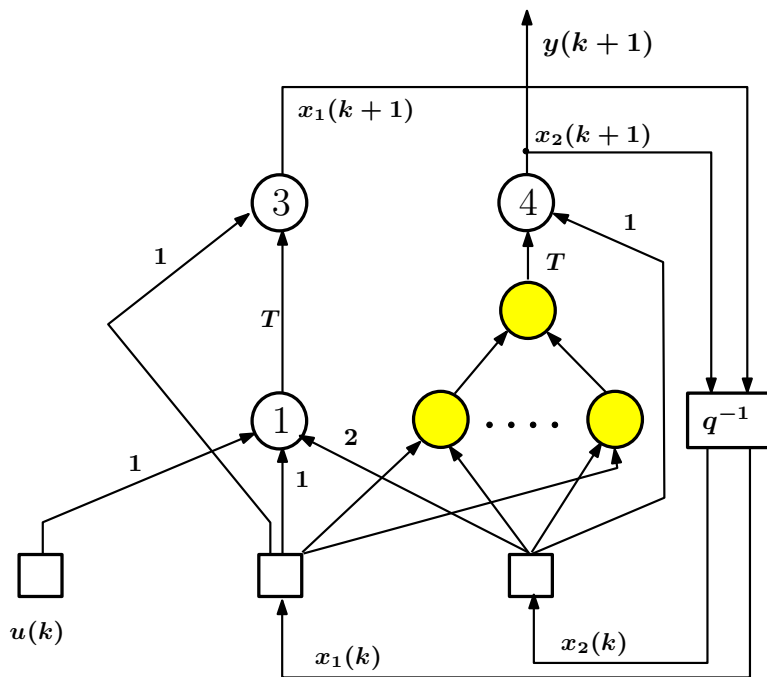


Рис. 5.10. Каноническая форма полуэмпирической модели (метод Эйлера), уточненная за счет введения величины x_2 в уравнение (5.26)

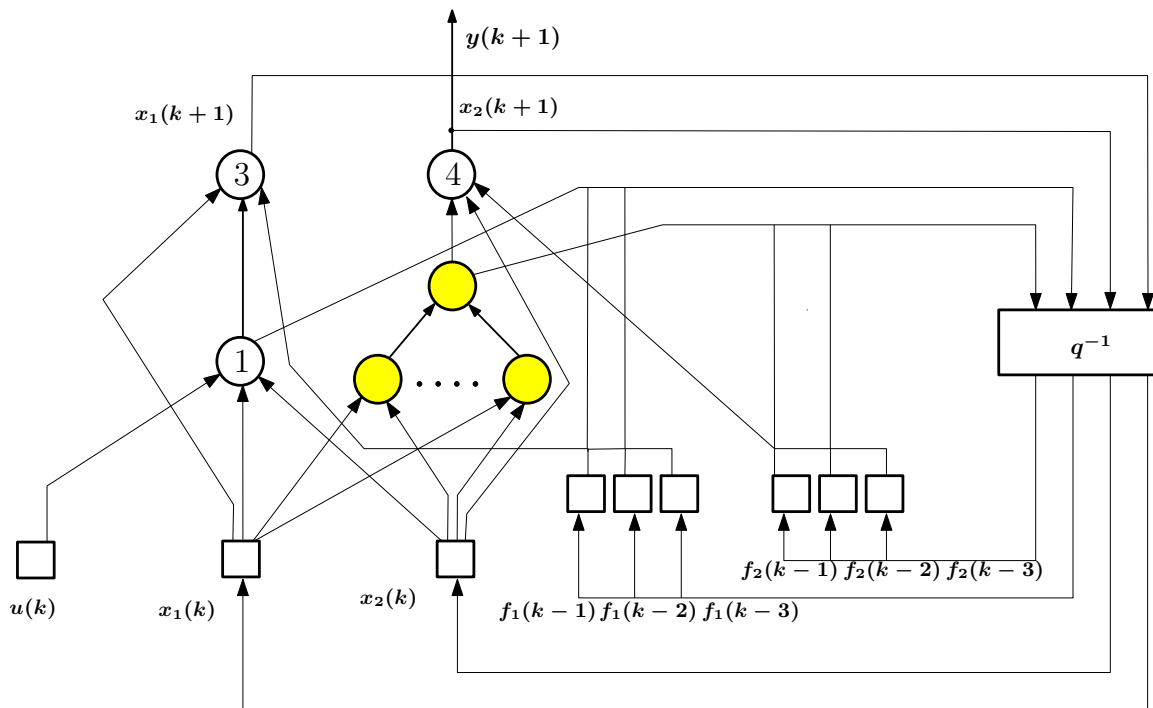


Рис. 5.11. Каноническая форма полуэмпирической модели (метод Адамса), уточненная путем введения величины x_2 в уравнение (5.20)

ных в третьем столбце табл. 5.1, позволило улучшить точность модели совсем незначительно, а именно, ошибка снизилось с 0.1278 до 0.1252 для метода Эйлера и с 0.0390 до 0.0375 для

метода Адамса. Это означает, что только за счет подбора подходящего значения нелинейной зависимости от x_1 в правой части уравнения (5.20) требуемой точности добиться нельзя, необходимо, очевидно, учитывать зависимость не только от x_1 , но и от x_2 , т. е. необходимо реализовать третье предложение из списка, представленного выше. Эта зависимость вводится в нейросетевую модель путем добавления в ее структуру связи от x_2 к введенному на предыдущем шаге фрагменту модели в виде однослойной сети. Структурная схема сети, отвечающая данному варианту НС-модели для схемы дискретизации Эйлера показана на рис. 5.10, аналогично модификация выполняется в случае схемы Адамса.

Как видно из столбца НС-3 табл. 5.1, введение дополнительной связи по x_2 помимо связи по x_1 в (5.20) позволило решить поставленную задачу. Точность модели теперь составляет 0.0141 для метода Эйлера и 0.0119 для метода Адамса.

Для сравнения с традиционным подходом, продемонстрируем результаты обучения NARX-модели (столбец Орт в табл. 5.1). Лучшая точность 0.0258 в данном примере была достигнута сетью с 3 нейронами и 5 задержками в обратной связи. Из этого сравнения видно неоспоримое преимущество полуэмпирической модели над эмпирической по достижимой точности решения: даже для метода Эйлера точность составляет 0.0141 против 0.0258 для NARX, в случае метода Адамса точность еще выше — она равна 0.0119, при этом возможно еще большее повышение точности за счет применения более эффективных разностных схем.

4. Анализ, аналогичный описанному выше, был проведен также еще для двух вариантов рассматриваемой системы (5.18), (5.20). В первом из них вместо уравнения (5.20) используется уравнение со смесью гармоник в правой части:

$$\dot{x}_2(t) = 8.322109 \sin(x_1(t)) + 0.7 \cos(1.33\pi x_2(t)). \quad (5.26)$$

Еще один вариант вместо (5.20) вводит уравнение, которое содержит в правой части более сложную смесь гармоник, чем в случае (5.20):

$$\dot{x}_2(t) = 8.322109 \sin(x_1(t)) + 0.7 \cos(1.33\pi x_2(t))^2. \quad (5.27)$$

Результаты вычислений по этим вариантам представлены в табл. 5.2 и 5.3, соответственно.

5. Сопоставление результатов экспериментов для всех трех примеров позволяет сделать следующие выводы.

С повышением сложности поведения моделируемого объекта точность моделирования несколько снижается, причем, как правило, в разной степени для различных схем дискретизации. Это свидетельствует о том, что варьируя вид используемой разностной схемы, можно

влиять на результаты моделирования, соответственно, возникает проблема выбора адекватной (в идеале — наилучшей) схемы дискретизации для конкретной решаемой прикладной задачи.

Второй важный вывод связан с сопоставлением результатов для эмпирических и полуэмпирических моделей. Видно, что полуэмпирические модели многократно превосходят по точности модели чисто эмпирического типа. При этом разрыв в точности моделирования увеличивается с усложнением поведения рассматриваемой системы. В связи с этим, в тех случаях, когда точность, получаемая от эмпирической модели недостаточна и есть теоретические знания об объекте моделирования, переход к модели полуэмпирического типа дает значительный эффект.

Полученные результаты показывают, что методы нейросетевого моделирования, особенно в сочетании со знаниями и опытом из соответствующей предметной области и из традиционного вычислительного моделирования, являются мощным и перспективным инструментом, потенциально пригодным для решения сложных прикладных проблем применительно к управляемым системам различных классов.

Концепция полуэмпирической модели обеспечивает поэтапное формирование НС-представления нелинейной управляемой динамической системы, используя на первом этапе как основу ее теоретическую модель. Введение теоретического знания об объекте моделирования в эмпирическую модель нейросетевого типа позволяет многократно повысить точность нейросетевой модели в сравнении с ее чисто эмпирическим вариантом.

В следующем разделе рассматривается применение полуэмпирических НС-моделей для решения задач анализа движения ЛА и идентификации аэродинамических характеристик ЛА.

6 Нейросетевое полуэмпирическое моделирование движения и идентификация характеристик летательных аппаратов

6.1 Полуэмпирическое моделирование продольного углового движения маневренного самолета

1. В данном разделе на примере продольного короткопериодического движения маневренного самолета демонстрируется высокая эффективность полуэмпирических НС-моделей (моделей типа «серый ящик») при решении прикладных задач. Теоретической моделью при этом является соответствующая традиционная модель движения самолетов в виде системы обыкновенных дифференциальных уравнений. Формируемая в рассматриваемом конкретном примере полуэмпирическая НС-модель включает два элемента-модуля типа «черный ящик», описывающих зависимости коэффициентов подъемной силы и момента тангажа от переменных состояния (угла атаки, угловой скорости тангажа и угла отклонения управляемого стабилизатора), подлежащие восстановлению на основе имеющихся экспериментальных данных для наблюдаемых переменных состояния динамической системы.

2. В примере используется модель углового продольного движения летательного аппарата, традиционная для динамики полета самолетов [39]. Данная модель имеет вид:

$$\begin{aligned}\dot{\alpha} &= \omega_z - \frac{qS}{mV} C_{y_a}(\alpha, \omega_z, \varphi) + \frac{g}{V} \cos \vartheta, \\ \dot{\omega}_z &= \frac{qS b_A}{J_{zz}} m_z(\alpha, \omega_z, \varphi), \\ T^2 \ddot{\varphi} &= -2T\zeta \dot{\varphi} - \varphi + \varphi_{act},\end{aligned}\tag{6.1}$$

где α — угол атаки, град; ϑ — угол тангажа, град; ω_z — угловая скорость тангажа, град/с; φ — угол отклонения управляемого стабилизатора, град; C_{y_a} — коэффициент подъемной силы; m_z — коэффициент момента тангажа; m — масса самолета, кг; V — воздушная скорость, м/с; $q = \rho V^2 / 2$ — скоростной напор; ρ — плотность воздуха, кг/м³; g — ускорение силы тяжести, м/с²; S — площадь крыла, м²; b_A — средняя аэродинамическая хорда крыла, м; J_{zz} — момент инерции самолета относительно боковой оси, кг·м²; безразмерные коэффициенты C_{y_a} и m_z являются нелинейными функциями своих аргументов; T , ζ — постоянная времени и коэффициент относительного демпфирования привода, φ_{act} — командный сигнал на привод цельноповоротного горизонтального оперения (управляемого стабилизатора) с ограничением на диапазон углов отклонения $\pm 25^\circ$) для него. В модели (6.1) величины α , ω_z , φ и $\dot{\varphi}$ —

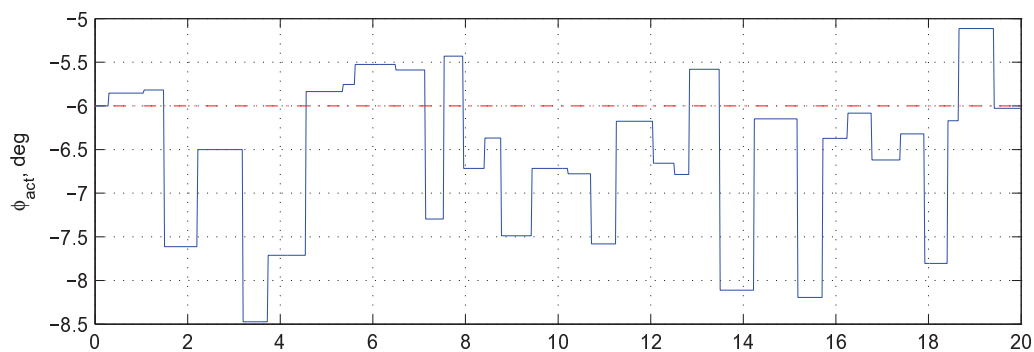
это состояния объекта управления, величина φ_{act} — управление. В качестве примера конкретного объекта моделирования рассматривался маневренный самолет F-16, исходные данные для которого были взяты из [44]. Вычислительный эксперимент с моделью (6.1) проводился для временного интервала $t \in [0; 20]$ с с шагом дискретизации $\Delta t = 0.02$ с для частично наблюдаемого вектора состояния $y(t) = [\alpha(t); \omega_z(t)]^T$, с аддитивным белым шумом со среднеквадратичным отклонением (СКО) $\sigma = 0.01$, воздействующим на выход системы $y(t)$.

3. Как уже ранее отмечалось, одним из критически важных вопросов, возникающих при формировании как эмпирических, так и полуэмпирических НС-моделей, является получение обучающего набора, обеспечивающего адекватное отражение особенностей поведения моделируемой системы. Это формирование осуществляется путем выработки соответствующего тестового управляющего воздействия на моделируемый объект и оценки реакции объекта на это воздействие.

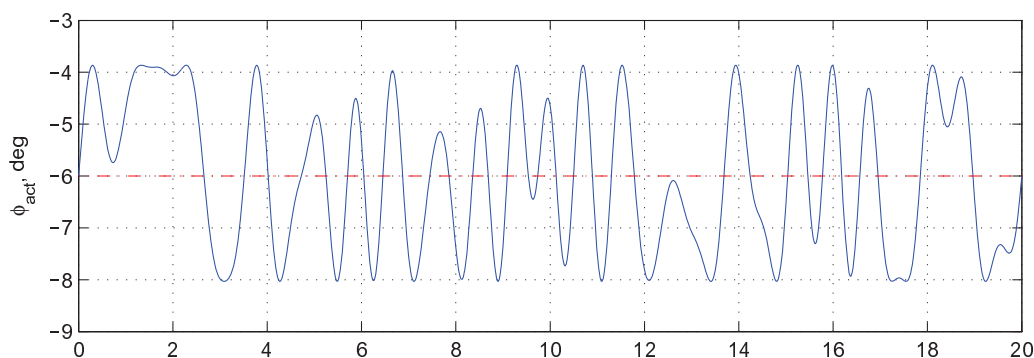
Анализируется влияние вида тестового управляющего воздействия на репрезентативность получаемой совокупности экспериментальных данных, используемой в качестве обучающего набора для НС-модели. Типовые возмущающие воздействия (ступенчатое, импульсное, дублет и случайный сигнал) сравниваются с полигармоническим воздействием, формируемым специальным образом. Сравнение, основанное на результатах вычислительного эксперимента, выполнялось при различных тестовых маневрах, включая прямолинейный горизонтальный полет с постоянной скоростью и полет с монотонно увеличивающимся углом атаки.

4. При решении проблем рассматриваемого вида одной из важнейших задач является формирование информативного (репрезентативного) набора данных, характеризующего поведение моделируемой ДС на всей области изменения значений величин, описывающих ДС и производных (скоростей изменения) этих величин. Как было показано в разд. 2.3, требуемые обучающие данные для формируемой НС-модели могут быть получены за счет специальным образом организованных тестовых возбуждающих воздействий на моделируемую ДС. Из перечня тестовых сигналов, рассмотренных в разд. 2.3, как показывают данные вычислительных экспериментов, требуемую информативность обучающего набора обеспечивают в рассматриваемой задаче только случайный и полигармонический сигналы (рис. 6.1).

Результативность применения различных видов тестовых сигналов, перечисленных выше, может быть наглядно представлена в графическом виде. Для этой цели будем использовать диаграммы информативности обучающих данных, представляющие собой диаграммы покрытия области допустимых значений величин и их производных, описывающих объект моде-



(a)



(b)

Рис. 6.1. Тестовые возбуждающие воздействия, используемые при изучении динамики управляемых систем: (a) — случайный сигнал; (b) — полигармонический сигнал. Здесь φ_{act} — командный сигнал привода руля высоты (цельноповоротного горизонтального оперения)

лирования, с помощью примеров, получаемых при воздействии на объект того или иного тестового сигнала. Применительно к задаче (6.1) в это переченъ входят α , $\dot{\alpha}$, ω_z , $\dot{\omega}_z$, φ , $\dot{\varphi}$, $\ddot{\varphi}$.

Диаграммы информативности позволяют сопоставить представительность обучающих наборов, получаемых применением различных тестовых воздействий на моделируемый объект. Набор будет тем лучше, чем он более плотно и равномерно покрывает требуемую область изменения величин, описывающих поведение рассматриваемого объекта. Однако исходное представление в семимерном пространстве, определяемом указанным перечнем, не будет наглядным, для получения наглядности следует перейти к двумерным представлениям. С этой целью рассматривались попарные комбинации $(\alpha, \dot{\alpha})$, (α, ω_z) , $(\alpha, \dot{\omega}_z)$, (α, φ) , $(\omega_z, \dot{\omega}_z)$, (φ, ω_z) , наиболее информативные с точки зрения решаемой прикладной задачи. В качестве примера на рис. 6.2 показаны диаграммы $\alpha, \dot{\alpha}$ для одного из популярных тестовых сигналов (дублет) и полигармонического сигнала, сформированного по введенным выше правилам. Наглядно

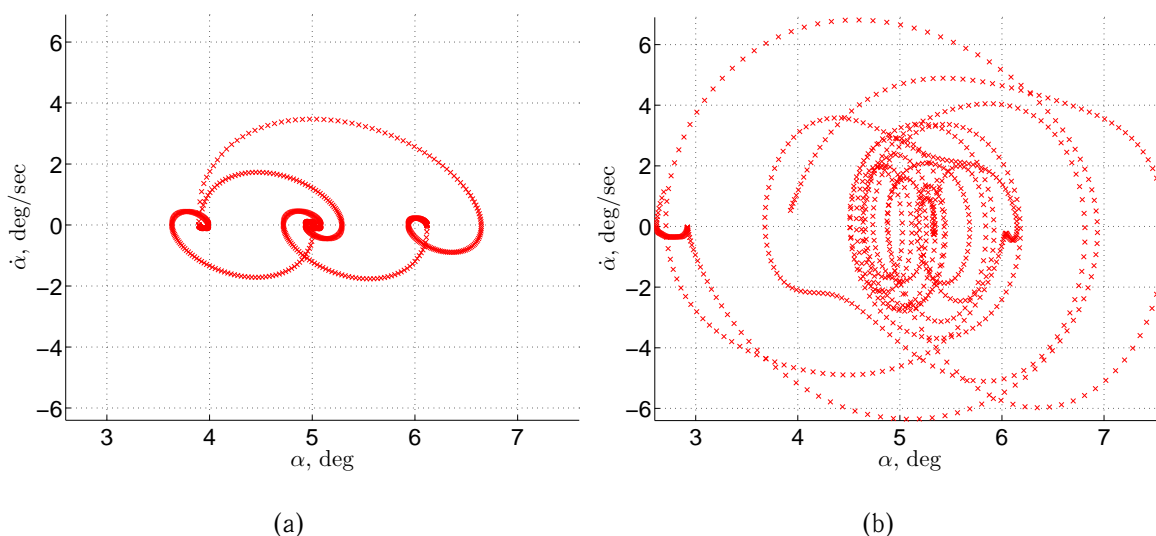


Рис. 6.2. Диаграммы информативности обучающего набора $(\alpha, \dot{\alpha})$ для воздействий типа дублет (а) и полигармонического (б) при равном числе (1000) обучающих примеров

видно преимущество полигармонического сигнала по критериям плотности и равномерности размещения обучающих примеров.

В Приложении Б для задачи моделирования продольного углового движения маневренного самолета даются диаграммы информативности для всех тестовых возмущающих сигналов, рассмотренных в разд. 2.3, применительно к двум вариантам тестовых маневров: для прямолинейного горизонтального полета («точечный режим») и для полета с монотонно увеличивающимся углом атаки («монотонный режим»). Эти диаграммы, а также фазовые траектории, порождающие их, приведены на рис. Б.1–Б.22 (точечный режим) и Б.23–Б.44 (монотонный режим).

В разд. 2.3.5.2 приведен алгоритм формирования полигармонического сигнала, обеспечивающий получение минимальной величины пик-фактора. Диаграммы покрытия дают возможность наглядно представить влияние пик-фактора на равномерность покрытия примерами области допустимых значений величин, характеризующих состояние ДС (рис. 6.4), а также сопоставить итоговый результат с тем, что дает один из популярных тестовых сигналов — дублет. Наглядно видно, что дублет существенно уступает полигармоническому сигналу в информативности получаемого набора обучающих примеров. Аналогичным образом можно убедиться, что полигармоническому сигналу уступают по информативности и все остальные тестовые сигналы, приведенные выше.

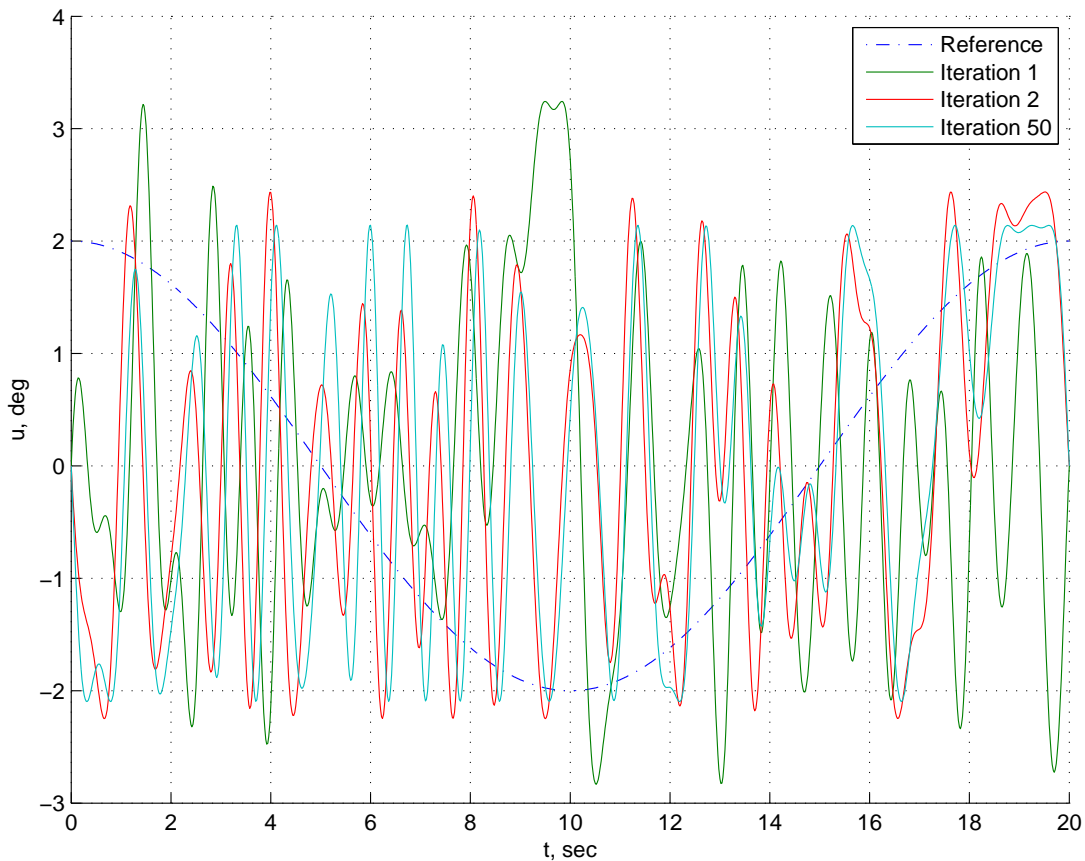


Рис. 6.3. Процесс формирования полигармонического сигнала

5. Как и в примере, рассмотренном в разд. 5.5, будем в качестве целевого значения ошибки моделирования использовать СКО аддитивного шума, действующего на выход системы.

Обучение на выборке $\{y_i\}$, $i = 1, \dots, N$, полученной с помощью исходной модели (6.1), проводится в системе Matlab для сетей в форме LDDN (Layered Digital Dynamic Networks) с использованием алгоритма Левенберга-Марквардта по критерию среднеквадратичной ошибки модели. Матрица Якоби вычисляется по алгоритму RTRL (Real-Time Recurrent Learning) [74].

6. Применение к (6.1) упомянутой выше процедуры формирования полуэмпирической НС-модели приводит к получению структуры, показанной на рис. 6.5 (она основана на использовании схемы дискретизации Эйлера). Аналогичная структура для случая полностью эмпирической модели (NARX), отвечающая той же самой задаче (6.1), показана на рис. 6.6.

7. Была проведена обширная серия вычислительных экспериментов по сравнению эффективности всех перечисленных выше тестовых сигналов для двух видов испытательных маневров: прямолинейный горизонтальный полет с постоянной скоростью и полет с монотонно увеличивающимся углом атаки. В качестве типичного примера на рис. 6.7 показано, насколько

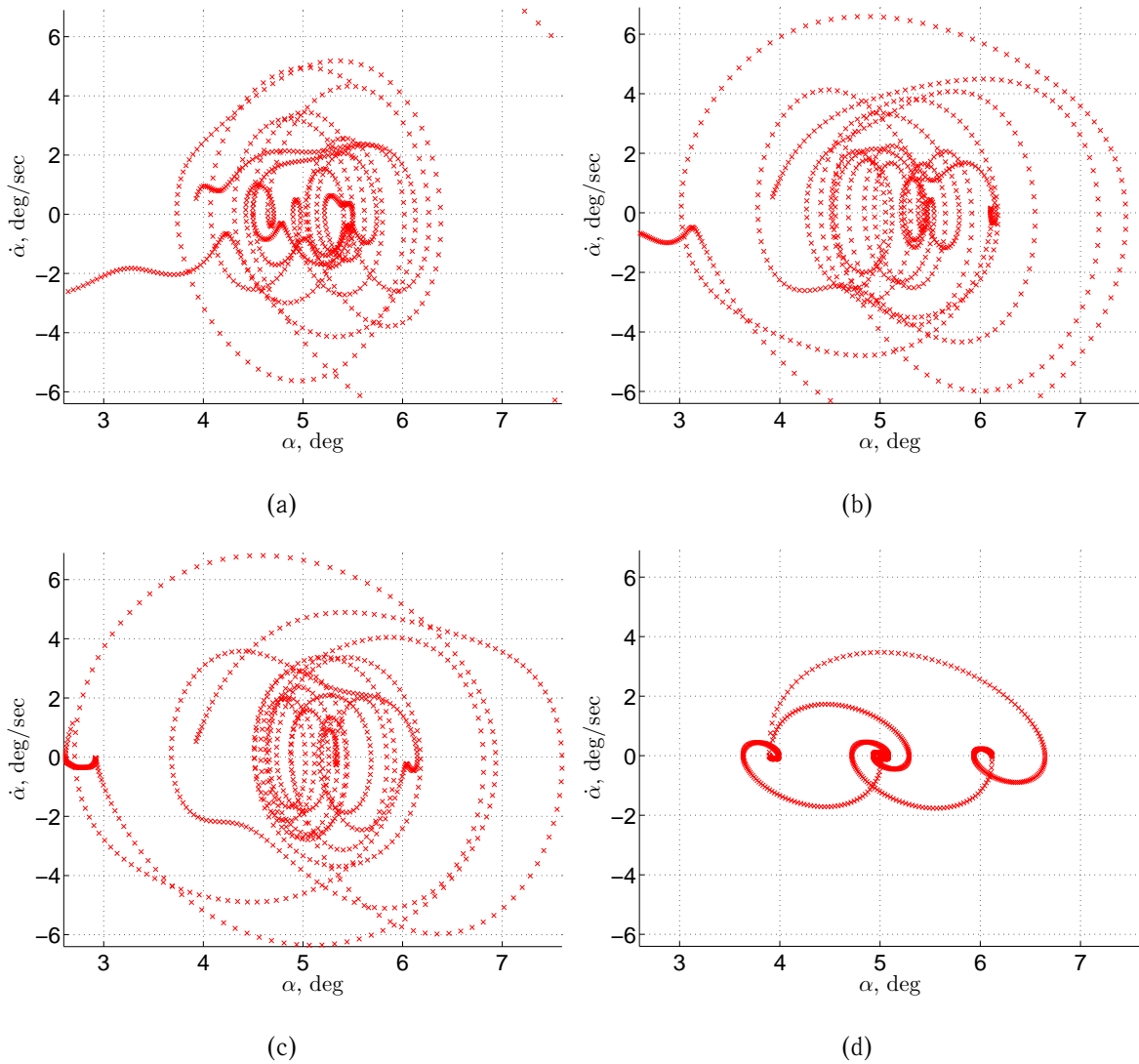


Рис. 6.4. Изменение вида диаграмм информативности для процесса формирования полигармонического сигнала $(\alpha, \dot{\alpha})$, показанного на рис. 6.3 для итераций 1, 2 и 50 ((a), (b) и (c)), соответственно в сравнении с воздействием типа «дублет» (d); число примеров (1000) везде одинаковое

ко точно восстанавливаются неизвестные зависимости (нелинейные функции $C_y(\alpha)$, $m_z(\alpha)$). Оценивалась точность полученной полуэмпирической НС-модели, использующей эти зависимости, в сравнении с исходной системой (6.1), в которой использовались точные представления функций $C_y(\alpha)$, $m_z(\alpha)$. Результаты для этих двух моделей настолько близки, что кривые на графиках практически совпадают. Числовые оценки точности получаемых моделей приведены в табл. 6.1 (оценка точности воспроизведения данных из обучающего набора) и табл. 6.2 (оценка обобщающих свойств НС-модели). Здесь же приведено сопоставление полученных моделей с NARX-моделями.

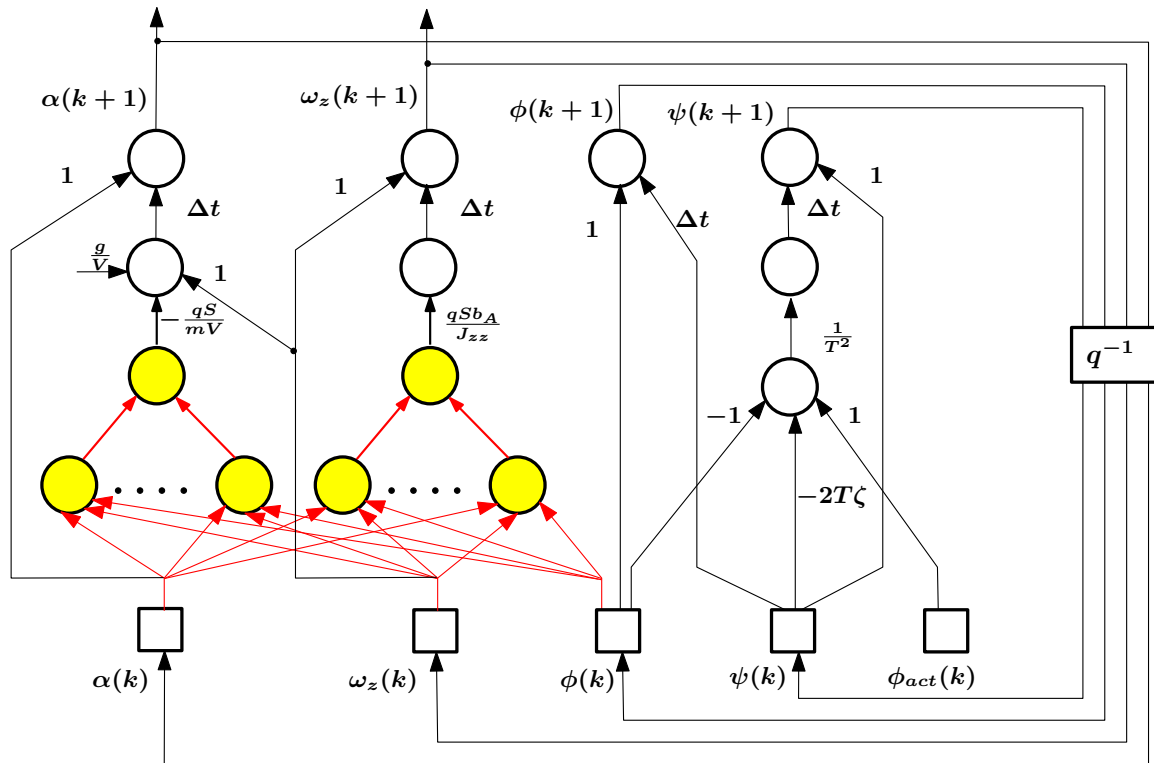


Рис. 6.5. Полуэмпирическая НС-модель продольного углового движения самолета (на основе схемы дискретизации Эйлера)

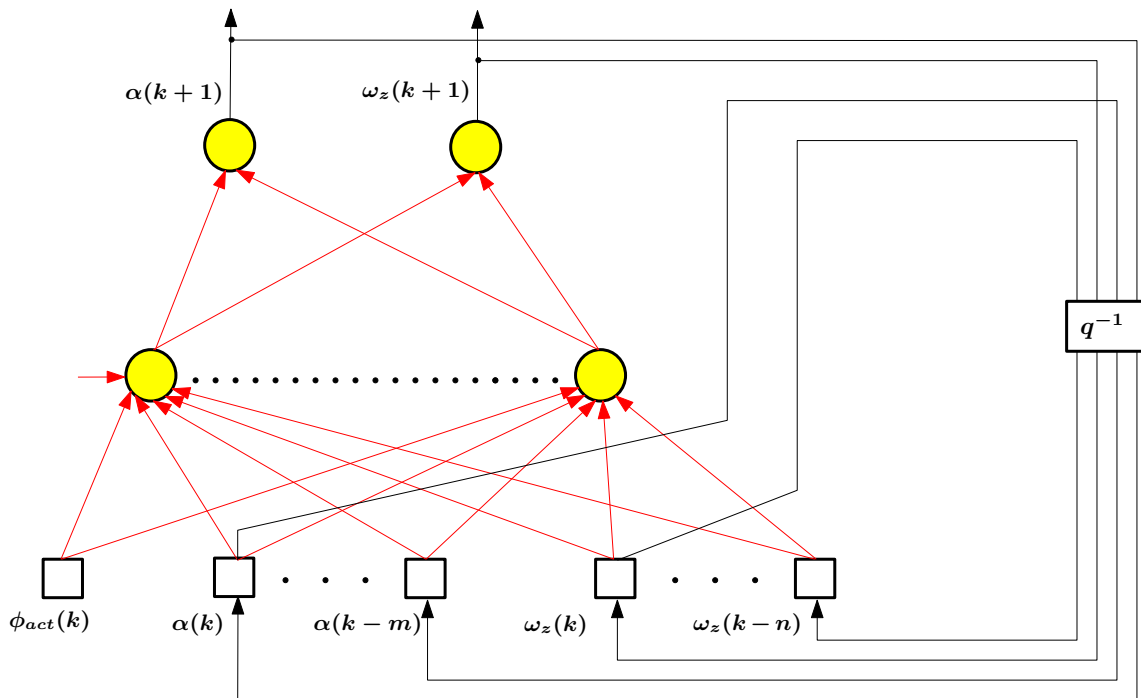


Рис. 6.6. Эмпирическая НС-модель продольного углового движения самолета (NARX)

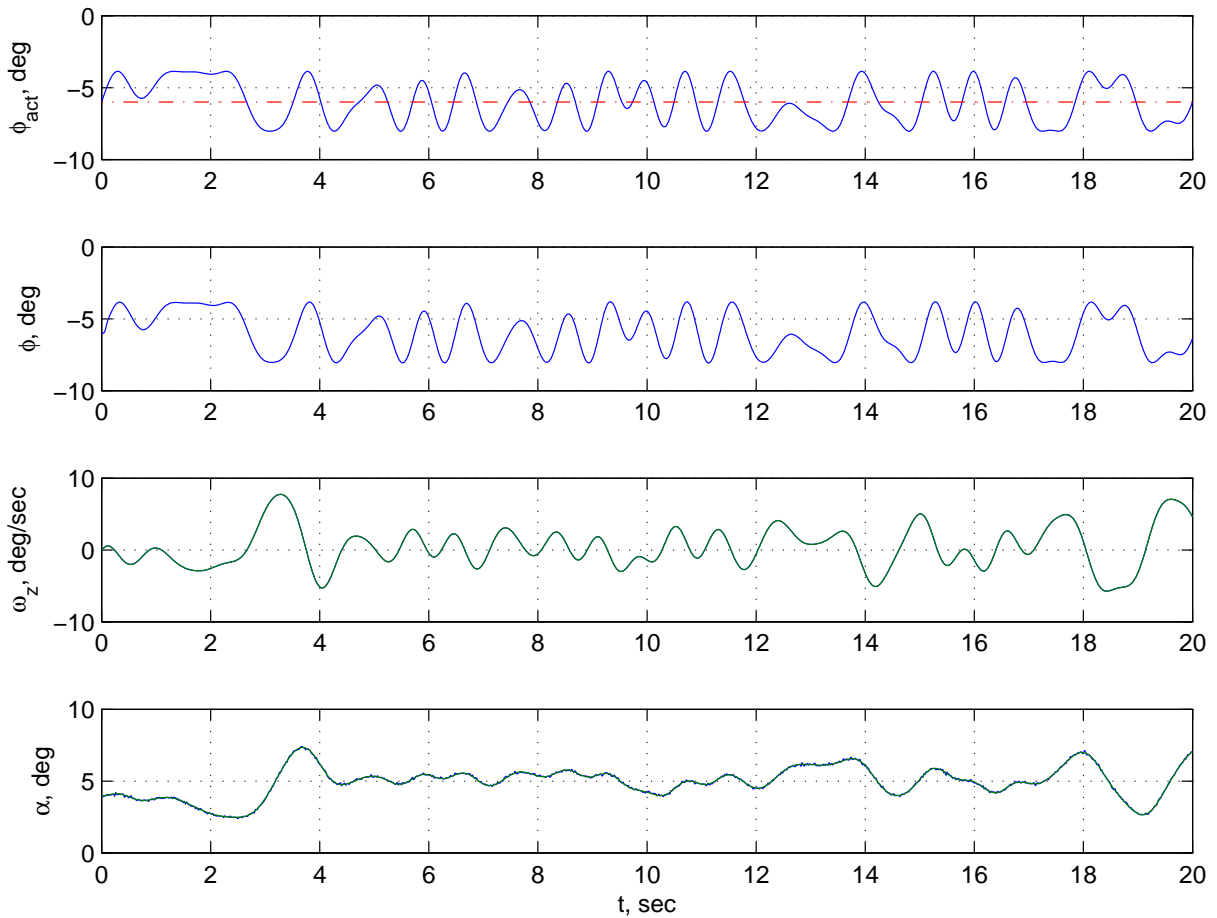


Рис. 6.7. Оценка точности восстановления зависимостей $C_{y\alpha}(\alpha)$ и $m_z(\alpha)$ по результатам тестирования НС-модели (точечный режим, идентификация и тестирование — полигармонический сигнал). Значения выходов объекта (6.1) и НС-модели показаны сплошной линией и пунктиром, соответственно

ТАБЛИЦА 6.1. Ошибка моделирования на обучающем множестве (полигармонический сигнал)

Задача	Точечный режим		Монотонный режим	
	СКО $_{\alpha}$	СКО $_{\omega_z}$	СКО $_{\alpha}$	СКО $_{\omega_z}$
Дообучение C_y	$1.02 \cdot 10^{-3}$	$1.24 \cdot 10^{-4}$	$1.02 \cdot 10^{-3}$	$1.24 \cdot 10^{-4}$
Обучение C_y	$1.02 \cdot 10^{-3}$	$1.23 \cdot 10^{-4}$	$1.02 \cdot 10^{-3}$	$1.24 \cdot 10^{-4}$
Обучение C_y, m_z	$1.02 \cdot 10^{-3}$	$1.19 \cdot 10^{-4}$	$1.02 \cdot 10^{-3}$	$1.27 \cdot 10^{-4}$
NARX	$1.85 \cdot 10^{-3}$	$3.12 \cdot 10^{-3}$	$1.12 \cdot 10^{-3}$	$7.36 \cdot 10^{-4}$

ТАБЛИЦА 6.2. Ошибка моделирования на тестовом множестве
(полигармонический сигнал)

Задача	Точечный режим		Монотонный режим	
	СКО _α	СКО _{ω_z}	СКО _α	СКО _{ω_z}
Дообучение C_y	$1.02 \cdot 10^{-3}$	$1.59 \cdot 10^{-4}$	$1.02 \cdot 10^{-3}$	$1.17 \cdot 10^{-4}$
Обучение C_y	$1.02 \cdot 10^{-3}$	$1.59 \cdot 10^{-4}$	$1.02 \cdot 10^{-3}$	$1.17 \cdot 10^{-4}$
Обучение C_y, m_z	$1.02 \cdot 10^{-3}$	$1.32 \cdot 10^{-4}$	$1.02 \cdot 10^{-3}$	$1.59 \cdot 10^{-4}$
NARX	$2.32 \cdot 10^{-2}$	$4.79 \cdot 10^{-2}$	$3.16 \cdot 10^{-2}$	$5.14 \cdot 10^{-2}$

ТАБЛИЦА 6.3. Ошибка моделирования на тестовом множестве для
полуэмпирической модели для трех видов возмущающих сигналов

Сигнал	Точечный режим		Монотонный режим	
	СКО _α	СКО _{ω_z}	СКО _α	СКО _{ω_z}
Дублет	0.0202	0.0417	8.6723	34.943
Случайный	0.0041	0.0071	0.0772	0.2382
Полигармонический	0.0029	0.0076	0.0491	0.1169

В табл. 6.3 сопоставляются значения ошибки моделирования в зависимости от вида возмущающего сигнала для рассматриваемой полуэмпирической модели продольного короткопериодического движения самолета. Аналогичные результаты для эмпирической модели намного менее точны, в частности, для полигармонического возмущающего сигнала СКО_α = 1.3293, СКО_{ω_z} = 2.7445.

В табл. 6.1, 6.2 и 6.2 как точечный режим обозначен прямолинейный горизонтальный полет с постоянной скоростью и как монотонный режим — полет с монотонно увеличивающимся углом атаки. Кроме того, термином «обучение» для соответствующих аэродинамических коэффициентов обозначается задача восстановления соответствующей неизвестной функции «с нуля», т. е. в предположении об отсутствии какой-либо информации о возможных значениях этих коэффициентов. Термином «дообучение» обозначается задача уточнения значений

соответствующего коэффициента, известных, например, по результатам продувок в аэродинамической трубе.

8. Как отмечалось выше, в ряде случаев помимо восстановления неизвестных функций (в рассматриваемом случае это $C_{y_a}(\alpha, \omega_z, \varphi)$ и $m_z(\alpha, \omega_z, \varphi)$) представляют интерес также и значения производных этих функций, например, $C_{y_a}^\alpha$ и m_z^α . После завершения обучения полуэмпирической модели можно извлечь из нее НС-модули для функций C_{y_a} и m_z . Производные выходов этих НС-модулей по их входам можно вычислить по алгоритму, аналогичному обратному распространению, предназначенному для вычисления производных функции ошибки сети по ее весам и смещениям. Пользуясь цепным правилом дифференцирования, можно выразить производную выхода a_k^m для k -го нейрона из m -го (выходного) слоя по входу p_i через чувствительности $s_{k,j}^{m,l}$:

$$\frac{\partial a_k^m}{\partial n_j^l} = \sum_{(j,l) \in IC_i} s_{k,j}^{m,l} \cdot \frac{\partial n_j^l}{\partial p_j}, \quad s_{k,j}^{m,l} = \frac{\partial a_k^m}{\partial n_j^l},$$

где n_j^l — взвешенный вход j -го нейрона l -го слоя; IC_i — множество пар индексов $\langle j, l \rangle$, определяющих номер j нейрона в l -ом слое, имеющего связь с i -м входом p_i . При этом чувствительности $s_{k,j}^{m,l}$ вычисляются в ходе выполнения алгоритма обратного распространения ошибки, а производные $(\partial n_j^l / \partial p_j)$ равняются весам соответствующих входных связей (для нейрона с взвешенным сумматором в качестве входного отображения). Использование данного алгоритма дает, например, следующие значения производных $C_{y_a}^\alpha$ и m_z^α , вычисленные для точечного режима ($\omega_z = 0$), для балансировочных значений угла отклонения стабилизатора φ и угла атаки α : $C_{y_a}^\alpha = 0.5$, $m_z^\alpha = -0.5$. Аналогичным образом можно найти производные и для любых других сочетаний значений аргументов функций C_{y_a} и m_z .

9. Полученные результаты позволяют сделать вывод о том, что методы полуэмпирического нейросетевого моделирования, сочетающие знания и опыт из соответствующей предметной области, а также из традиционного вычислительного моделирования, являются мощным и перспективным инструментом, потенциально пригодным для решения сложных задач описания и анализа управляемого движения летательных аппаратов. Сопоставление результатов, получаемых в рамках полуэмпирического подхода, с тем, что дает традиционное НС-моделирование (модели типа NARX), показывает неоспоримые преимущества полуэмпирических моделей.

6.2 Идентификация аэродинамических характеристик летательного аппарата

1. В предыдущем разделе эффективность полуэмпирического подхода к НС-моделированию ДС была продемонстрирована на задаче продольного углового движения маневренного самолета. Это относительно простая задача, вследствие ее невысокой размерности и, что более существенно, в силу использования одноканального управления (канал тангажа, используется единственный орган управления — цельноповоротный стабилизатор). В данном разделе решается существенно более сложная задача — формирование НС-модели полного углового движения (с тремя используемыми одновременно органами управления — цельноповоротным стабилизатором, рулем направления и элеронами), а также идентификации пяти из шести коэффициентов сил и моментов. Как и в предыдущем случае, теоретической моделью в решаемой задаче является соответствующая традиционная модель движения самолета, в которой содержится ряд факторов неопределенности. Для устранения имеющихся неопределенностей формируемая полуэмпирическая НС-модель включает пять элементов-модулей типа «черный ящик», описывающих коэффициенты подъемной и боковой силы, коэффициенты момента тангажа, рыскания и крена, каждый из которых нелинейно зависит от нескольких параметров движения самолета. Эти пять зависимостей требуется найти (восстановить) на основе имеющихся экспериментальных данных для наблюдаемых переменных динамической системы, т. е. решить задачу идентификации аэродинамических характеристик самолета.

2. Предлагаемый подход к идентификации аэродинамических характеристик самолета существенно отличается от традиционно принятого при решении задач данного класса. А именно, при традиционном подходе [114–125] принято, как правило, использовать линеаризованную модель возмущенного движения летательного аппарата. При этом зависимости для аэродинамических сил и моментов, действующих на ЛА, представляются в виде разложения их в ряд Тейлора, с оставлением в нем только членов первого порядка (в редких случаях — еще и членов второго порядка). В таком варианте решение задачи идентификации сводится к восстановлению по экспериментальным данным зависимостей, описывающих коэффициенты упомянутого разложения, определяющими в которых являются частные производные безразмерных коэффициентов аэродинамических сил и моментов по различным параметрам движения ЛА (C_y^α , C_z^β , m_z^α , $m_z^{\omega_z}$ и т. п.). В отличие от этого, полуэмпирический подход реализует восстановление соотношений для коэффициентов сил C_x , C_y , C_z и моментов m_x , m_y , m_z как целостных нелинейных зависимостей от соответствующих аргументов, не прибегая

к разложению их в ряд, т. е. оцениваются сами функции, а не коэффициенты разложения их в ряд. Каждая из таких зависимостей реализуется как отдельный НС-модуль, встроенный в полуэмпирическую модель. Если производные C_y^α , C_z^β , m_z^α , $m_z^{\omega_z}$ и т. п. необходимы для решения каких-либо задач, например, для анализа характеристик устойчивости и управляемости самолета, их несложно найти с использованием соответствующих НС-модулей, полученных при формировании полуэмпирической НС-модели.

3. Теоретическая модель полного углового движения самолета, которая является исходной для процесса формирования полуэмпирической НС-модели, представляет собой систему обыкновенных дифференциальных уравнений, традиционную для динамики полета самолетов [39]. Данная модель имеет следующий вид:

$$\begin{cases} \dot{\omega}_x = c_3 M_x + c_4 M_y - \omega_z (c_1 \omega_y + c_2 \omega_x) \\ \dot{\omega}_y = c_4 M_x + c_9 M_y + \omega_z (c_2 \omega_y - c_8 \omega_x) \\ \dot{\omega}_z = c_7 M_z - c_5 \omega_x \omega_y + c_6 (\omega_x^2 - \omega_y^2) \end{cases} \quad (6.2)$$

$$\begin{cases} \dot{\gamma} = \omega_x - \omega_y \operatorname{tg} \vartheta \cos \gamma + \omega_z \operatorname{tg} \vartheta \sin \gamma \\ \dot{\psi} = \omega_y \frac{\cos \gamma}{\cos \vartheta} - \omega_z \\ \dot{\vartheta} = \omega_y \sin \gamma + \omega_z \cos \gamma \frac{\sin \gamma}{\cos \vartheta} \end{cases} \quad (6.3)$$

$$\begin{cases} \dot{\alpha} = \omega_z + (\omega_y \sin \alpha - \omega_x \cos \alpha) \operatorname{tg} \beta + \frac{1}{m V_T \cos \beta} (-Y_a - m g_{a_y}) \\ \dot{\beta} = \omega_x \sin \alpha + \omega_y \cos \alpha + \frac{1}{m V_T} (Z_a + m g_{a_z}) \end{cases} \quad (6.4)$$

$$\begin{cases} T_B^2 \ddot{\delta}_B = -2 T_B \xi_B \dot{\delta}_B - \delta_B + \delta_{B_{act}} \\ T_\delta^2 \ddot{\delta}_\delta = -2 T_\delta \xi_\delta \dot{\delta}_\delta - \delta_\delta + \delta_{\delta_{act}} \\ T_H^2 \ddot{\delta}_H = -2 T_H \xi_H \dot{\delta}_H - \delta_H + \delta_{H_{act}} \end{cases} \quad (6.5)$$

В рассматриваемой модели приняты следующие обозначения: $\omega_x, \omega_y, \omega_z$ — угловые скорости крена, рыскания и тангажа, град/с; γ, ψ, ϑ — углы крена, рыскания и тангажа, град; α, β — углы атаки и скольжения, град; $\delta_B, \delta_H, \delta_\delta$ — углы отклонения управляемого стабилизатора, руля направления и элеронов, град; $\dot{\delta}_B, \dot{\delta}_H, \dot{\delta}_\delta$ — угловые скорости отклонения управляемого стабилизатора, руля направления и элеронов, град/с; V_T — воздушная скорость, м/с; $\delta_{B_{act}},$

$\delta_{\text{накт}}, \delta_{\text{эакт}}$ — командные сигналы на приводы управляемого стабилизатора, руля направления и элеронов, град; $T_{\text{в}}, T_{\text{н}}, T_{\text{э}}$ — постоянные времени приводов управляемого стабилизатора, руля направления и элеронов, с; $\xi_{\text{в}}, \xi_{\text{н}}, \xi_{\text{э}}$ — коэффициенты относительного демпфирования приводов управляемого стабилизатора, руля направления и элеронов; X_a, Y_a, Z_a — лобовое сопротивление, подъемная и боковая силы; M_x, M_y, M_z — моменты крена, тангажа и рыскания; m — масса самолета, кг.

Коэффициенты c_1, \dots, c_9 в уравнениях (6.2) определяются следующим образом: $c_0 = J_x J_y - J_{xy}^2$, $c_1 = [(J_z - J_y)J_y - J_{xy}^2]/c_0$, $c_2 = [(J_x + J_y - J_z)J_{xy}]/c_0$, $c_3 = J_y/c_0$, $c_4 = J_{xy}/c_0$, $c_5 = (J_y - J_x)/J_z$, $c_6 = J_{xy}/J_z$, $c_7 = 1/J_z$, $c_8 = [J_x(J_x - J_z) + J_{xy}^2]/c_0$, $c_9 = J_x/c_0$, где J_x, J_y, J_z — моменты инерции самолета относительно продольной, нормальной и боковой оси, кг·м²; J_{xy}, J_{yz}, J_{zx} — центробежные моменты инерции самолета, кг·м².

Аэродинамические силы X_a, Y_a, Z_a в уравнениях (6.3) и моменты M_x, M_y, M_z в уравнениях (6.2) определяются соотношениями вида:

$$\begin{cases} X_a = -X \cos \alpha \cos \beta - Y \sin \alpha \cos \beta + Z \sin \beta \\ Y_a = -X \sin \alpha + Y \cos \alpha \\ Z_a = X \cos \alpha \sin \beta + Y \sin \alpha \sin \beta + Z \cos \beta \end{cases} \quad (6.6)$$

$$\begin{cases} X = \bar{q} S C_x(\alpha, \beta, \delta_{\text{в}}, \omega_z) \\ Y = \bar{q} S C_y(\alpha, \beta, \delta_{\text{в}}, \omega_z) \\ Z = \bar{q} S C_z(\alpha, \beta, \delta_{\text{н}}, \delta_{\text{э}}, \omega_x, \omega_y) \end{cases} \quad (6.7)$$

$$\begin{cases} M_x = \bar{q} S l m_x(\alpha, \beta, \delta_{\text{в}}, \delta_{\text{н}}, \delta_{\text{э}}, \omega_x, \omega_y) \\ M_y = \bar{q} S l m_y(\alpha, \beta, \delta_{\text{в}}, \delta_{\text{н}}, \delta_{\text{э}}, \omega_x, \omega_y) \\ M_z = \bar{q} S b_A m_z(\alpha, \beta, \delta_{\text{в}}, \omega_z) \end{cases} \quad (6.8)$$

Величины $g_{a_x}, g_{a_y}, g_{a_z}$ — проекции ускорения силы тяжести на оси скоростной системы координат, м/с², требуемые в уравнениях (6.4), задаются соотношениями:

$$\begin{cases} g_{a_x} = g(-\sin \vartheta \cos \alpha \cos \beta + \cos \gamma \cos \vartheta \sin \alpha \cos \beta + \sin \gamma \cos \vartheta \sin \beta) \\ g_{a_y} = g(-\sin \vartheta \sin \alpha - \cos \gamma \cos \vartheta \cos \alpha) \\ g_{a_z} = g(\sin \vartheta \cos \alpha \sin \beta - \cos \gamma \cos \vartheta \sin \alpha \sin \beta + \sin \gamma \cos \vartheta \cos \beta) \end{cases} \quad (6.9)$$

Кроме того, в соотношениях (6.7), (6.8) использованы обозначения: X, Y, Z — аэродинамическая продольная, нормальная и поперечная сила; S — площадь крыла самолета, м²; l, b_A

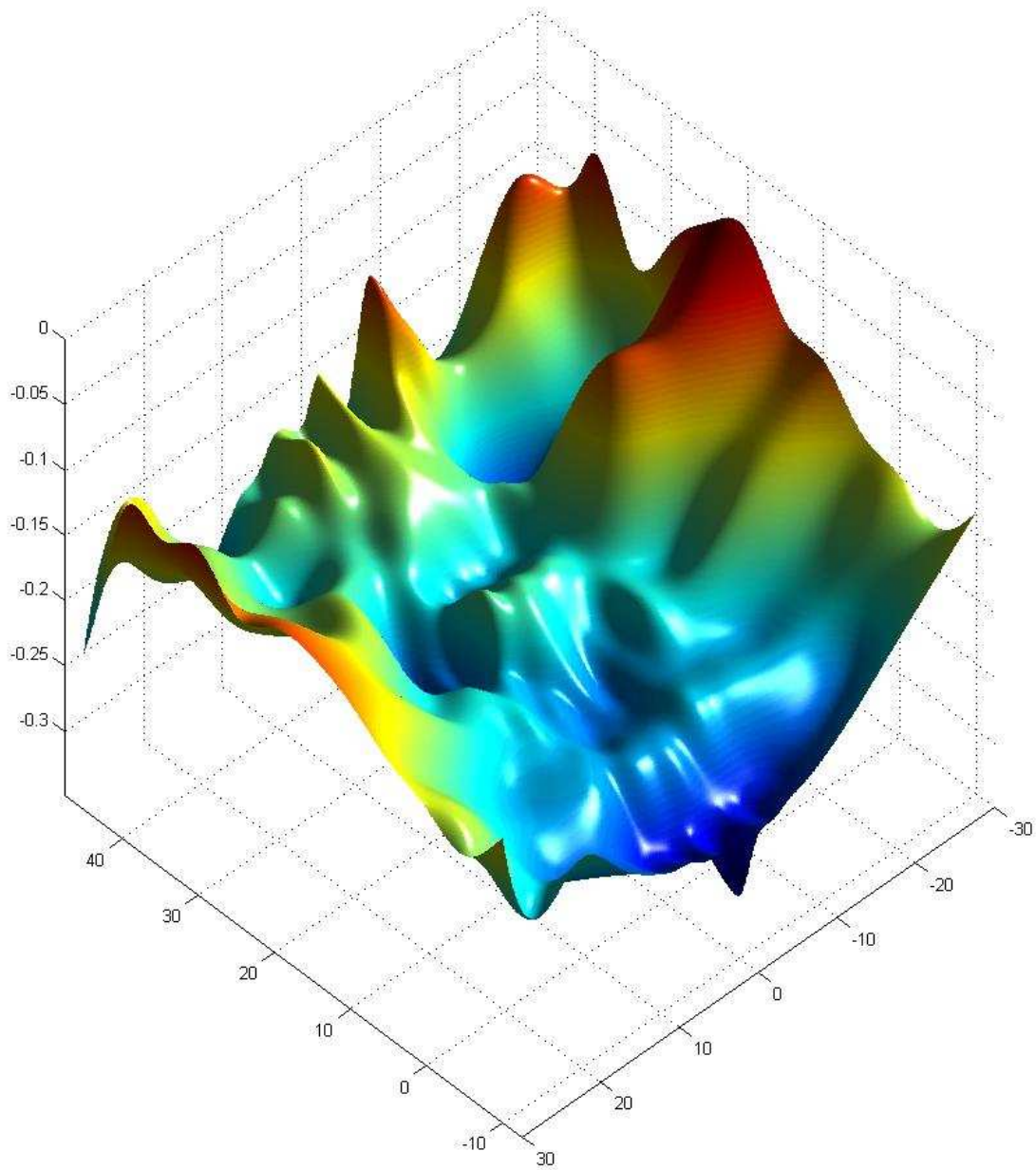


Рис. 6.8. Сечение функции $m_z = m_z(\alpha, \beta, \delta_v, \omega_z)$ по углу атаки α (диапазон изменения от -10° до 45°) и углу скольжения β (диапазон изменения от -30° до 30°)

— размах крыла и средняя аэродинамическая хорда, м; \bar{q} — скоростной напор, Па. Здесь также $C_x, C_y, C_z, m_x, m_y, m_z$ — безразмерные коэффициенты продольной, нормальной и поперечной сил, моментов крена, рысканья и тангажа, представляющие собой нелинейные функции своих аргументов, перечисленных в (6.7) и (6.8).

Следует отметить, что восстанавливаемые зависимости коэффициентов аэродинамических сил и, особенно, моментов, имеют достаточно сложный характер, что существенно затрудняет процесс идентификации АДХ для маневренного самолета, у которого значения аргументов соответствующих функций изменяются в широких диапазонах. В качестве примера на рис. 6.8

показано сечение рельефа, образуемого функцией $m_z = m_z(\alpha, \beta, \delta_v, \omega_z)$ по углу атаки α (диапазон изменения от -10^0 до 45^0) и углу скольжения β (диапазон изменения от -30^0 до 30^0).

4. В качестве примера конкретного объекта моделирования рассматривался маневренный самолет F-16. Исходные данные для него были взяты из работы [44], в которой приводятся экспериментальные результаты, полученные путем продувок в аэродинамических трубах и в летных испытаниях.

В проводившихся вычислительных экспериментах были приняты следующие конкретные значения соответствующих величин в (6.2)–(6.9): масса самолета $m = 9295.44$ кг; размах крыла $l = 9.144$ м; площадь крыла $S = 27.87$ м²; средняя аэродинамическая хорда крыла $b_A = 3.45$ м; моменты инерции $J_x = 12874.8$ кг·м², $J_y = 85552.1$ кг·м², $J_z = 75673.6$ кг·м², $J_{xy} = -1331.4$ кг·м², $J_{yz} = J_{zx} = 0$ кг·м²; центровка 5% САХ; постоянные времени приводов $T_v = T_n = T_s = 0.025$ с; коэффициенты относительного демпфирования приводов $\xi_v = \xi_n = \xi_s = 0.707$; угол отклонения предкрылков $\delta_{LEF} = 0$ град.

За время протекания переходных процессов в угловом движении самолета такие величины, характеризующие его траекторное движение, как скорость V_T и высота h полета не успевают существенно измениться. В силу этого, уравнения, описывающие траекторное движение, в рассматриваемую модель не включены. Значение скорости V_T и высоты h полета, требуемые для вычисления сил и моментов, действующих на самолет, принимаются постоянными. В проводившихся экспериментах они были приняты следующими: высота над уровнем моря $h = 3000$ м; воздушная скорость $V_T = 147.8593$ м/с. Соответственно, остальные величины, зависящие от V_T и h , принимают значения: ускорение силы тяжести $g = 9.8066$ м/с²; плотность воздуха $\rho = 0.8365$ кг/м³; местная скорость звука $a = 328.5763$ м/с; число Маха $M = 0.45$; скоростной напор $\bar{q} = 9143.6389$ Па.

5. В модели (6.2)–(6.5) величины $\omega_x, \omega_y, \omega_z, \gamma, \psi, \vartheta, \alpha, \beta, \delta_v, \delta_n, \delta_s, \dot{\delta}_v, \dot{\delta}_n, \dot{\delta}_s$ — это состояния объекта управления, величины $\delta_{v_{act}}, \delta_{n_{act}}, \delta_{s_{act}}$ — управления. На значения управляющих величин наложены ограничения: $\delta_{v_{act}} (\pm 25^\circ), \delta_{n_{act}} (\pm 30^\circ), \delta_{s_{act}} (\pm 21.5^\circ)$ — для командных сигналов на приводы управляемого стабилизатора, руля направления и элеронов. При формировании обучающего набора, а также при тестировании полученной полуэмпирической НС-модели управляющие воздействия на самолет осуществлялись одновременно по всем трем каналам, при этом сигналы $\delta_{v_{act}}, \delta_{n_{act}}, \delta_{s_{act}}$ формировались как полигармонические для получения обучающего набора и как случайные — при тестировании обученной модели.

Вычислительный эксперимент с моделью (6.2)–(6.5) проводился для временного интер-

вала $t \in [0, 20]$ с при получении данных для обучения НС-модели и $t \in [0, 40]$ с при тестировании полученной НС-модели, в обоих случаях с шагом дискретизации $\Delta t = 0.02$ с для частично наблюдаемого вектора состояния $y(t) = [\alpha(t); \beta(t); \omega_x(t); \omega_y(t); \omega_z(t)]^T$. На выход системы $y(t)$ воздействует аддитивный белый шум со среднеквадратичным отклонением (СКО) $\sigma_\alpha = \sigma_\beta = 0.02$ град, $\sigma_{\omega_x} = 0.1$ град/с, $\sigma_{\omega_y} = \sigma_{\omega_z} = 0.05$ град/с.

Как и в предыдущем примере, за целевое значение ошибки моделирования принимается значение СКО шума, воздействующего на выход системы. Обучение на выборке $\{y_i\}$, $i = 1, \dots, N$, полученной с помощью исходной модели (6.2)–(6.5), проводится в системе Matlab для сетей в форме LDDN (Layered Digital Dynamic Networks) с использованием алгоритма Левенберга-Марквардта по критерию среднеквадратичной ошибки модели. Матрица Якоби вычисляется по алгоритму RTRL (Real-Time Recurrent Learning) [74]. Стратегия обучения НС-модели была основанной на сегментировании обучающей последовательности, рассмотренном в разд. 2.4.1.5.

6. Структурную схему полуэмпирической модели, отвечающую исходной системе (6.2)–(6.5), не представляется возможным показать в силу ее громоздкости. Однако такая схема для случая системы (6.2)–(6.5) в целом аналогична схеме, показанной на рис. 6.5, но включает, разумеется, значительно большее число элементов и связей между ними, входящих в неизменяемую часть НС-модели, т. е. в ту ее часть, которая реализует теоретические знания об объекте моделирования. Кроме того, число НС-модулей типа «черный ящик», реализующих неизвестные восстанавливаемые зависимости коэффициентов аэродинамических сил и моментов, в НС-модели, отвечающей системе (6.2)–(6.5), увеличивается до пяти (C_y, C_z, m_x, m_y, m_z) по сравнению с двумя (C_y, m_z) для системы (6.1).

7. Как видно из соотношений (6.6), в НС-модели должна быть реализована также и зависимость для коэффициента продольной силы C_x . Эта зависимость также включается в НС-модель как модуль типа «черный ящик», но реализуемая им нелинейная функция не входит в число восстанавливаемых при формировании и обучении полуэмпирической модели. Соответствующий НС-модуль (все НС-модули, как для функций C_y, C_z, m_x, m_y, m_z , так и для функции C_x формируются в виде сигмоидальной НС прямого распространения с одним скрытым слоем) формируется автономно на основе данных из [44], после чего данный модуль вставляется в формируемую полуэмпирическую НС-модель и «замораживается», т. е. на варьирование его настраиваемых параметров (синаптических весов и смещений) накладывается запрет. Такой подход обусловлен следующими соображениями. Такие траекторные

параметры движения самолета, как высота h и скорость полета V_T , за время, которое занимает переходный процесс в короткопериодическом угловом движении, не успевают существенно измениться, поэтому при анализе только углового движения можно принять, что $h = \text{const}$ и $V_T = \text{const}$ и понизить порядок используемой системы дифференциальных уравнений за счет исключения из них уравнений, описывающих траекторное движение самолета. Это, однако, приводит к тому, что пропадает возможность восстановления зависимости для C_x совместно и одновременно с зависимостями для C_y, C_z, m_x, m_y, m_z , поскольку нет возможности получить обучающие данные для C_x , формируемые как реакция параметров траекторного движения самолета на соответствующие управляющие воздействия — это, в первую очередь, изменение тяги двигательной установки самолета и, возможно, отклонение воздушных тормозов.

8. Если расширить исходную теоретическую модель (6.2)–(6.5), добавив в нее уравнения, описывающие траекторное движение самолета, появляется возможность восстанавливать в ходе формирования полуэмпирической НС-модели сразу все шесть функций $C_x, C_y, C_z, m_x, m_y, m_z$. Эта задача в целом аналогична рассматриваемой (с отдельно формируемым НС-модулем для C_x), хотя процесс решения ее и будет несколько более трудоемким из-за увеличившейся размерности. Вообще говоря, модель движения ЛА может включать в себя в качестве восстанавливаемых при ее обучении зависимостей не только аэродинамические характеристики ЛА, но и любые другие функциональные зависимости, точный вид которых неизвестен. Чтобы можно было восстановить вид и этих зависимостей, требуются экспериментальные данные о том, как величина, описываемая данной зависимостью, реагирует на изменения ее переменных-аргументов.

9. Как уже отмечалось, для обеспечения адекватности создаваемой полуэмпирической НС-модели требуется формирование репрезентативного обучающего набора, отражающего специфику поведения моделируемого объекта, для управляющих воздействий, принимающих значения из заданного диапазона. Введение ограничений на значения управляющих воздействий как следствие порождает также и ограниченность значений переменных состояния, описывающих объект. Об адекватности полученной модели, т.е. о наличии у нее обобщающих свойств, обеспечивающих требуемую точность моделирования поведения объекта, можно с уверенностью говорить лишь в той области значений управляющих переменных и переменных состояния, которая образована упомянутыми выше ограничениями.

В проводившихся вычислительных экспериментах управляющие переменные $\delta_{\text{вact}}, \delta_{\text{нact}}, \delta_{\text{эact}}$ при обучении (полигармонический входной сигнал) и тестировании (случайный входной

ТАБЛИЦА 6.4. Диапазоны изменения переменных в модели (6.2)–(6.5)

	Обучающий набор		Тестовый набор	
	min	max	min	max
α	3.8405	6.3016	3.9286	5.8624
β	-1.9599	1.7605	-0.4966	0.9754
ω_x	-16.0310	18.1922	-10.1901	11.8683
ω_y	-4.1017	4.6205	-4.1661	0.9682
ω_z	-3.0298	3.1572	-1.2555	3.6701
δ_B	-7.2821	-4.7698	-7.2750	-5.0549
$\dot{\delta}_B$	-8.1746	8.0454	-39.4708	36.8069
δ_Θ	-1.2714	1.2138	-2.0423	1.0921
$\dot{\delta}_\Theta$	-8.6386	8.7046	-56.8323	48.9997
δ_H	-2.5264	1.7844	-1.7308	1.4222
$\dot{\delta}_H$	-20.4249	17.8579	-48.6391	58.5552
ψ	0	11.9927	-98.5980	0.0099
ϑ	0	5.3013	-20.8143	3.8094
γ	-22.3955	7.7016	0	59.6928
δ_{Bact}	-7.2629	-4.7886	-7.0105	-5.3111
$\delta_{\Theta act}$	-1.2518	1.1944	-1.4145	0.7694
δ_{Hact}	-2.4772	1.7321	-1.3140	1.0044

сигнал) формируемой модели изменялись в пределах, указанных в табл. 6.4. Здесь же показаны и граничные значения переменных состояния ω_x , ω_y , ω_z , γ , ψ , ϑ , α , β , которые отвечают данным ограничениям на управления.

Расширение области изменения управляющих и фазовых переменных вплоть до полной эксплуатационной области, присущей моделируемому объекту, требует развития алгоритмов формирования модели. Один из подходов к решению данной проблемы, активно развиваемый в настоящее время, состоит в придании как НС-модели, так алгоритму ее обучения свойств инкрементности [102, 103]. Для такого рода модели вначале формируется ее ядро, работающее только на части эксплуатационной области, а затем выполняются несколько шагов расширения области действия модели, причем на каждом из них модель, полученная ранее, не изменяется (или подвергается небольшой корректировке), а лишь расширяет свою область

адекватности.

Данный алгоритм был успешно применен к сформулированной выше задаче идентификации аэродинамических коэффициентов C_y, C_z, m_x, m_y, m_z для выполнения прогноза на 1000 шагов. Результаты решения данной задачи представлены в табл. 6.5, на рис. 6.9 и рис. 6.10.

Таблица 6.5. Ошибка моделирования на тестовом множестве для полуэмпирической модели на различных стадиях обучения

Кол-во шагов прогноза	СКО $_{\alpha}$	СКО $_{\beta}$	СКО $_{\omega_x}$	СКО $_{\omega_y}$	СКО $_{\omega_z}$
2	0.1376	0.2100	1.5238	0.4523	0.4517
4	0.1550	0.0870	0.5673	0.2738	0.4069
6	0.1647	0.0663	0.4270	0.2021	0.3973
9	0.1316	0.0183	0.1751	0.0530	0.2931
14	0.0533	0.0109	0.1366	0.0300	0.1116
21	0.0171	0.0080	0.0972	0.0193	0.0399
1000	0.0171	0.0080	0.0972	0.0193	0.0399

Анализ полученных результатов моделирования позволяет сделать следующие выводы.

Определяющими для сформированной модели являются ее обобщающие свойства, под которыми традиционно для НС-модели понимается ее способность обеспечивать требуемый уровень точности не только для данных, на которых модель обучалась, но и для любых значений и сочетаний значений управляющих и фазовых переменных в пределах области их определения. Такого рода проверка осуществляется на тестовых данных, покрывающих упомянутую область определения и не совпадающих с обучающими данными. Успешно решенная задача идентификации и моделирования должна, во-первых, обеспечить получение требуемой точности моделирования на всей области определения модели и, во-вторых, обеспечить восстановление с заданной точностью аэродинамических характеристик ЛА как соответствующих нелинейных функциональных зависимостей. Насколько успешно решается первая из этих задач, можно судить по результатам, представленным на рис. 6.9, а также в табл. 6.5. Из рис. 6.9 видно, во-первых, что ошибки по всем наблюдаемым переменным состояния незначительны и во-вторых, что эти ошибки с течением времени практически не растут, что свидетельствует о хороших обобщающих свойствах полученной НС-модели — модель не «разваливается» при достаточно долгом прогоне. Тестирование проводилось на интервале в 40 с — достаточно большое время для короткопериодического движения ЛА. При этом

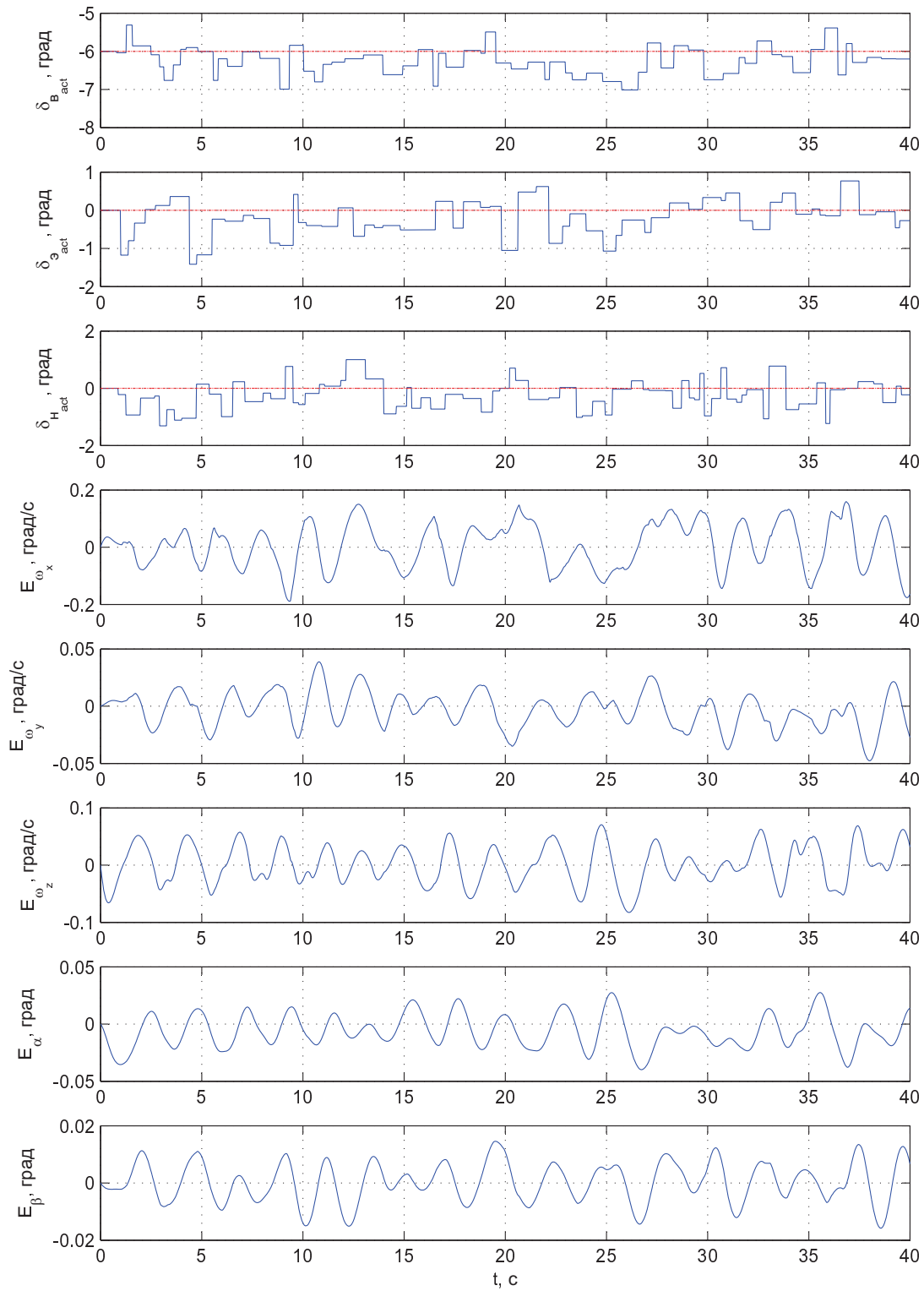


Рис. 6.9. Оценка обобщающей способности НС-модели после завершающего 1000-шагового этапа обучения: E_α , E_β , E_{ω_x} , E_{ω_y} , E_{ω_z} — погрешность воспроизведения соответствующих наблюдаемых величин; прямыми линиями на трех верхних графиках показаны значения управляющих величин, соответствующие тестовому маневру

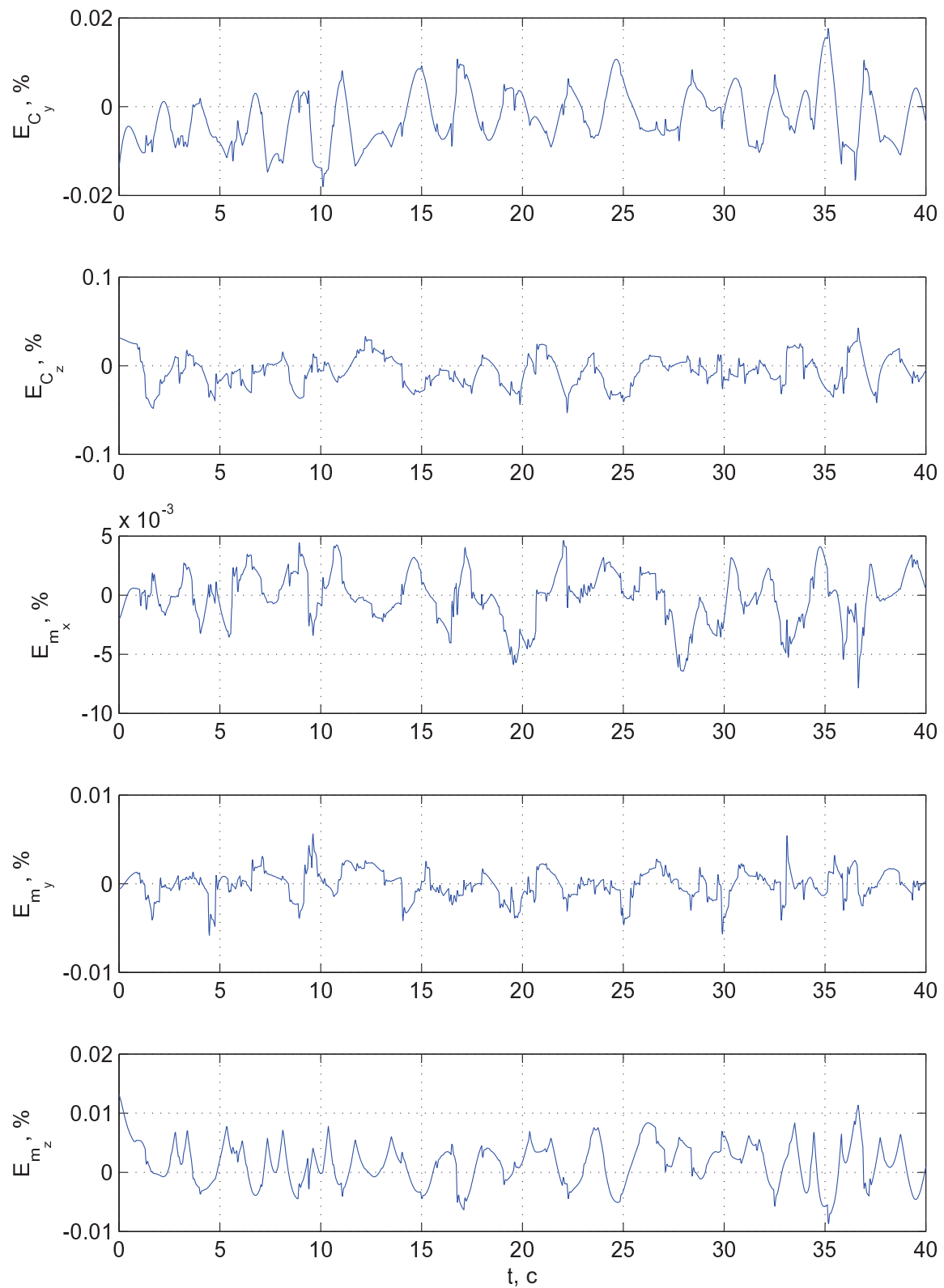


Рис. 6.10. Значения ошибки воспроизведения величин C_y , C_z , m_x , m_y , m_z согласно восстановленным зависимостям для них в процессе тестирования полуэмпирической модели (отнесены к диапазонам изменения этих величин, полученным при тестировании)

следует подчеркнуть, что тестирование модели осуществлялось в весьма жестком режиме. Из рис. 6.9 видно, что реализуется очень активная работа органами управления ЛА (управляемый стабилизатор, руль направления, элероны), выражающаяся в частом изменении значения командных сигналов $\delta_{вact}$, $\delta_{нact}$, $\delta_{эact}$ приводов органов управления при существенных перепадах между соседними значениями (эти командные сигналы формировались случайным образом). Цель применения такого способа формирования обучающего набора состоит в том, чтобы обеспечить возможно большее разнообразие состояний моделируемой системы (чтобы покрыть по-возможности равномерно и плотно все пространство состояний системы), а также возможно большее разнообразие перепадов соседних по времени состояний (чтобы максимально достоверно отразить в НС-модели динамику моделируемой системы). Дополнительный осложняющий фактор состоит в том, что очередное возмущающее воздействие на ЛА выдается на фоне еще не завершившихся переходных процессов от одного или нескольких предыдущих воздействий. Рис. 6.9 характеризует модель, для которой обучающий цикл, описанный выше, уже завершен. О том, как менялись точностные характеристики данной модели в промежуточных точках этого цикла можно судить по данным, представленным в табл. 6.5. Точность формируемой модели определяется тем, насколько точно восстановлены нелинейные функции, описывающие аэродинамические характеристики ЛА. Данные, приведенные на рис. 6.9, характеризуют интегральное влияние точности восстановления функций на точность модели в целом, которую можно признать вполне удовлетворительной. Представляет, однако, интерес также и вопрос о том, насколько точно решена собственно задача идентификации аэродинамических характеристик ЛА. Для получения ответа на этот вопрос можно извлечь НС-модули, отвечающие восстановленным функциям для C_y , C_z , m_x , m_y , m_z , а затем сравнить выдаваемые ими значения с имеющимися экспериментальными данными [44]. При этом можно получить значения среднеквадратической ошибки воспроизведения каждой из функций C_y , C_z , m_x , m_y , m_z соответствующим НС-модулем. В проведенных экспериментах эти значения составляют: $СКО_{C_y} = 9.2759E-04$, $СКО_{C_z} = 5.4257E-04$, $СКО_{m_x} = 2.1496E-05$, $СКО_{m_y} = 1.3873E-05$, $СКО_{m_z} = 1.4952E-04$. Это интегральная оценка точности восстановления упомянутых зависимостей. Кроме нее, представляет интерес также динамика изменения текущих значений ошибки воспроизведения величин C_y , C_z , m_x , m_y , m_z в процессе тестирования модели. Эти данные, приведенные на рис. 6.10, показывают, что уровень ошибки по времени изменяется незначительно, существенных изменений в нем, которые могли бы отрицательно повлиять на адекватность эмпирической НС-модели, не обнаруживается.

Заключение

Как отмечалось во введении, объектом исследования в данной диссертационной работе являются адаптивные динамические системы, действующие в условиях разнообразных неопределенностей, в частности, ЛА различных классов. Применительно к такого рода системам предмет исследования диссертации — адаптивные законы управления многомерных нелинейных управляемых ДС, а также адаптивные модели таких ДС, обладающие высокой точностью и быстродействием; Полученные результаты позволяют сделать вывод о том, что адаптивное управление, реализованное в таких системах средствами нейросетевых технологий, представляет собой эффективный инструмент управления ДС в условиях неопределенности, включая непредвиденные изменения свойств ДС вследствие отказов ее оборудования и повреждений конструкции.

Вместе с тем, применительно к задачам, связанным с управлением движения ЛА, в адаптивных ДС, основанных на НС-технологиях, критически важную роль начинают играть модели ДС как объектов управления. Традиционные НС-технологии, реализующие чисто эмпирический подход к моделированию и трактующие в соответствии с ним модель ДС как «черный ящик», не позволяют получить НС-модели движения ЛА, обладающие точностью, достаточной для эффективной реализации схем адаптивного управления. Кроме того, такие НС-технологии не дают возможности решать задачи идентификации характеристик ЛА, что является их существенным недостатком.

В связи с этим, в качестве цели диссертационной работы было принято формирование нового подхода к математическому и компьютерному моделированию адаптивных ДС, свободного от недостатков традиционных подходов. Для преодоления указанных затруднений в диссертации введена концепция полуэмпирического моделирования, которая приводит к получению НС-моделей гибридного типа (моделей типа «серый ящик»), сочетающих как эмпирические, так и теоретические знания о ДС. Полученные результаты позволяют сделать вывод о том, что методы полуэмпирического НС-моделирования, сочетающие знания и опыт из соответствующей предметной области, а также из традиционного вычислительного моделирования, с методами и средствами формирования и обучения искусственных нейронных сетей, являются мощным и перспективным инструментом, потенциально пригодным для решения сложных задач описания и анализа управляемого движения летательных аппаратов.

Развитие предлагаемого класса моделей в направлении придания им свойств адаптивности

за счет применения вставочных подсетей, а также методов и средств инкрементного обучения позволяет формировать модели, адекватные в широком диапазоне условий их применения, что дает возможность эффективно использовать их в условиях неопределенности.

Сформированный подход обеспечивает решение задач анализа поведения, синтеза управления и идентификации неизвестных характеристик ДС, возникающих при создании и эксплуатации сложных технических систем, в частности, роботизированных пилотируемых и беспилотных ЛА.

Предложенный в диссертации подход к структуризации НС-моделей ориентирован на развитие полученных методов в направлении автоматизации процессов формирования НС-моделей под заданный комплекс требований к ним. Развитие данного подхода открывает возможность эффективно решать разнообразные задачи синтеза адаптивных и интеллектуальных систем управления поведением летательных аппаратов различных классов, включая перспективные пилотируемые и беспилотные ЛА, обеспечивая повышение безопасности полета и живучести для них, а также существенное повышение эффективности решения ими соответствующих целевых задач.

В процессе выполнения диссертационной работы для достижения сформулированной выше цели были решены следующие задачи:

1. С позиций общей теории систем построена классификация ДС по их существенным признакам (типология ДС), а также выявлены варианты среды, с которой взаимодействуют ДС, что обеспечивает единый контекст для решения задач анализа поведения, синтеза управления и идентификации характеристик при создании перспективных технических систем.
2. Разработан нейросетевой подход к обеспечению адаптивности ДС (в таких элементах как закон управления ДС и модель ДС) за счет ансамблевой архитектуры используемых нейронных сетей (НС), введения в НС вставочных подсетей и использования инкрементного обучения НС.
3. Разработан класс математических и компьютерных моделей, представляющих собой НС-модели гибридного типа, обладающие адаптивностью, объединяющие теоретические знания о ДС и экспериментальные данные о ее поведении.
4. Разработано унифицированное структурное (модульное) описание НС-моделей, обеспечивающее единообразное представление всех видов статических и динамических сетей, позволяющее автоматизировать процесс синтеза НС-моделей.

5. Разработан композиционный подход к синтезу (формированию) НС-моделей, основанный на интерпретации НС-модели как разложения по обобщенному функциональному базису (ФБ специального вида).
6. Систематизированы имеющиеся результаты в области моделирования ДС с использованием НС традиционного типа (модели типа «черный ящик»), выявлены ограничения и область возможного использования этих средств.
7. Систематизированы и обобщены имеющиеся результаты в области моделирования ДС с использованием моделей типа «серый ящик», выявлены перспективные направления развития данной области.
8. Разработаны алгоритмы формирования полуэмпирических НС-моделей как одного из классов моделей типа «серый ящик» на основе теоретического знания (в виде систем обыкновенных дифференциальных или дифференциально-алгебраических уравнений) об объекте моделирования, а также экспериментальных данных о его поведении.
9. Разработаны методы и алгоритмы получения обучающих данных для НС-моделей динамических систем.
10. Разработаны алгоритмы параметрической настройки (обучения) полуэмпирических НС-моделей.
11. В серии вычислительных экспериментов проведена верификация разработанных полуэмпирических НС-моделей и методов их формирования.
12. В серии вычислительных экспериментов проведено формирование эмпирических и полуэмпирических НС-моделей для объектов различных классов, проведено сопоставление свойств этих двух классов моделей.
13. Разработан подход к решению задачи идентификации характеристик ДС, описываемых нелинейными функциями многих переменных. Работоспособность этого подхода продемонстрирована на примере задачи идентификации аэродинамических коэффициентов ЛА.
14. В рамках предложенного подхода решено значительное число задач адаптивного управления для ЛА различных классов, в которых модель объекта и закон управления были реализованы в виде динамических НС-структур.

При решении этих задач были получены следующие новые научные результаты, выносимые на защиту:

- сформирована типология ДС, обеспечивающая единый контекст для решения задач анализа поведения, синтеза управления и идентификации характеристик при создании перспективных технических систем различных классов;
- разработано унифицированное структурное описание НС-моделей, обеспечивающее единообразное представление всех видов статических и динамических сетей, позволяющее автоматизировать процесс синтеза НС-моделей;
- разработан композиционный подход к синтезу статических и динамических НС-моделей, основанный на интерпретации НС-модели как разложения по обобщенному функциональному базису;
- разработан класс гибридных нейросетевых моделей полуэмпирического типа, объединяющий теоретические знания об объекте моделирования и экспериментальные данные о его поведении;
- разработаны алгоритмы формирования гибридных нейросетевых моделей полуэмпирического типа, а также алгоритмы их структурной корректировки и параметрической настройки;
- разработаны методы и алгоритмы получения обучающих данных для НС-моделей динамических систем;
- разработан подход к решению задачи идентификации характеристик ДС как нелинейных функций многих переменных.

Практическая значимость полученных результатов состоит в том, что появляется возможность создания перспективных роботизированных пилотируемых и беспилотных ЛА различного назначения, предназначенных для решения сложных целевых задач и комплексов таких задач в условиях неопределенности на основе использования адаптивных и интеллектуальных систем управления поведением ЛА. Роботизация ЛА, осуществляемая на этой основе, позволит значительно повысить их эффективность при решении целевых задач, а также выживаемость в неблагоприятных условиях.

Полученные результаты могут быть использованы разработчиками перспективных технических систем, включая роботизированные ЛА, при формировании алгоритмов адаптивного и интеллектуального управления поведением этих систем, а также анализа их поведения и решения задач идентификации характеристик ЛА.

Список литературы

1. Авиационные правила. Часть 23. Нормы летной годности гражданских легких самолетов. – М.: Межгосударственный авиационный комитет, 2013. – 210 с.
2. Авиационные правила. Часть 25. Нормы летной годности самолетов транспортной категории. – М.: Межгосударственный авиационный комитет, 2014. – 278 с.
3. Аэродинамика и динамика полета магистральных самолетов / Под ред. Г. С. Бюшгенса. – Москва-Пекин: Издат. отдел ЦАГИ; Авиа-издательство КНР, 1995. – 772 с.
4. Аэродинамика, устойчивость и управляемость сверхзвуковых самолетов / Под ред. Г. С. Бюшгенса. – М.: Наука, 1998. – 816 с.
5. Cook M. V. On the design of command and stability augmentation systems for advanced technology aeroplanes // *Trans. of the Inst. of Measurement and Control*. – 1999. – Vol. 21, No. 2/3. – pp. 85–98.
6. Глумов В. М., Земляков С. Д., Рутковский В. Ю., Силаев А. В. Алгоритмическое обеспечение отказоустойчивости систем автоматического управления: Обзор // *Автоматика и телемеханика*. – 1988. – №9. – с.3–33.
7. Земляков С. Д., Рутковский В. Ю., Силаев А. В. Реконфигурация систем управления летательными аппаратами при отказах: Обзор // *Автоматика и телемеханика*. – 1996. – №1. – с.3–20.
8. Blanke M., Kinnaert M., Lunze J., Staroswiecki M. *Diagnosis and fault-tolerant control*: 2nd Ed. – Springer, 2006. – xviii+672 pp.
9. Richter J. H. *Reconfigurable control of nonlinear dynamical systems: A fault-hiding approach*. – Springer, 2011. – xv+291 pp.
10. Tomayko J. E. *The story of self-repairing control systems / Dryden Historical Study No. 1*. – NASA Dryden Flight Research Center, 2003. – 154 pp.
11. Witczak M. *Modelling and estimation strategies for fault diagnosis of non-linear systems: From analytical to soft-computing approaches*. – Springer, 2007. – xv+208 pp.
12. Austin R. *Unmanned aircraft systems: UAVs design, development and deployment*. – John Wiley & Sons, 2010. – xxx+332 pp.
13. *Introduction to unmanned aircraft systems / Barnhart R. W., Hoffman S. B., Marshall D. M., Shappee E. (Eds.)*. – CRC Press, 2012. – xvii+194 pp.

14. *Beard R. W., McLain T. W.* Small unmanned aircraft: Theory and practice. – Princeton University Press, 2012. – xvii+300 pp.
Рус. перевод: *Биард Р. У., МакЛэйн Т. У.* Малые беспилотные летательные аппараты: теория и практика. – М.: Техносфера, 2015. – 312 с.
15. *Intelligent unmanned systems / Budiyo A., Riyanto B., Joelianto E. (Eds.).* – Springer, 2009. – x+275 pp.
16. *Castillo P., Lozano R., Dzul A. E.* Modeling and control of mini-flying machines. – Springer, 2005. – xvi+251 pp.
17. *Ducard P. J. J.* Fault-tolerant flight control and guidance systems: Practical methods for small unmanned aerial vehicles. – Springer, 2009. – xxi+266 pp.
18. *Finn A., Scheduling S.* Developments and challenges for autonomous unmanned vehicles: A compendium. – Springer, 2010. – xv+228 pp.
19. *Gundlach J.* Designing unmanned aircraft systems: A comprehensive approach. – AIAA, 2012. – 870 pp.
20. *Nonami K., Kendoul F., Suzuki S., Wang W., Nakazawa D.* Autonomous flying robots: Unmanned aerial vehicles and micro aerial vehicles. – Springer, 2010. – xvii+329 pp.
21. Современные информационные технологии в задачах навигации и наведения беспилотных маневренных летательных аппаратов / Под ред. *М. Н. Красильщикова и Г. Г. Себрякова.* – М.: Физматлит, 2009. – 556 с.
22. *Брусов В. С., Петручик В. П., Морозов Н. И.* Аэродинамика и динамика полета малоразмерных беспилотных летательных аппаратов / Под ред. *В. С. Брусова.* – М.: МАИ-Принт, 2010. – 340 с.
23. *Low Reynolds number aerodynamics and transition / Genç M. S. (Ed.).* – InTech, 2012. – ix+162 pp.
24. *Shyy W., Lian Y., Tang J., Viieru D., Liu H.* Aerodynamics of low Reynolds number flyers. – Cambridge University Press, 2008. – 213 pp.
25. *Калман Р., Фалб П., Арбиб М.* Очерки по математической теории систем: Пер. с англ. под ред. *Я. З. Цыпкина.* – М.: Едиториал УРСС, 2004. – 400 с.
26. *Месарович М., Такахара Я.* Общая теория систем: Математические основы: Пер. с англ. под ред. *С. В. Емельянова.* – М.: Мир, 1978. – 312 с.
27. *Каток А. Б., Хассельблат Б.* Введение в современную теорию динамических систем: Пер. с англ. – М.: Факториал, 1999. – 768 с.

28. *Арнольд В. И.* Математические методы классической механики: 3-е изд. – М.: Наука, 1989. – 472 с.
29. *Ljung L., Glad T.* Modeling of dynamic systems. – Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1994. – 361 pp.
30. *Holmgren R. A.* A first course in discrete dynamical systems. – New York ao: Springer, 1994. – xii+214 pp.
31. *Pearson R. K.* Discrete-time dynamic models. – New York–Oxford: Oxford University Press, 1999. – xiv+466 pp.
32. *Анищенко В. С.* Знакомство с нелинейной динамикой. – Москва-Ижевск: Институт компьютерных исследований, 2002. – 144 с.
33. *Малинецкий Г. Г., Потапов А. Б.* Современные проблемы нелинейной динамики. – Москва: Эдиториал УРСС, 2000. – 336 с.
34. *Халил Х. К.* Нелинейные системы: 3-е изд.: Пер. с англ. под. ред. *А. Л. Фрадкова.* – Москва-Ижевск: Институт компьютерных исследований, 2009. – 832 с.
35. *Лю Б.* Теория и практика неопределенного программирования: Пер. с англ. под ред. *Ю. В. Тюменцева.* – Москва: БИНОМ. Лаборатория знаний, 2005. – 416 с.
36. *Martyniuk A. A., Martyniuk-Chernenko Yu. A.* Uncertain dynamic systems: Stability and motion control. – London ao.: CRC Press, 2012. – xiii+295 pp.
37. *Пегат А.* Нечеткое моделирование и управление: Пер. с англ. под ред. *Ю. В. Тюменцева.* – Москва: БИНОМ. Лаборатория знаний, 2009. – 798 с.
38. *Горбатенко С. А., Макашов Э. М., Полушкин Ю. Ф., Шефтель Л. В.* Расчет и анализ движения летательных аппаратов: Инженерный справочник. – М.: Машиностроение, 1971. – 352 с.
39. Аэромеханика самолета: Динамика полета – 2-е изд., перераб. и доп. / *А. Ф. Бочкарев, В. В. Андреевский, В. М. Белоконов и др.* Под ред. *А. Ф. Бочкарева и В. В. Андреевского.* – М.: Машиностроение, 1985. – 360 с.
40. Динамика полета / *А. В. Ефремов, В. Ф. Захарченко, В. Н. Овчаренко, В. Л. Суханов* / Под ред. *Г. С. Бюшгенса.* – М.: Машиностроение, 2011. – 776 с.
41. *Николаев Л. Ф.* Аэродинамика и динамика полета транспортных самолетов. – М.: Транспорт, 1990. – 392 с.

42. *Доброленский Ю. П.* Динамика полета в неспокойной атмосфере. – М.: Машиностроение, 1969. – 256 с.
43. *Школьный Е. П., Майборода Л. А.* Атмосфера и управление движением летательных аппаратов / Под ред. *В. М. Пономарева.* – Л.: Гидрометеиздат, 1973. – 308 с.
44. *Nguyen L. T., Ogburn M. E., Gilbert W. P., Kibler K. S., Brown P. W., Deal P. L.* Simulator study of stall/post-stall characteristics of a fighter airplane with relaxed longitudinal static stability. – NASA TP-1538, Dec. 1979. – 223 pp.
45. *Davidson J., Lallman J. D. a.o.* Flight control laws for NASA's Hyper-X research vehicle. – *AIAA-99-4124.* – 11 pp.
46. *Engelund W. C., Holland S. D. a.o.* Propulsion system airframe integration issues and aerodynamic database development for the Hyper-X flight research vehicle. – *ISOABE-99-7215.* – 12 pp.
47. *Shaughnessy J. D., Pinckney S. Z. a.o.* Hypersonic vehicle simulation model: Winged-cone configuration. – *NASA-TM-102610.* – November 1990. – 142 pp.
48. *Ильин В. Е., Левин М. А.* Истребители. – М.: Виктория, АСТ, 1997. – 288 с.
49. *Глазунов Ю. Т.* Моделирование целеполагания – М.–Ижевск: НИЦ «Регулярная и хаотическая динамика», Институт компьютерных исследований, 2012. – 216 с.
50. *Красовский А. А.* Системы автоматического управления полетом и их аналитическое конструирование. – М.: Наука, 1973. – 560 с.
51. Справочник по теории автоматического управления / Под ред. *А. А. Красовского.* – М.: Наука, 1987. – 712 с.
52. *Растрюгин Л. А.* Адаптация сложных систем: Методы и приложения. – Рига: Зинатне, 1981. – 375 с.
53. *Сигеру О., Марзуки К., Рубия Ю.* Нейроуправление и его приложения: Пер. с англ. под общ. ред. *А. И. Галушкина и В. А. Птичкина.* – М.: ИПРЖР, 2000. – 272 с. – (Серия «Нейрокомпьютеры и их применение». Кн. 2)
54. *Тюкин И. Ю., Терехов В. А.* Адаптация в нелинейных динамических системах. – М.: Изд-во ЛКИ, 2008. – 384 с. – (Серия «Синергетика: от прошлого к будущему»)
55. *Фомин В. Н., Фрадков А. Л., Якубович В. А.* Адаптивное управление динамическими объектами. – М.: Наука, 1981. – 441 с.

56. *Фрадков А. Л.* Адаптивное управление в сложных системах: Беспойсковые методы. – М.: Наука, 1990. – 296 с.
57. *Astolfi A.* Nonlinear and adaptive control: Tools and algorithms for the user. – London: Imperial College Press, 2006. – 312 pp.
58. *Astolfi A., Karagiannis D., Ortega R.* Nonlinear and adaptive control with applications. – Berlin a.o.: Springer, 2008. – 290 pp. – (Communications and Control Engineering)
59. *Gros C.* Complex and adaptive dynamical systems: A primer. – Berlin a.o.: Springer, 2008. – 260 pp. – (Springer Complexity)
60. *Ioannou P., Fidan B.* Adaptive control tutorial. – SIAM, 2006. – 403 pp.
61. *Ioannou P. A., Sun J.* Robust adaptive control. – Prentice Hall, 1995. – 848 pp.
62. *Mosca E.* Optimal, predictive, and adaptive control. – Englewood Cliffs, New Jersey: Prentice Hall, 1994. – 480 pp.
63. *Omidvar O. M., Elliott D. I.* (Eds.) Neural systems for control. – Academic Press, 1997. – 358 pp.
64. *Sastry S., Bodson M.* Adaptive control: Stability, convergence, and robustness. – Englewood Cliffs, New Jersey: Prentice Hall, 1989. – 377 pp. – (Prentice Hall Advanced Reference Series)
65. *Spooner J. T., Maggiore M., Ordóñez R., Passino K. M.* Stable adaptive control and estimation for nonlinear systems: Neural and fuzzy approximator techniques. – John Wiley & Sons, Inc., 2002. – 545 pp.
66. *Tao G.* Adaptive control design and analysis. – John Wiley & Sons, Inc., 2003. – 618 pp. – (Adaptive and Learning Systems for Signal Processing, Communication, and Control / Editor: *Simon Haykin*)
67. *Khalil H. K.* Nonlinear systems: 2nd Ed. – Prentice Hall, 1996. – 747 pp.
68. *Крутько П. Д.* Обратные задачи динамики в теории автоматического управления. – М.: Машиностроение, 2004. – 576 с.
69. *Ким Д. П.* Теория автоматического управления. Т. 1. Линейные системы. – М.: Физматлит, 2007. – 312 с.
70. *Ким Д. П.* Теория автоматического управления. Т. 2. Многомерные, нелинейные, оптимальные и адаптивные системы. – М.: Физматлит, 2004. – 464 с.

71. Глушков В. М., Цейтлин Г. Е., Ющенко Е. Л. Алгебра. Языки. Программирование: 2-е изд. – Киев: Наукова думка, 1978. – 320 с.
72. Оллонгрен А. Определение языков программирования интерпретирующими автоматами: Пер. с англ. под ред. В. Ш. Кауфмана. – Киев: Наукова думка, 1978. – 320 с.
73. Фу К. Структурные методы в распознавании образов: Пер. с англ. под ред. М. А. Айзермана. – М.: Мир, 1977. – 320 с.
74. Хайкин С. Нейронные сети: Полный курс: Под ред. Н. Н. Куссуль. – М.: Вильямс, 2006. – 1104 с.
75. Haykin S. Kalman filtering and neural networks / Ed. by S. Haykin. – New York a. o.: John Wiley & Sons, 2001. – 298 pp.
76. Dreyfus G. Neural networks: Methodology and applications. – Berlin ao.: Springer, 2005. – xviii+497 pp.
77. Hagan M. T., Demuth H. B., Beale M. H., De Jesús O. Neural network design: 2nd Ed. – PSW Publishing Co., 2014. – 800 pp.
78. Mandic D. P., Chambers J. A. Recurrent neural networks for prediction: Learning algorithms, architectures and stability. – New York ao.: John Wiley & Sons, 2001. – 297 pp.
79. Recurrent neural networks: Design and applications / Ed. by L. R. Medsker, L. C. Jain. – New York ao.: CRC Press, 2001. – 390 pp.
80. Neural network systems techniques and applications. Vol. 7: Control and dynamic systems / Ed. by C. T. Leondes. – New York ao.: Academic Press, 1999. – 459 pp.
81. Pinkus A. Approximation theory of the MLP model in neural networks. // *Acta Numerica*. – 1999. – Vol. 8. – pp. 143–195.
82. Терехов В. А., Ефимов Д. В., Тюкин И. Ю. Нейросетевые системы управления. – М.: ИПРЖР, 2002. – 480 с.
83. Tutschku K. Recurrent multilayer perceptrons for identification and control: The road to applications. – University of Würzburg, Institute of Computer Science, Research Report Series, Report No. 118, June 1995. – 28 pp.
84. Heister F., Müller R. An approach for the identification of nonlinear, dynamic processes with Kalman-filter-trained recurrent neural structures. – University of Würzburg, Institute of Computer Science, Research Report Series, Report No. 193, April 1999. – 29 pp.

85. Hochreiter S., Bengio Y., Frasconi P., Schmidhuber J. Gradient flow in recurrent nets: The difficulty of learning long-term dependencies // Chapter 14 in “A Field Guide to Dynamical Recurrent Networks” / Ed. by J. F. Kolen and S. C. Kremer. – IEEE Press: New York, 2001. – pp. 237–243.
86. Schaefer A. M., Udluft S., Zimmermann H.-G. Learning long-term dependencies with recurrent neural networks // *Neurocomputing*. – 2008. – Vol. 71, No. 13–15. – pp. 2481–2488.
87. Phan M. C., Hagan M. T. Error surface of recurrent neural networks // *IEEE Transactions on Neural Networks*. – 2009. – Vol. 24, No. 11. – pp. 1709–1721.
88. Horn J., De Jesús O., Hagan M. T. Spurious valleys in the error surface of recurrent networks — analysis and avoidance // *IEEE Transactions on Neural Networks*. – 2009. – Vol. 20, No. 4. – pp. 686–700.
89. Pascanu R., Mikolov T., Bengio Y. On the difficulty of training recurrent neural networks // *Proc. of the 30th Intern. Conf. on Machine Learning, ICML 2013, Atlanta, GA, USA, 16–21 June 2013*.
90. Elman J. L. Learning and development in neural networks: The importance of starting small // *Cognition*. – 1993. – Vol. 48. – pp. 71–99.
91. Ludik J., Cloete I. Incremental increased complexity training // *Proc. ESANN 1994, 2nd European Symposium on Artificial Neural Networks, Brussels, Belgium, April 20–22, 1994*.
92. Suykens J. A. K., Vandewalle J. Learning a simple recurrent neural state space model to behave like Chua’s double scroll // *IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications*. – 1995. – Vol. 42, No. 8. – pp. 499–502.
93. Bengio Y., Louradour J., Collobert R., Weston J. Curriculum learning // *Proc. of the 26th Annual Intern. Conf. on Machine Learning, ICML 2009, New York, NY, USA, pp. 41–48*.
94. Rumelhart D. E. *ao*. Learning internal representations by error propagation // In: “Parallel Distributed Processing” / Ed. by D. E. Rumelhart and J. L. McClelland. Vol. 1, 1986, p. 355.
95. Werbos P. J. Backpropagation through time: What it does and how to do it // *Proceedings of the IEEE*. – 1990. – Vol. 78, No. 10. – pp. 1550–1560.
96. Čerňanský M., Beňušková Ľ. Simple recurrent network trained by RTRL and extended Kalman filter algorithms // *Neural Network World*. – 2003. – Vol. 13, No. 3. – pp. 223–234.
97. Elman J. L. Finding structure in time // *Cognitive Science*. – 1990. – Vol. 14, No. 2. – pp. 179–211.
98. Williams R. J., Zipser D. A learning algorithm for continually running fully recurrent neural networks // *Neural Computation*. – 1989. – Vol. 1, No. 2. – pp. 270–280.

99. *De Jesús O., Hagan M. T.* Backpropagation algorithms for a broad class of dynamic networks // *IEEE Transactions on Neural Networks*. – 2007. – Vol. 18, No. 1. – pp. 14–27.
100. *Самарин А. И.* Нейронные сети с преднастройкой // В сб.: «Лекции по нейроинформатике», VII Всероссийская научно-технической конференция «Нейроинформатика-2005», Москва, МИФИ, 19–23 января 2015 г. – М.: Изд-во МИФИ, 2015. – с. 10–20.
101. *Горбань А. Н.* Обобщенная аппроксимационная теорема и вычислительные возможности нейронных сетей // *Сибирский журнал вычислительной математики*. – 1998. – т. 1, № 1. – с. 11–24.
102. *Dietterich T. G.* Machine-learning research: Four current directions // *AI Magazine*. – 1997. – Vol. 18, No. 7. – pp. 97–136.
103. *Joshi P., Kulkarni P.* Incremental learning: Areas and methods — a survey // *Intern. J. of Data Mining and Knowledge Management Process*. – 2012. – Vol. 2, No. 5. – pp. 43–51.
104. *Гладков Л. А., Курейчик В. В., Курейчик В. М.* Генетические алгоритмы: 2-е изд. – М.: Физматлит, 2006. – 320 с.
105. *Емельянов В. В., Курейчик В. М., Курейчик В. В.* Теория и практика эволюционного моделирования. – М.: Физматлит, 2003. – 432 с.
106. *Грон Д.* Методы идентификации систем: Пер. с англ. под ред. *Е. И. Кринецкого*. – М.: Мир, 1979. – 302 с.
107. *Льюнг Л.* Идентификация систем: Теория для пользователя: Пер. с англ. под ред. *Я. З. Цыпкина*. – М.: Наука, 1991. – 432 с.
108. *Сейдж Э. П., Мелса Л.* Идентификация систем управления: Пер. с англ. под ред. *Н. С. Райбмана*. – М.: Наука, 1974. – 248 с.
109. *Цыпкин Я. З.* Информационная теория идентификации. – М.: Наука, 1995. – 336 с.
110. *Isermann R., Münchhof M.* Identification of dynamic systems: An introduction with applications. – Berlin a.o.: Springer-Verlag, 2011. – xxv+705 pp.
111. *Juang J.-N., Phan M. Q.* Identification and control of mechanical systems. – Cambridge Univ. Press, 1994. – xv+334 pp.
112. *Nelles O.* Nonlinear system identification: From classical approaches to neural networks and fuzzy models. – Berlin a.o.: Springer-Verlag, 2001. – xvii+785 pp.

113. *Pintelon R., Schoukens J.* System identification: A frequency domain approach. – IEEE Press, 2001. – xxxviii+605 pp.
114. *Берестов Л. М., Поплавский Б. К., Мирошниченко Л. Я.* Частотные методы идентификации летательных аппаратов. – М.: Машиностроение, 1985. – 184 с.
115. *Васильченко К. К., Кочетков Ю. А., Леонов В. А., Поплавский Б. К.* Структурная идентификация математической модели движения самолета. – М.: Машиностроение, 1993. – 352 с.
116. *Jategaonkar R. V.* Flight vehicle system identification: A time domain methodology. – Reston, VA: AIAA, Inc., 2006. – xviii+534 pp.
117. *Klein V., Morelli E. A.* Aircraft system identification: Theory and practice. – Reston, VA: AIAA, Inc., 2006. – xiv+484 pp.
118. *Tischler M. B., Remple R. K.* Aircraft and rotorcraft system identification: Engineering methods with flight-test examples. – Reston, VA: AIAA, Inc., 2006. – xxxvi+523 pp.
119. *Hamel P. G., Jategaonkar R. V.* Evolution of flight vehicle system identification // *Journal of Aircraft.* – 1996. – v. 33, No. 1. – pp. 9–28.
120. *Hamel P. G., Kaletka J.* Advances in rotorcraft system identification // *Progress in Aerospace Sciences.* – 1997. – v. 33, No. 3–4. – pp. 259–284.
121. *Iliff K. W.* Parameter estimation for flight vehicles // *Journal of Guidance, Control and Dynamics.* – 1989. – v. 12, No. 5. – pp. 609–622.
122. *Jategaonkar R. V., Fischenberg D., von Gruenhagen W.* Aerodynamic modeling and system identification from flight data – Recent applications at DLR // *Journal of Aircraft.* – 2004. – v. 41, No. 4. – pp. 681–691.
123. *Klein V.* Estimation of aircraft aerodynamic parameters from flight data // *Progress in Aerospace Sciences.* – 1989. – v. 26, No. 1. – pp. 1–77.
124. *Morelli E. A., Klein V.* Application of system identification to aircraft at NASA Langley Research Center // *Journal of Aircraft.* – 2005. – v. 42, No. 1. – pp. 12–25.
125. *Wang K. C., Iliff K. W.* Retrospective and recent examples of aircraft parameter identification at NASA Dryden Flight Research Center // *Journal of Aircraft.* – 2004. – v. 41, No. 4. – pp. 752–764.
126. *Morelli E. A.* Multiple input design for real-time parameter estimation in the frequency domain // 13th IFAC Conf. on System Identification, Aug. 27–29, 2003, Rotterdam, The Netherlands. – Paper REG-360. – 7 pp.

127. *Morelli E. A.* In-flight system identification // *AIAA Paper*, No. 98-4261. – 10 pp.
128. *Morelli E. A., Klein V.* Real-time parameter estimation in the frequency domain // *J. of Guidance, Control, and Dynamics*. – 2000. – Vol. 23, No. 5. – pp. 812–818.
129. *Smith M. S., Moes T. R., Morelli E. A.* Flight investigation of prescribed simultaneous independent surface excitations for real-time parameter identification // *AIAA Paper*, No. 2003-5702. – 23 pp.
130. *Schroeder M. R.* Synthesis of low-peak-factor signals and binary sequences with low autocorrelation // *IEEE Trans. on Inform. Theory*. – Jan. 1970. – pp. 85–89.
131. *Васильев А. Н., Тархов Д. А.* Нейросетевое моделирование: Принципы. Алгоритмы. Приложения. – СПб.: Изд-во Политехнического ун-та, 2009. – 527 с.
132. *Васильев А. Н., Тархов Д. А.* Нейросетевые методы и алгоритмы математического моделирования. – СПб.: Изд-во Политехнического ун-та, 2014. – 582 с.
133. *Васильев А. Н., Тархов Д. А.* Принципы и техника нейросетевого моделирования. – СПб.: Нестор-История, 2014. – 218 с.
134. *Васильев А. Н., Тархов Д. А., Шемякина Т. А.* Нейросетевой подход к задачам математической физики. – СПб.: Нестор-История, 2015. – 260 с.
135. *Billings S. A.* Nonlinear system identification: NARMAX methods in the time, frequency and spatio-temporal domains. – John Wiley & Sons, 2013. – xvii+590 pp.
136. *Codrons B.* Process modelling for control: A unified framework using standard black-box techniques. – London: Springer, 2005. – xxxiii+229 pp.
137. *Narendra K. S., Parthasarathy K.* Identification and control of dynamic systems using neural networks // *IEEE Trans. on Neural Networks*. – 1990. – v. 1, No. 1. – pp. 4–27.
138. *Chen S., Billings S. A.* Neural networks for nonlinear dynamic systems modelling and identification. // *Int. J of Control*. – 1992. – v. 56, No. 2. – pp. 319–346.
139. *Conti M., Turchetti C.* Approximation of dynamical systems by continuous-time recurrent approximate identity neural networks. // *Neural, Parallel and Scientific Computations*. – 1994. – v. 2, No. 3. – pp. 299–320.
140. *Elanayar S., Shin Y. C.* Radial basis function neural network for approximation and estimation of nonlinear stochastic dynamic systems. // *IEEE Trans. on Neural Networks*. – 1994. – v. 5, No. 4. – pp. 594–603.

141. Pal C., Kayaba N., Morishita S., Hagiwara I. Dynamic system identification by neural network: A new fast learning method based on error back propagation. // *JSME Int. J. Series C: Dynamics, Control, Robotics, and Manufacturing*. – 1995. – v. 38, No. 4. – pp. 686–692.
142. Sjöberg J., Zhang Q., Ljung L., Benveniste A., Deylon B., Glorennec P.-Y., Hjalmarsson H., Juditsky A. Nonlinear black-box modeling in system identification: A unified overview // *Automatica*. – 1995. – v. 31, No. 12. – pp. 1691–1724.
143. Juditsky A., Hjalmarsson H., Benveniste A., Deylon B., Ljung L., Sjöberg J., Zhang Q. Nonlinear black-box modeling in system identification: Mathematical foundations // *Automatica*. – 1995. – v. 31, No. 12. – pp. 1725–1750.
144. Rivals I., Personnaz L. Black-box modeling with state-space neural networks // In: *Neural Adaptive Control Technology*. Eds. R. Zbikowski and K. J. Hint. – World Scientific. – 1996. – pp. 237–264.
145. Billings S. A., Jamaluddin H. B., Chen S. Properties of neural networks with applications to modelling nonlinear dynamic systems // *Int. J of Control*. – 1992. – v. 55, No. 1. – pp. 193–224.
146. Chen S., Billings S. A. Representation of non-linear systems: The NARMAX model // *Int. J of Control*. – 1989. – v. 49, No. 3. – pp. 1013–1032.
147. Chen S., Billings S. A. Nonlinear system identification using neural networks // *Int. J of Control*. – 1990. – v. 51, No. 6. – pp. 1191–1214.
148. Chen S., Billings S. A., Cowan C. F. N., Grant P. M. Practical identification of NARMAX using radial basis functions // *Int. J of Control*. – 1990. – v. 52, No. 6. – pp. 1327–1350.
149. Bohlin T. Interactive system identification: Prospects and Pitfalls. – Berlin ao.: Springer, 1991. – xii+365 pp.
150. Bohlin T. Practical grey-box identification: Theory and applications. – Berlin ao.: Springer, 2006. – xix+351 pp.
151. Ljung L. System identification toolbox: User's guide. – MathWorks, 2015. – 982 pp.
152. Abonyi J., Madar J., Szeifert F. Combining first principles models and neural networks for generic model control // In: *Soft Computing and Industry – Recent Applications*. Eds. R. Roy, M. Köppen ao. – London: Springer-Verlag. – 2002. – pp. 111–122.
153. Bohlin T. A case study of grey box identification // *Automatica*. – 1994. – v. 30, No. 2. – pp. 307–318.

154. Cen Z., Wei J., Jiang R. A gray-box neural network based model identification and fault estimation scheme for nonlinear dynamic systems // *International Journal of Neural Systems*. – 2013. – v. 23, No. 6. – pp. 1–15.
155. Frasconi P., Gori M., Maggini M., Soda G. Unified integration of explicit knowledge and learning by example in recurrent networks // *IEEE Transactions on Knowledge and Data Engineering*. – 1995. – v. 7, No. 2. – pp. 340–346.
156. Garatti S., Bittanti S. A new paradigm for parameter estimation in system modeling // *International Journal of Adaptive Control and Signal Processing*. – 2013. – v. 27, No. 8. – pp. 667–687.
157. Hasiewicz Z., Śliwiński P., Mzyk G. Nonlinear system identification under various prior knowledge // *Proceedings of the 17th World Congress of the International Federation of Automatic Control (IFAC)*, Seoul, Korea, July 6–11, 2008. – 10 pp.
158. Jorgensen S. B., Hangos K. M. Grey box modelling for control: Qualitative models as a unifying framework // *International Journal of Adaptive Control and Signal Processing*. – 1995. – v. 9, No. 6. – pp. 547–562.
159. Lindskog P., Ljung L. Tools for semiphysical modeling // *International Journal of Adaptive Control and Signal Processing*. – 1995. – v. 9, No. 6. – pp. 509–523.
160. Oussar Y., Dreyfus G. How to be a gray box: Dynamic semi-physical modeling // *Neural Networks*. – 2001. – v. 14, No. 9. – pp. 1161–1172.
161. Peres J., Oliveira R., Feyeo de Azevedo S. Knowledge based modular networks for process modelling and control // *Computers and Chemical Engineering*. – 2001. – v. 25, Nos. 4–6. – pp. 783–791.
162. Psychogios D. C., Ungar L. H. A hybrid neural network-first principles approach to process modeling // *AIChE Journal*. – 1992. – v. 38, No. 10. – pp. 1499–1511.
163. Yuan W., Katupitiya J. A grey-box system identification procedure for scale model helicopters based on frequency-domain estimation methods // *Proceedings of the 28th Congress of the International Council of the Aeronautical Sciences*, Brisbane, Australia, Sept. 23–28, 2012. ICAS Paper 2012–0837, 10 pp.
164. Dreyfus G., Idan Y. The canonical form of nonlinear discrete-time models // *Neural Computation*. – 1998. – v. 10. – pp. 133–164.
165. Nerrand O., Roussel-Ragot P., Personnaz L., Dreyfus G. Neural networks and non-linear adaptive filtering: Unifying concepts and new algorithms // *Neural Computation*. – 1993. – v. 5, No. ??? . – pp. 165–197.

166. *Nerrand O., Roussel-Ragot P., Urbani D., Personnaz L., Dreyfus G.* Training recurrent neural networks: Why and how? An illustration in dynamical process modeling // *IEEE Trans. on Neural Networks.* – 1994. – v. 5, No. 2. – pp. 178–184.
167. *Хайпер Э., Нёрсетт С., Ваннер Г.* Решение обыкновенных дифференциальных уравнений: Нежесткие задачи: Пер. с англ. – М.: Мир, 1990. – 512 с.
168. *Хайпер Э., Ваннер Г.* Решение обыкновенных дифференциальных уравнений: Жесткие и дифференциально-алгебраические задачи: Пер. с англ. – М.: Мир, 1999. – 685 с.
169. *Scott L. R.* Numerical analysis. – Princeton University Press, 2011. – xiv+325 pp.
170. *Брусов В. С., Тюменцев Ю. В.* Нейросетевое моделирование движения летательных аппаратов. – М.: Изд-во МАИ, 2016. – 192 с.
171. *Тюменцев Ю. В., Чернышев А. В.* Искусственные нейронные сети: основные понятия. Учебное пособие. – М.: Изд-во МАИ, 2012. – 48 с.
172. *Тюменцев Ю. В., Чернышев А. В.* Обучение многослойных искусственных нейронных сетей прямого распространения. Учебное пособие. – М.: Изд-во МАИ, 2012. – 48 с.
173. *Морозов Н. И., Тюменцев Ю. В., Яковенко А. В.* Корректировка динамических свойств объекта управления с использованием искусственных нейронных сетей // *Вестник Московского авиационного института.* – 2002, том 9, № 1. – с. 73–94. ISSN 0869-6101.
174. *Тюменцев Ю. В.* Интеллектуальные автономные системы // *Авиакосмическое приборостроение.* – 2004, № 10. – с. 2–14. ISSN 2073-0020.
175. *Тюменцев Ю. В.* Интеллектуальное управление движением летательных аппаратов с использованием средств мягких и полумягких вычислений // *Вестник компьютерных и информационных технологий.* – 2006, № 4. – с. 9–20. ISSN 1810-7206.
176. *Морозов Н. И., Тюменцев Ю. В.* Возможности формирования интеллектуальных законов управления на основе средств мягких и полумягких вычислений // *Вестник компьютерных и информационных технологий.* – 2006, № 5. – с. 8–21. ISSN 1810-7206.
177. *Брусов В. С., Тюменцев Ю. В.* Синтез оптимального ансамбля нейроконтроллеров для многорежимного летательного аппарата // *Вестник Московского авиационного института.* – 2006. – том 13, № 2. – с. 67–78. ISSN 0869-6101.

178. *Ефремов А. В., Оглоблин А. В., Тань В., Тюменцев Ю. В.* Нейросетевая модель управляющих действий летчика // *Вестник Московского авиационного института.* – 2007. – том 14, № 2. – с. 53–66. ISSN 0869-6101.
179. *Ефремов А. В., Тань В., Тюменцев Ю. В.* Оценка пилотажных свойств самолета с применением нейросетевой модели предсказания характеристик управляющих действий летчика // *Вестник Московского авиационного института.* – 2008. – том 15, № 1. – с. 92–108. ISSN 0869-6101.
180. *Ефремов А. В., Тань В., Тюменцев Ю. В.* Построение и применение нейросетевой модели характеристик управляющих действий летчика для оценки и выбора пилотажных свойств самолета // *Полет.* – 2008. – № 6. – с. 8–16. ISSN 1684-1301.
181. *Кондратьев А. И., Тюменцев Ю. В.* Нейросетевое моделирование управляемого движения летательных аппаратов // *Вестник Московского авиационного института.* – 2010, том 17, № 4. – с. 5–11. ISSN 0869-6101.
182. *Козлов Д. С., Тюменцев Ю. В.* Применение неоднородных NARX-сетей для обнаружения отказных ситуаций // *Нейрокомпьютеры: разработка, применение.* – 2012. № 9. – с. 13–22. ISSN 1999-8554.
183. *Козлов Д. С., Тюменцев Ю. В.* Нейросетевые методы обнаружения отказов датчиков и приводов летательного аппарата // *Электронный журнал «Труды МАИ».* – 2012, № 52. – 16 с. ISSN 1727-6942.
184. *Кондратьев А. И., Тюменцев Ю. В.* Применение нейронных сетей для синтеза алгоритмов управления полетом – 1. Нейросетевой метод обратной динамики для управления полетом самолета // *Известия ВУЗов. Авиационная техника.* – 2013, № 2. – с. 23–30. ISSN 0579-2975.
185. *Кондратьев А. И., Тюменцев Ю. В.* Применение нейронных сетей для синтеза алгоритмов управления полетом – 2. Адаптивная настройка нейросетевого закона управления // *Известия ВУЗов. Авиационная техника.* – 2013, № 3. – с. 34–39. ISSN 0579-2975.
186. *Егорчев М. В., Козлов Д. С., Тюменцев Ю. В., Чернышев А. В.* Нейросетевые полуэмпирические модели управляемых динамических систем // *Вестник информационных и компьютерных технологий.* – 2013, № 9. – с. 3–10. ISSN 1810-7206.
187. *Простов Ю. С., Тюменцев Ю. В.* Нейросетевой механизм адаптации при решении кусочно-постоянной задачи анализа независимых компонент // *Электронный журнал «Труды МАИ».* – 2014, № 78. – 20 с. ISSN 1727-6942.

188. *Егорчев М. В., Козлов Д. С., Тюменцев Ю. В.* Идентификация аэродинамических характеристики летательного аппарата: нейросетевой полуэмпирический подход // *Вестник Московского авиационного института*. – 2014. – том 21, № 4. – с. 13–24. ISSN 0869-6101.
189. *Егорчев М. В., Козлов Д. С., Тюменцев Ю. В.* Моделирование продольного углового движения самолета: сопоставление теоретического, эмпирического и полуэмпирического подходов // *Научный вестник МГТУ ГА*. – 2015, № 211 (1). – с. 116–123. ISSN 2079-0619.
190. *Егорчев М. В., Козлов Д. С., Тюменцев Ю. В.* Нейросетевая полуэмпирическая модель продольного короткопериодического движения маневренного самолета // *Полет*. – 2015, № 1. – с. 53–60. ISSN 1684-1301.
191. *Prostov Yu. S., Tiumentsev Yu. V.* A hysteresis micro ensemble as a basic element of an adaptive neural net // *Optical Memory and Neural Networks (Information Optics)*. – 2015, Vol. 24, No. 2. – pp. 116–122. ISSN 1060-992X.
192. *Ефремов А. В., Кошеленко А. В., Тяглик М. С., Тюменцев Ю. В., Тянь Вэнь Цзянь.* Математическое моделирование характеристик управляющих действий летчика при исследовании задач ручного управления // *Известия ВУЗов. Авиационная техника*. – 2015, № 2. – с. 173–179. ISSN 0579-2975.
193. *Egorchev M. V., Tiumentsev Yu. V.* Learning of semi-empirical neural network model of aircraft three-axis rotational motion // *Optical Memory and Neural Networks (Information Optics)*. – 2015, Vol. 24, No. 3. – pp. 210–217. ISSN 1060-992X.
194. *Простов Ю. С., Тюменцев Ю. В.* Нейросетевая модель конечного автомата на основе гистерезисных микроансамблей // *Вестник Национального исследовательского ядерного университета «МИФИ»*. – 2015, том 4, № 5. – с. 447–453. ISSN 0579-2975.
195. *Kozlov D. S., Tiumentsev Yu. V.* Learning of semi-empirical neural network model of aircraft three-axis rotational motion // *Optical Memory and Neural Networks (Information Optics)*. – 2015, Vol. 24, No. 4. – pp. 279–287. ISSN 1060-992X.
196. *Брусов В. С., Тюменцев Ю. В.* Синтез оптимального ансамбля нейроконтроллеров для многорежимного летательного аппарата // *Всероссийская научно-техническая конференция «Нейроинформатика-99»*, Сб. науч. трудов. Часть 2. ISBN 5-7262-0268-6. – М.: Изд-во МИФИ, 1999. – с. 190–201.
197. *Тюменцев Ю. В.* Интеллектуальные автономные системы: Вызов информационным технологиям // *Восьмая Национальная конференция по искусственному интеллекту (КИИ'2002)*, Коломна, 7–12 октября 2002 г. Труды конф., том 2. ISBN 5-94052-056-1. – М.: Физматлит, 2002. – с. 827–836.

198. *Кондратьев А. И., Тюменцев Ю. В.* Нейросетевое адаптивное отказоустойчивое управление движением маневренного самолета // Всероссийская научно-техническая конференция «Нейроинформатика-2010», Сб. науч. трудов. Часть 2. – ISBN 978-5-7262-1226-5. – М.: Изд-во МИФИ, 2010. – с. 262–272.
199. *Brusov V.S., Petruchik V.P., Tiumentsev Yu. V.* Theoretical and experimental investigations of aerodynamics and flight dynamics for micro-UAV // 27th International Congress of the Aeronautical Sciences (ICAS 2010), 19–24 September 2010, Nice, France, CD-ROM Proceedings, ISBN 978-0-9565333-0-2. – 9 pp.
200. *Козлов Д. С., Тюменцев Ю. В.* Нейросетевые методы выявления отказных ситуаций // Сборник научных трудов XIV Всероссийской научно-технической конференции «Нейроинформатика-2012», Часть 2, Москва, МИФИ, 23–27 января 2012 г. – ISBN 978-5-7262-1380-4. – М.: Изд-во МИФИ, 2012. – с. 257–266.
201. *Тюменцев Ю. В.* Нейросетевая корректировка динамических свойств летательного аппарата // В кн.: «Нейрокомпьютеры в интеллектуальных технологиях XXI века» под ред. *Ю. И. Нечаева*, ISBN 978-5-88070-310-4, М.: Радиотехника, 2012 г. – с. 102–113.
202. *Кондратьев А. И., Тюменцев Ю. В.* Адаптивное отказоустойчивое управление движением маневренного самолета и его нейросетевая реализация // В кн.: «Нейрокомпьютеры в интеллектуальных технологиях XXI века» / Под ред. *Ю. И. Нечаева*, ISBN 978-5-88070-310-4, М.: Радиотехника, 2012 г. – с. 192–201.
203. *Кондратьев А. И., Тюменцев Ю. В.* Адаптивный нейросетевой закон управления пространственным движением самолета // Сб. трудов конф.: «Информационные технологии в управлении» (ИТУ-2012). – Санкт-Петербург, 9–11 октября 2012 г., ISBN 978-5-900780-94-8. – СПб.: ОАО Концерн ЦНИИ «Электроприбор», 2012. – с. 795–805.
204. *Егорчев М. В., Тюменцев Ю. В.* Нейросетевые полуэмпирические модели управляемых динамических систем // Сборник научных трудов XV Всероссийской научно-технической конференции «Нейроинформатика-2013», Часть 2, Москва, МИФИ, 21–25 января 2013 г., ISBN 978-5-7262-1782-6. – М.: Изд-во МИФИ, 2013. – с. 22–31.
205. *Кондратьев А. И., Тюменцев Ю. В.* Применение нейронных сетей в контуре управления и синтез автопилота // Сборник научных трудов XV Всероссийской научно-технической конференции «Нейроинформатика-2013», Часть 2, Москва, МИФИ, 21–25 января 2013 г., ISBN 978-5-7262-1782-6. – М.: Изд-во МИФИ, 2013. – с. 50–59.

206. *Егорчев М. В., Тюменцев Ю. В.* Обучение полуэмпирической нейросетевой модели управляемого движения самолета // Сборник научных трудов XVI Всероссийской научно-технической конференции «Нейроинформатика-2014», Часть 2, Москва, МИФИ, 27–31 января 2014 г., ISBN 978-5-7262-1782-6. – М.: Изд-во МИФИ, 2014. – с. 263–272.
207. *Efremov A. V., Popov S. A., Nikitchenko Yu. A., Koshelenko A. V., Tiumentsev Yu. V.* Aerodynamics and flight dynamics problems of high speed vehicles and ways for their solution // Proceedings of the 29th Congress of the International Council of the Aeronautical Sciences, St. Petersburg, Russia, Sept. 7–12, 2014. ICAS Paper 2014–0487, 8 pp.
208. *Egorchev M. V., Kozlov D. S., Tiumentsev Yu. V.* Neural network adaptive semi-empirical models for aircraft controlled motion // Proceedings of the 29th Congress of the International Council of the Aeronautical Sciences, St. Petersburg, Russia, Sept. 7–12, 2014. ICAS Paper 2014–0530, 8 pp.
209. *Простов Ю. С., Тюменцев Ю. В.* Исследование нейросетевой модели на базе гистерезисных микроансамблей // Сборник научных трудов XVII Всероссийской научно-технической конференции «Нейроинформатика-2015», Часть 1, Москва, МИФИ, 19–23 января 2015 г., ISBN 978-5-7262-2043-7. – М.: Изд-во МИФИ, 2015. – с. 116–126.
210. *Козлов Д. С., Тюменцев Ю. В.* Нейросетевые полуэмпирические модели динамических систем, описываемых дифференциально-алгебраическими уравнениями // Сборник научных трудов XVII Всероссийской научно-технической конференции «Нейроинформатика-2015», Часть 2, Москва, МИФИ, 19–23 января 2015 г., ISBN 978-5-7262-2044-4. – М.: Изд-во МИФИ, 2015. – с. 10–20.
211. *Козлов Д. С., Тюменцев Ю. В.* Нейросетевые полуэмпирические модели динамических систем, представляемых в форме дифференциально-алгебраических уравнений индекса 1 // Сборник научных трудов XVIII Международной научно-технической конференции «Нейроинформатика-2016», Часть 3, Москва, МИФИ, 25–29 апреля 2016 г., ISBN 978–5–7262–2240–0. – М.: НИЯУ МИФИ, 2016. – с. 61–71.

Приложения

А Результаты вычислительных экспериментов с адаптивными системами

А.1. Адаптивное управление с эталонной моделью (ЭМ)

А.1.1. Компенсирующий контур в схеме адаптивного управления с ЭМ

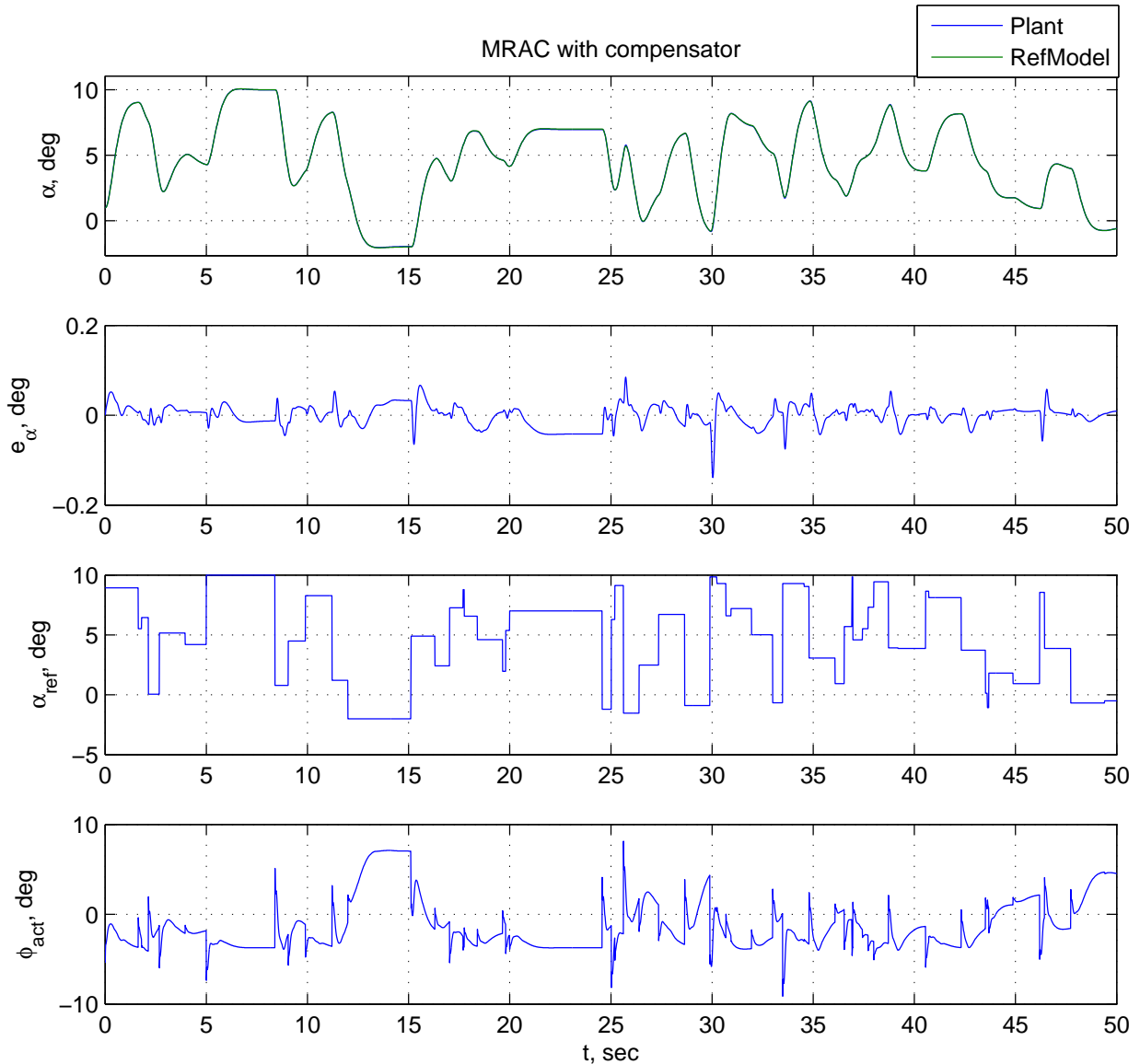


Рис. А.1. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором (ГЗЛА Х-43, режим полета $M = 6$). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

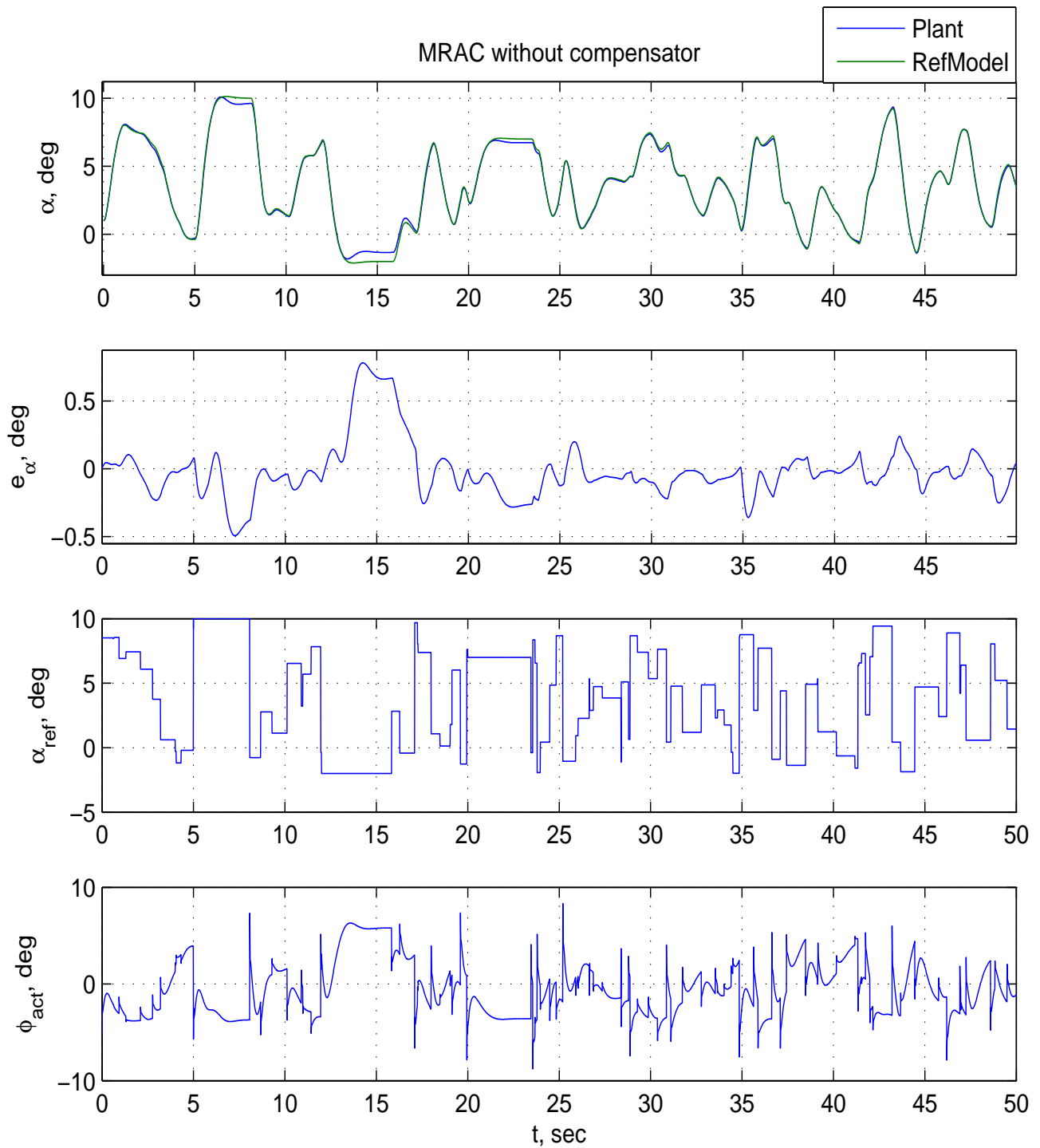


Рис. А.2. Результаты вычислительного эксперимента для системы управления с эталонной моделью и без компенсатора (ГЗЛА Х-43, режим полета $M = 6$). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

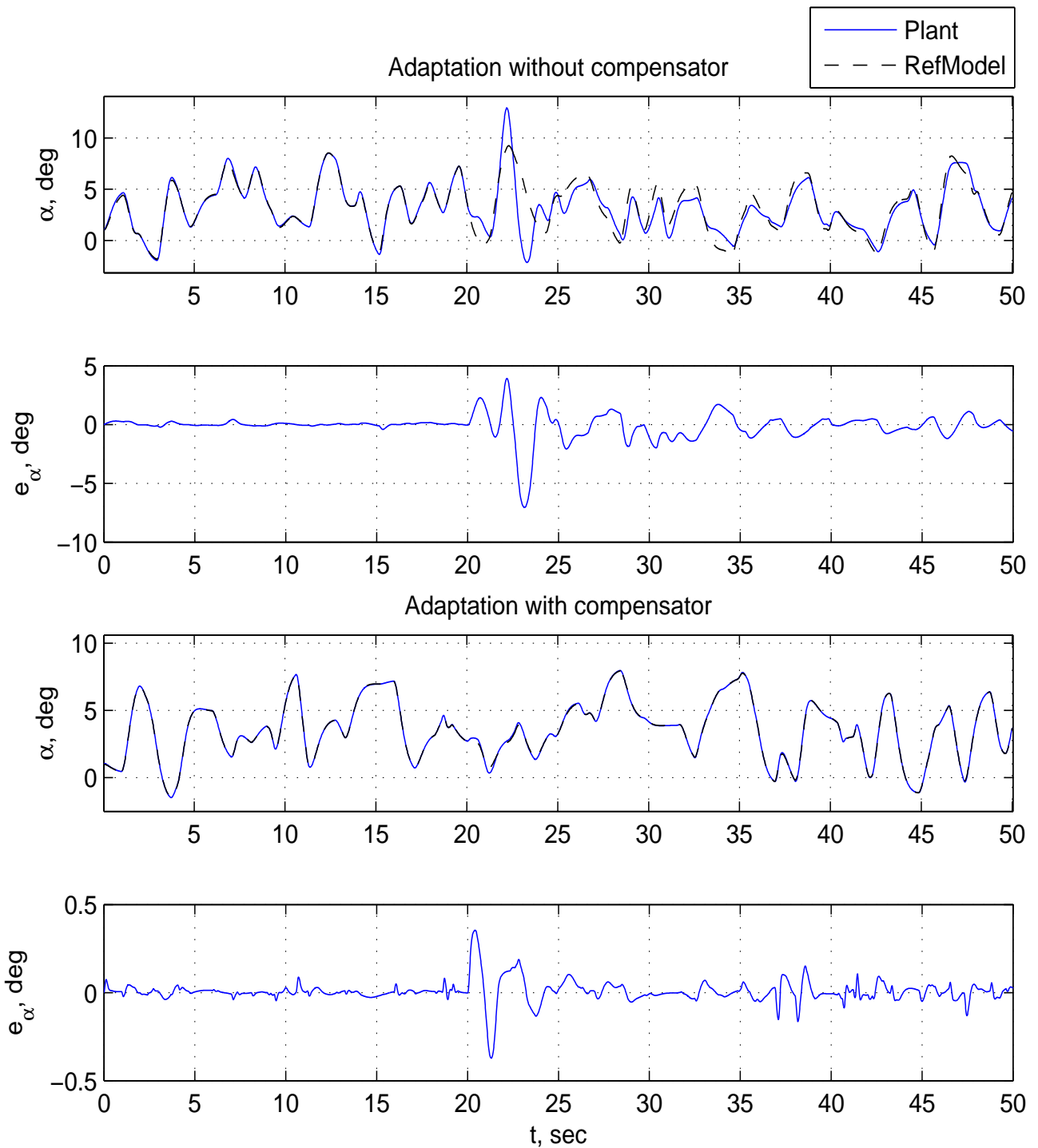


Рис. А.3. Результаты вычислительного эксперимента для системы управления с ЭМ, без компенсатора (верхняя пара графиков) и с компенсатором (нижняя пара графиков) для ГЗЛА Х-43, режим полета $M = 6$ при возникновении отказной ситуации в момент времени $t = 20$ с (смещение центровки на 5% назад). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

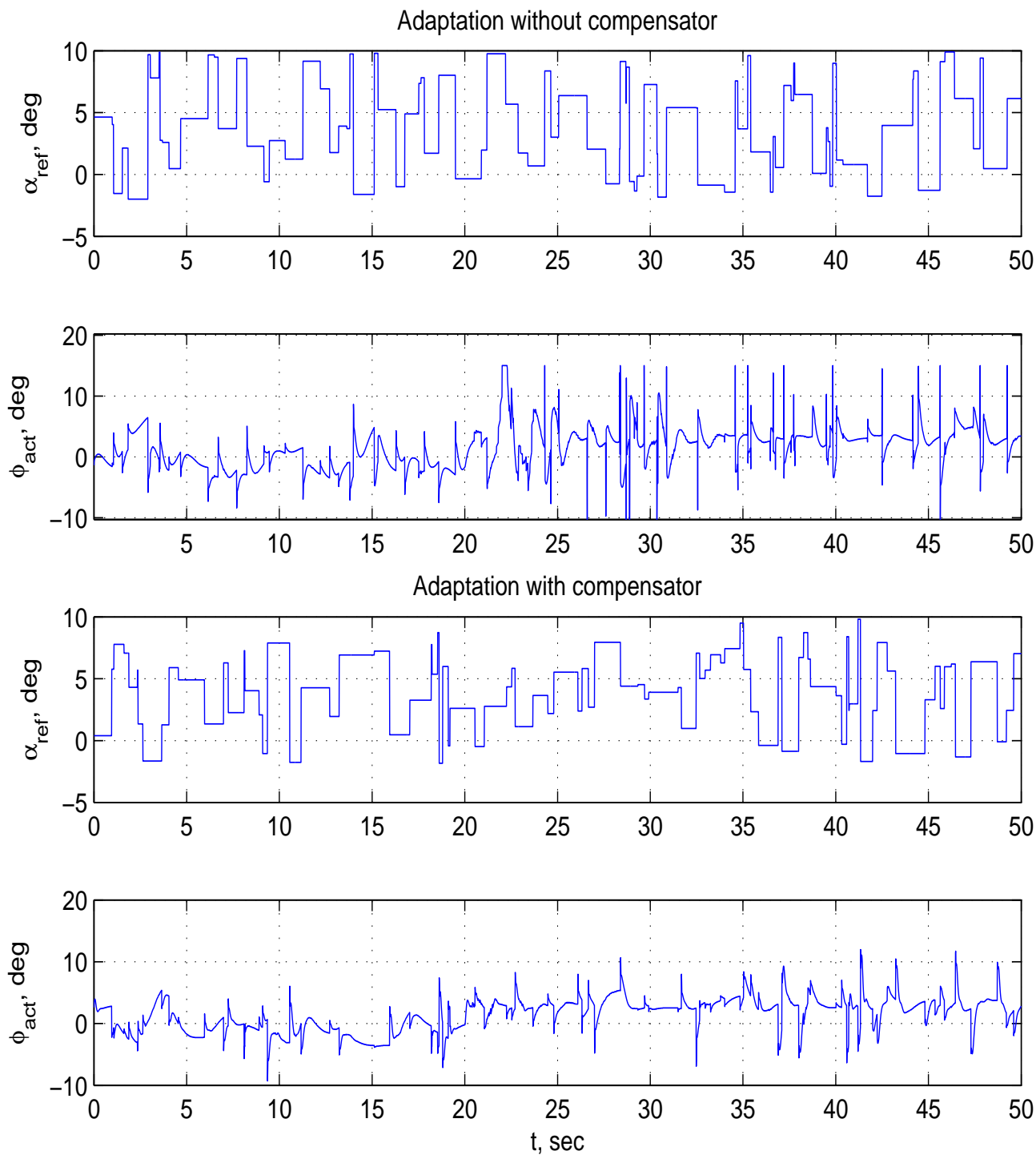


Рис. А.4. Результаты вычислительного эксперимента для системы управления с ЭМ, без компенсатора (верхняя пара графиков) и с компенсатором (нижняя пара графиков) для ГЗЛА Х-43, режим полета $M = 6$ при возникновении отказной ситуации в момент времени $t = 20$ с (смещение центровки на 5% назад). Обозначения: α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

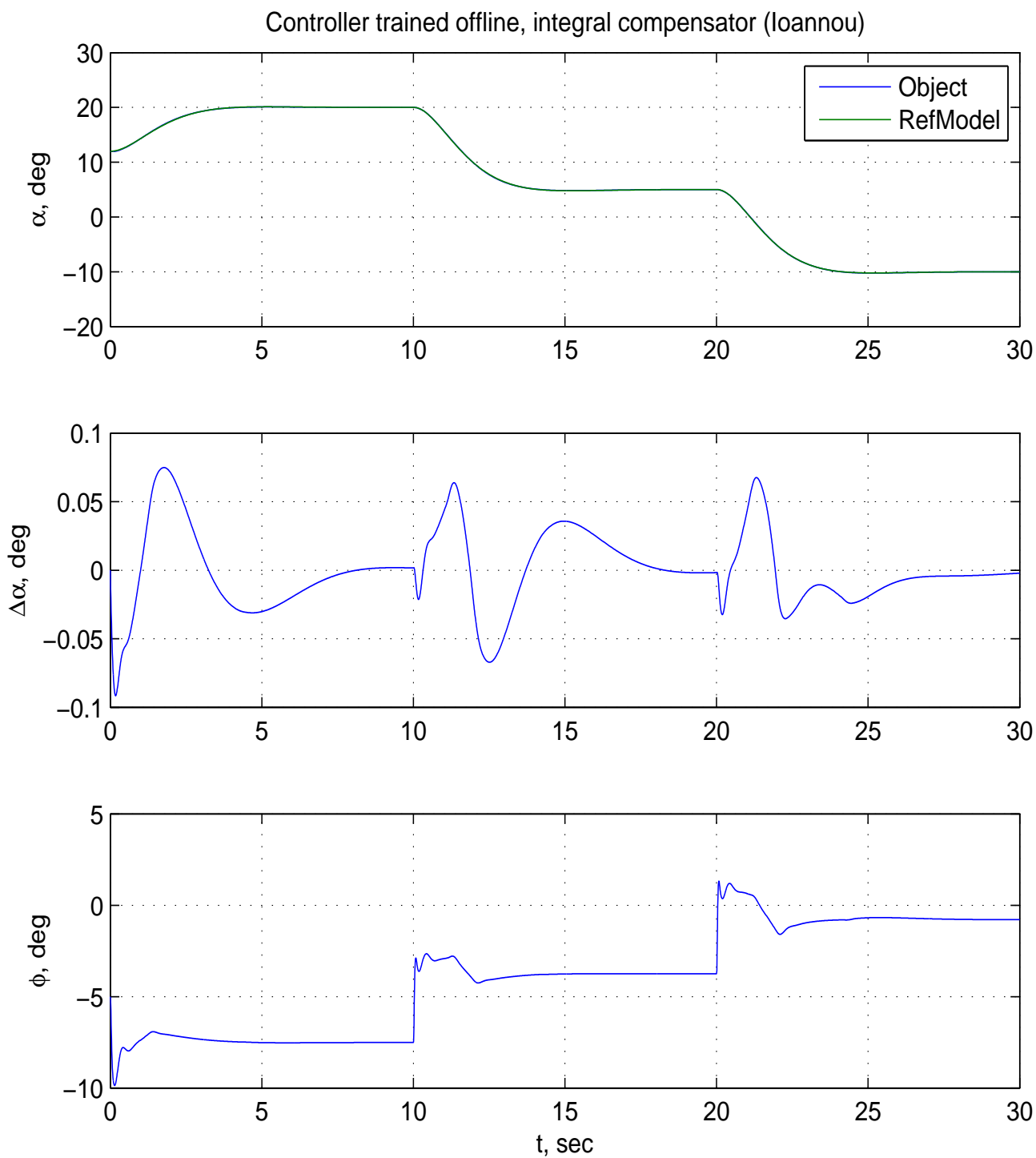


Рис. А.5. Результаты вычислительного эксперимента для системы управления с ЭМ и интегральным компенсатором для самолета F-16, режим полета с индикаторной скоростью $V_{ind} = 300$ км/ч . Обозначения: α — угол атаки, град.; $\Delta\alpha$ — ошибка отслеживания заданного угла атаки, град.; ϕ — командный сигнал для привода цельноповоротного стабилизатора, град.; t — время, с; Object — объект управления; RefModel — эталонная модель

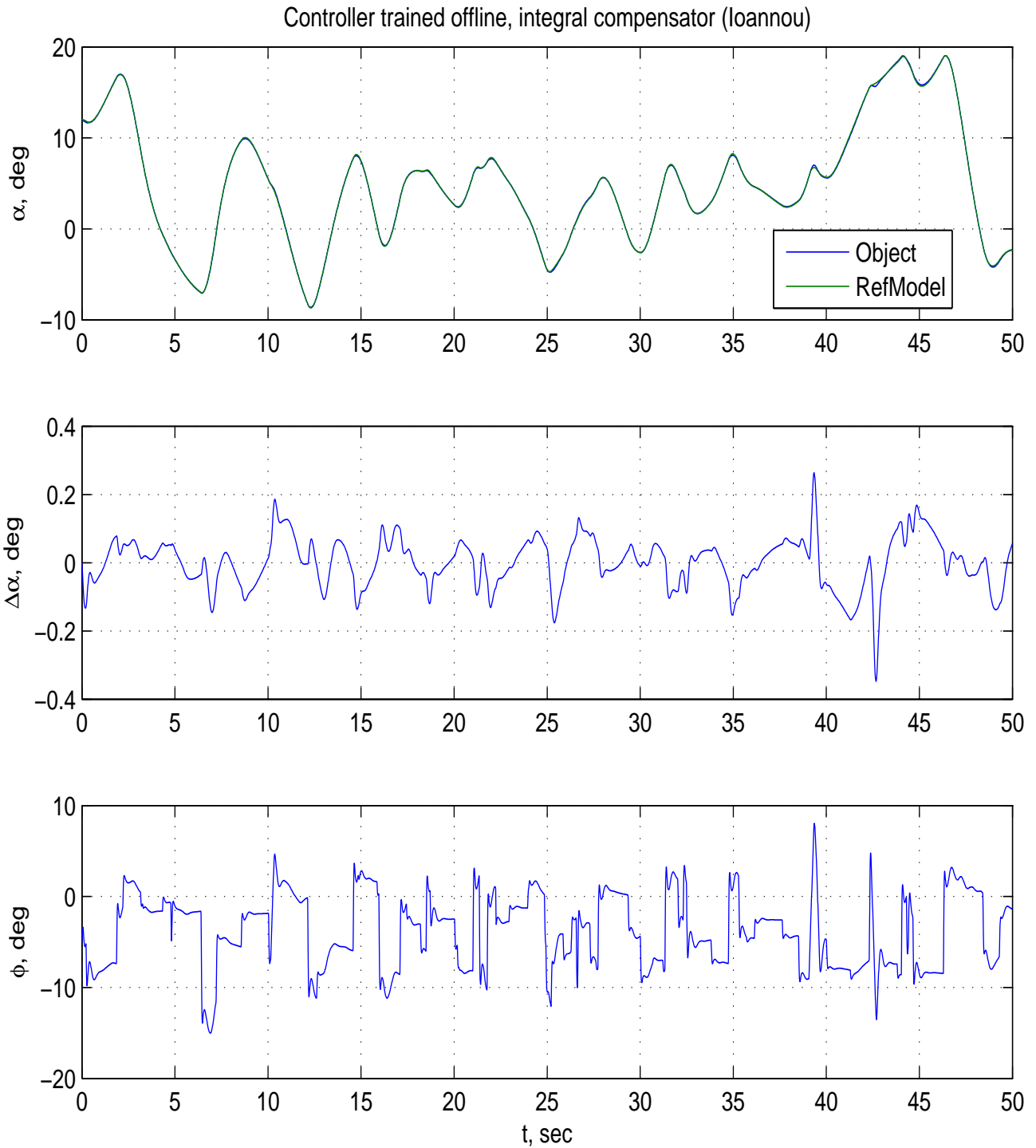


Рис. А.6. Результаты вычислительного эксперимента для системы управления с ЭМ и интегральным компенсатором для самолета F-16, режим полета с индикаторной скоростью $V_{ind} = 300$ км/ч . Обозначения: α — угол атаки, град.; $\Delta\alpha$ — ошибка отслеживания заданного угла атаки, град.; ϕ — командный сигнал для привода цельноповоротного стабилизатора, град.; t — время, с; Object — объект управления; RefModel — эталонная модель

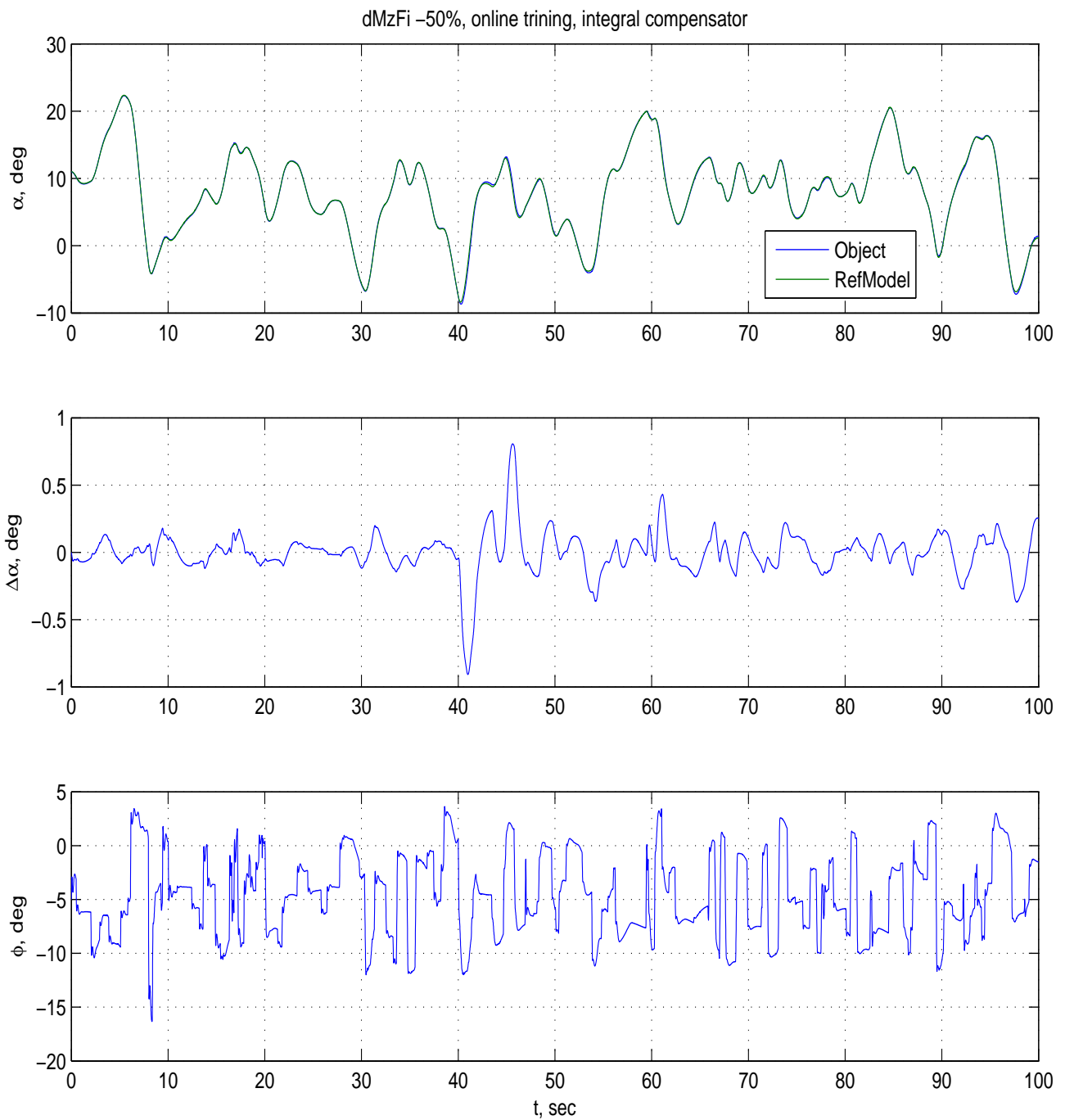


Рис. А.7. Результаты вычислительного эксперимента для системы управления с ЭМ и интегральным компенсатором для самолета F-16, режим полета с индикаторной скоростью $V_{ind} = 300$ км/ч при возникновении отказной ситуации в момент времени $t = 40$ с (уменьшение на 50% эффективности органа продольного управления). Обозначения: α — угол атаки, град.; $\Delta\alpha$ — ошибка отслеживания заданного угла атаки, град.; ϕ — командный сигнал для привода цельноповоротного стабилизатора, град.; t — время, с; Object — объект управления; RefModel — эталонная модель

А.1.2. Оценка влияния параметров эталонной модели на работоспособность синтезируемой системы управления

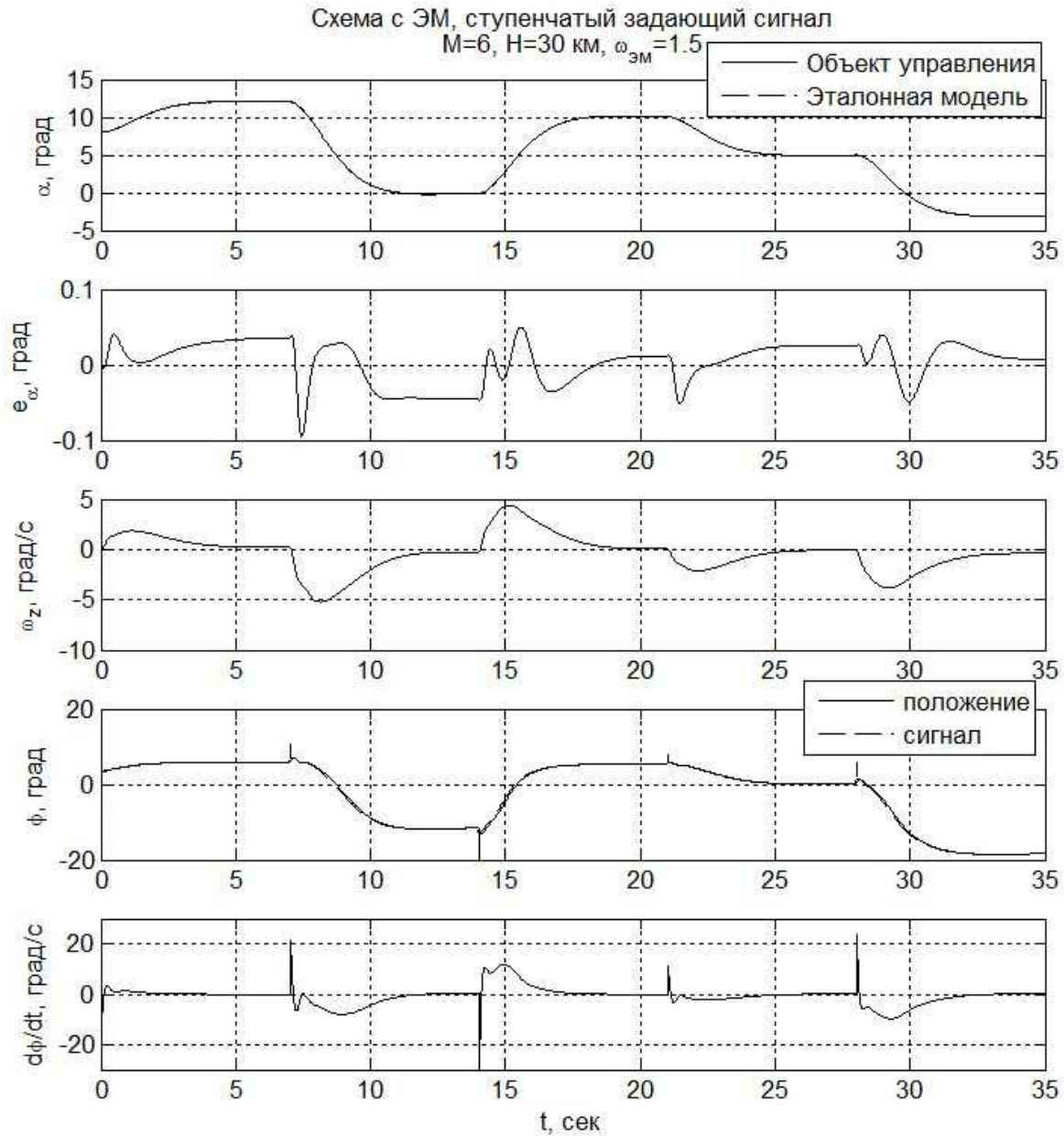


Рис. А.8. Результаты вычислительного эксперимента для системы управления с эталонной моделью применительно к перспективному ГЗЛА для оценки влияния собственной частоты $\omega_{\text{ЭМ}}$ эталонной модели ($\omega_{\text{ЭМ}} = 1.5$, ступенчатый задающий сигнал по углу атаки, режим полета $M = 6, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

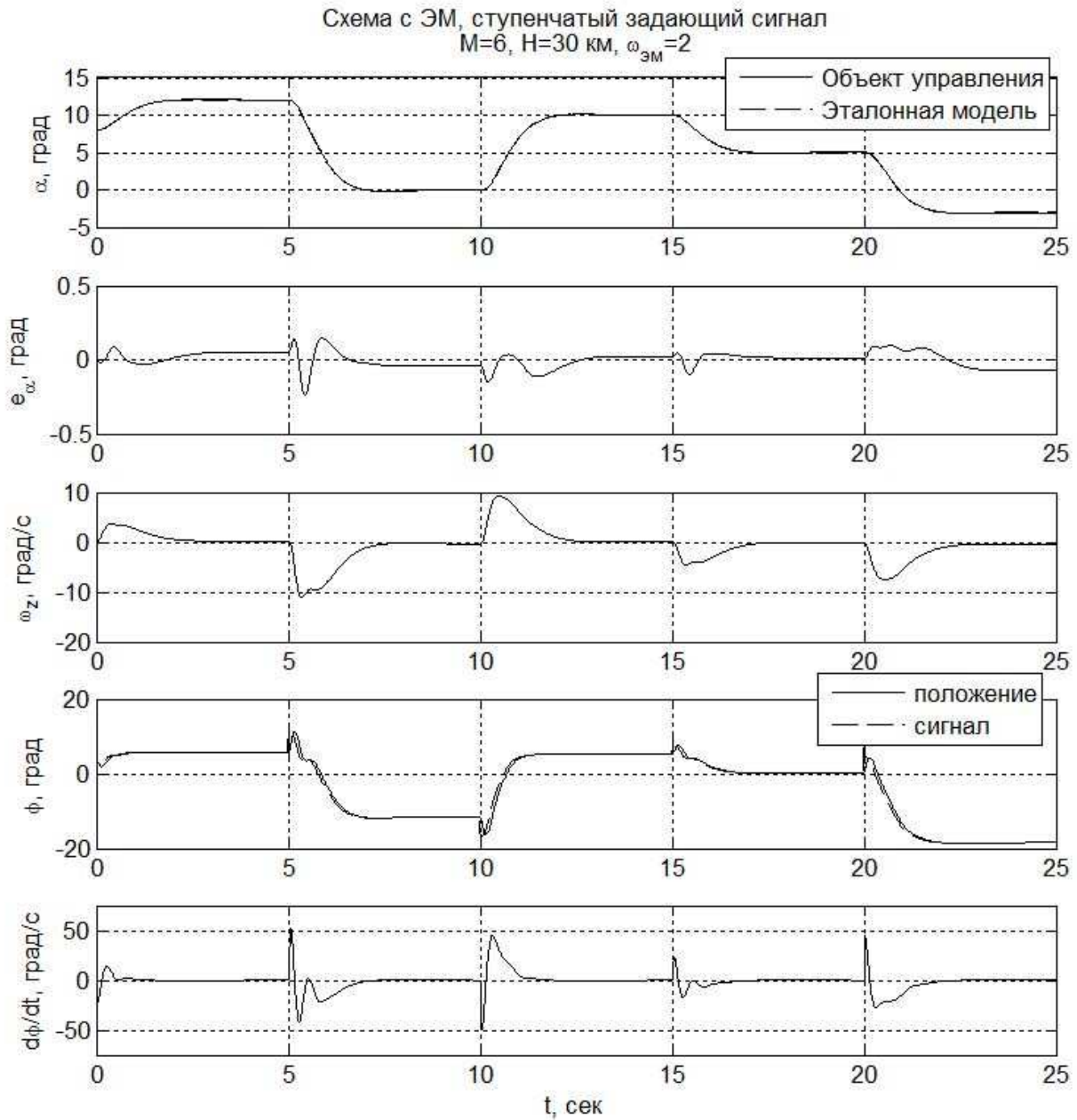


Рис. А.9. Результаты вычислительного эксперимента для системы управления с эталонной моделью применительно к перспективному ГЗЛА для оценки влияния собственной частоты $\omega_{\text{ЭМ}}$ эталонной модели ($\omega_{\text{ЭМ}} = 2$, ступенчатый задающий сигнал по углу атаки, режим полета $M = 6, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

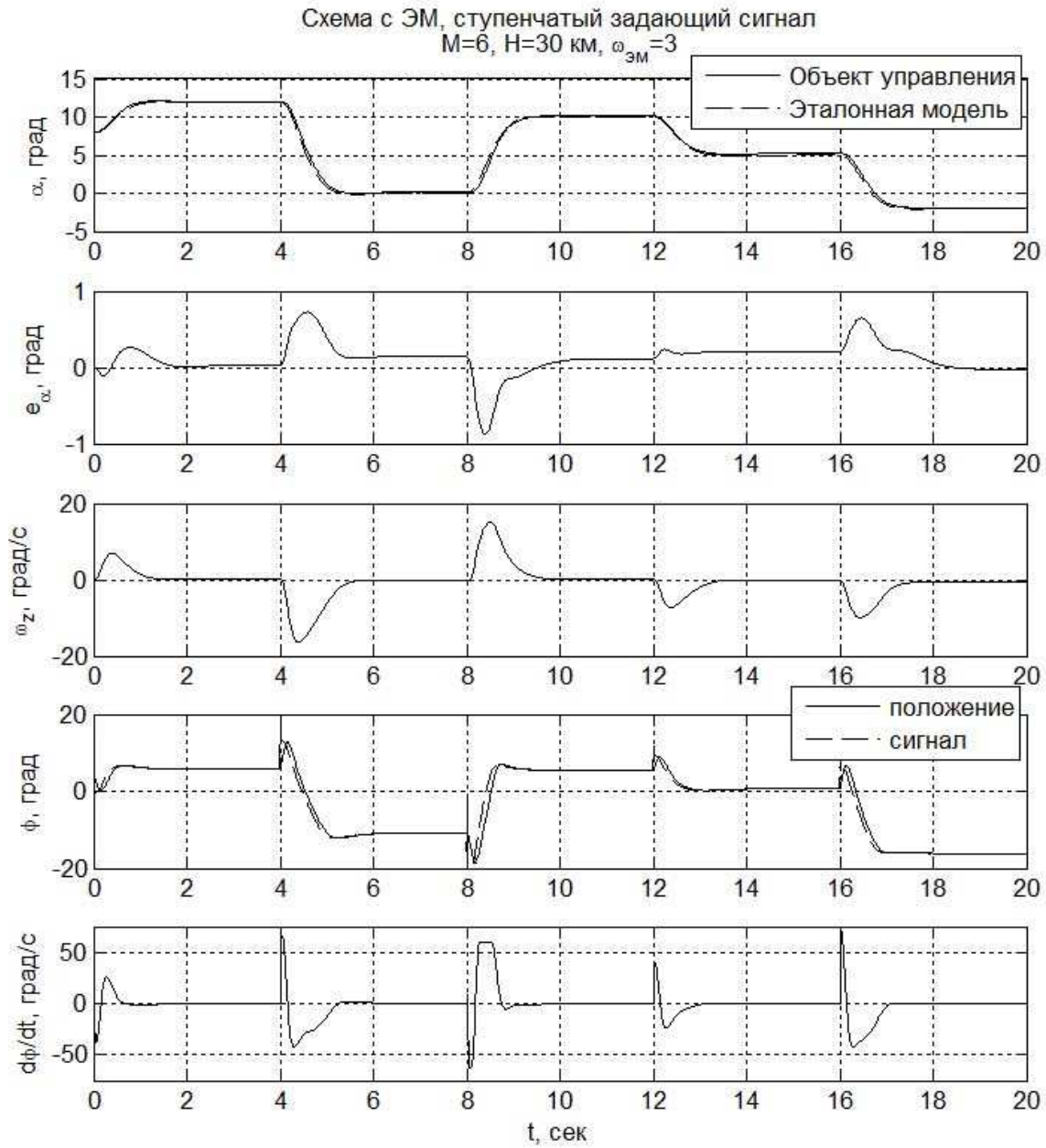


Рис. А.10. Результаты вычислительного эксперимента для системы управления с эталонной моделью применительно к перспективному ГЗЛА для оценки влияния собственной частоты $\omega_{\text{ЭМ}}$ эталонной модели ($\omega_{\text{ЭМ}} = 3$, ступенчатый задающий сигнал по углу атаки, режим полета $M = 6, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

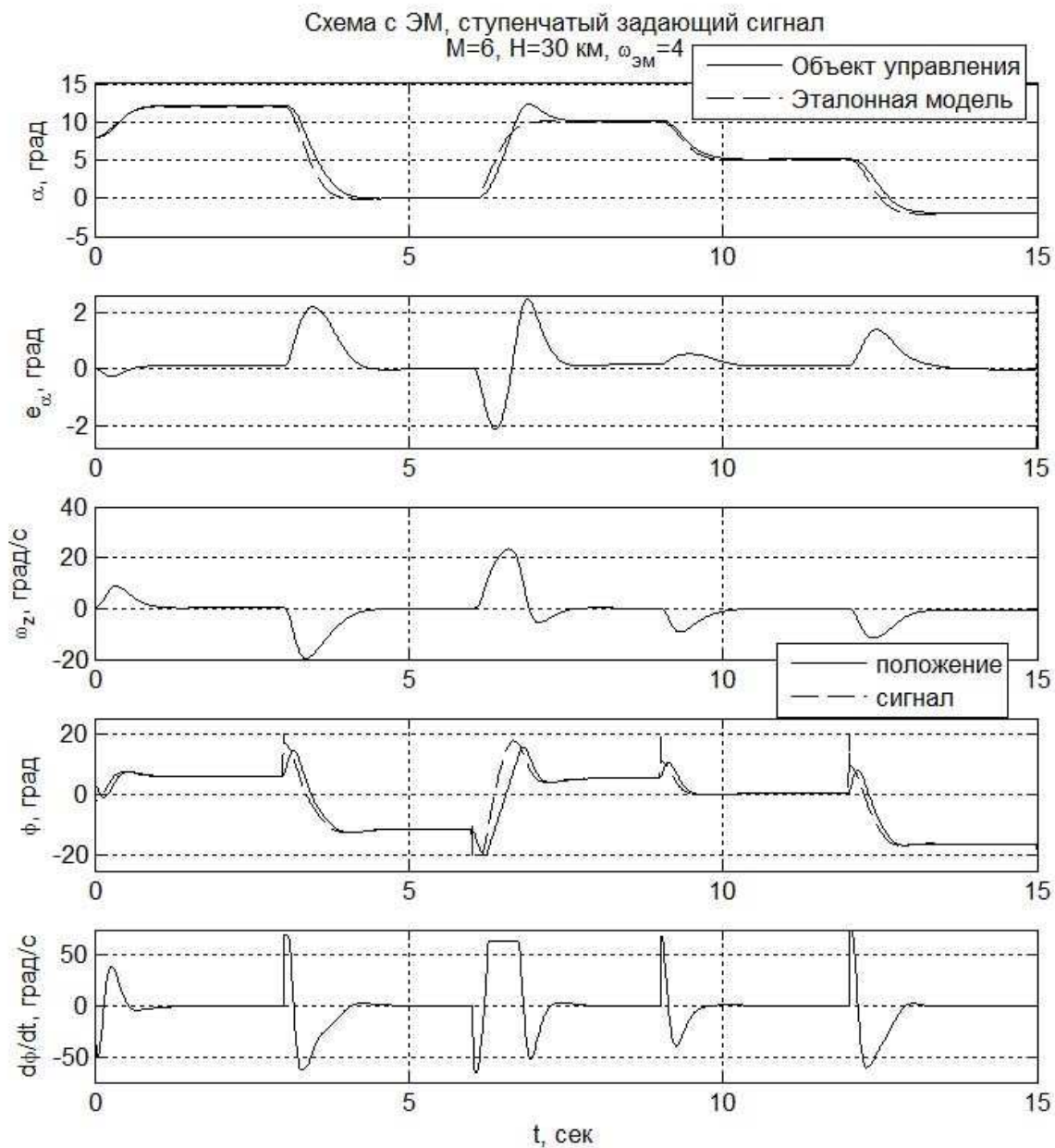


Рис. А.11. Результаты вычислительного эксперимента для системы управления с эталонной моделью применительно к перспективному ГЗЛА для оценки влияния собственной частоты $\omega_{\text{ЭМ}}$ эталонной модели ($\omega_{\text{ЭМ}} = 4$, ступенчатый задающий сигнал по углу атаки, режим полета $M = 6, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

А.1.3. Результаты вычислительного эксперимента для схемы адаптивного управления с эталонной моделью применительно к управлению угловым движением ЛА

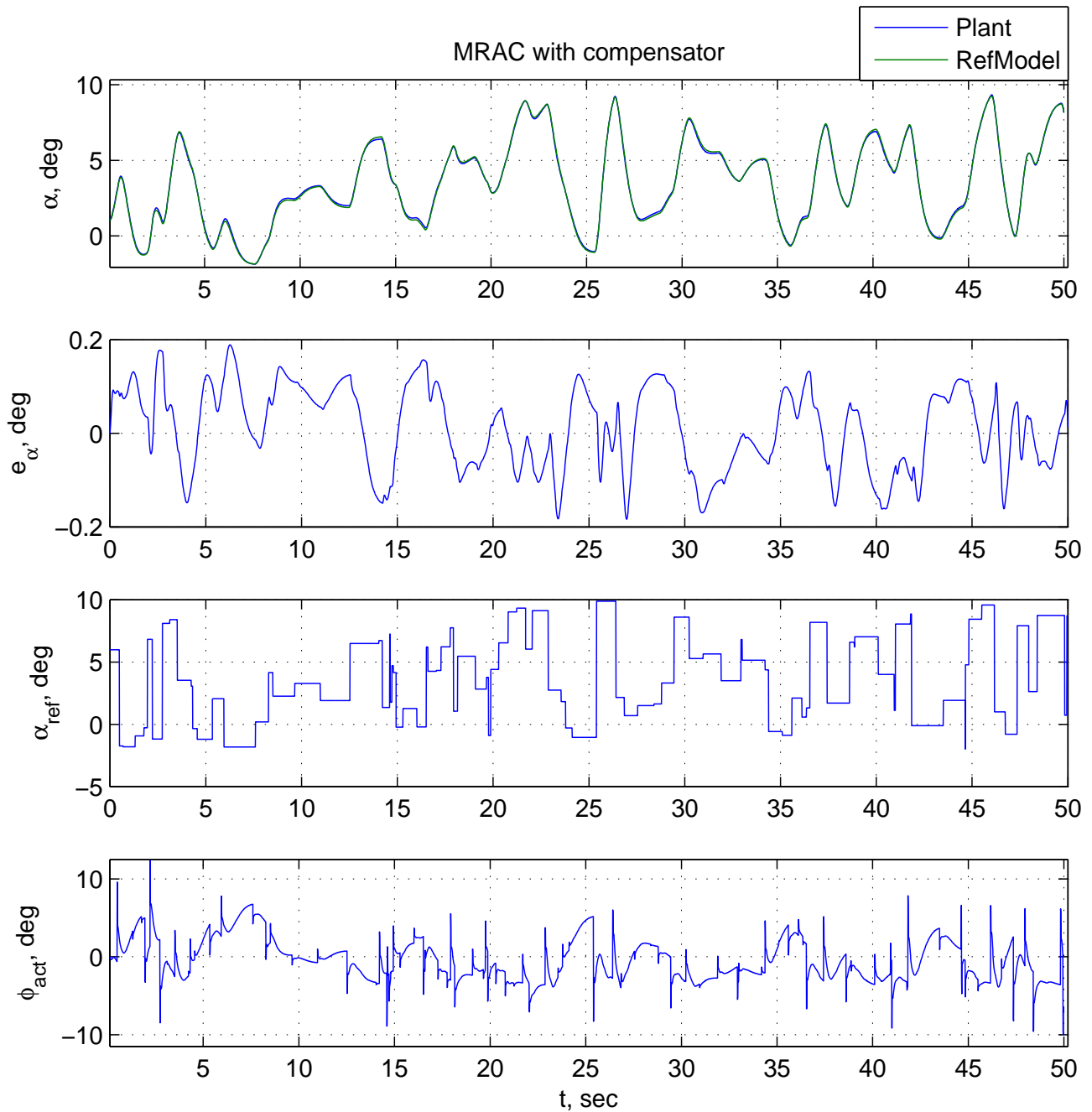


Рис. А.12. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором при недостаточно точной НС-модели объекта управления (ГЗЛА Х-43, режим полета $M = 6$). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

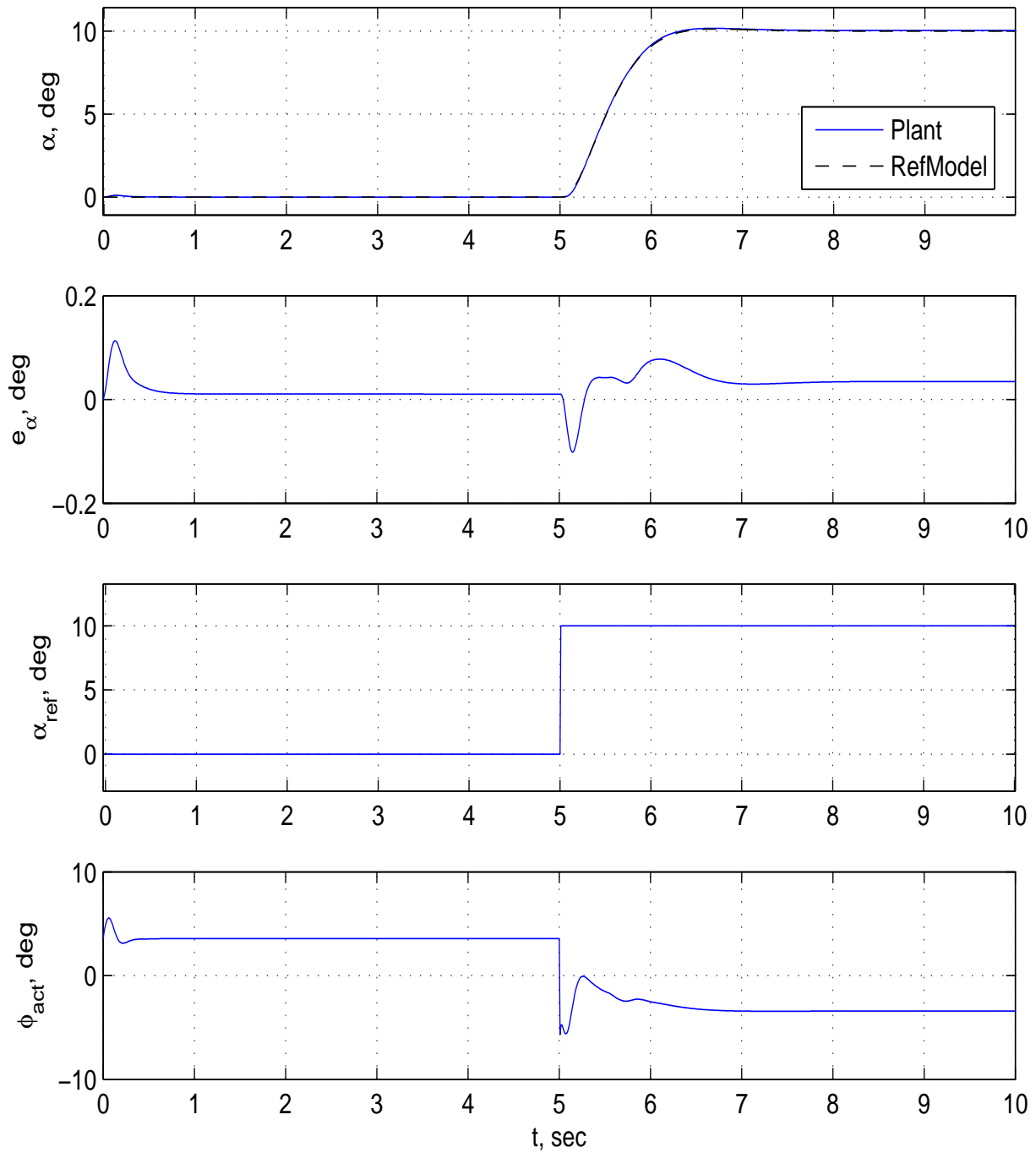


Рис. А.13. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором при ступенчатом изменении задающего сигнала (ГЗЛА Х-43, режим полета $M = 6$, вариант 1). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

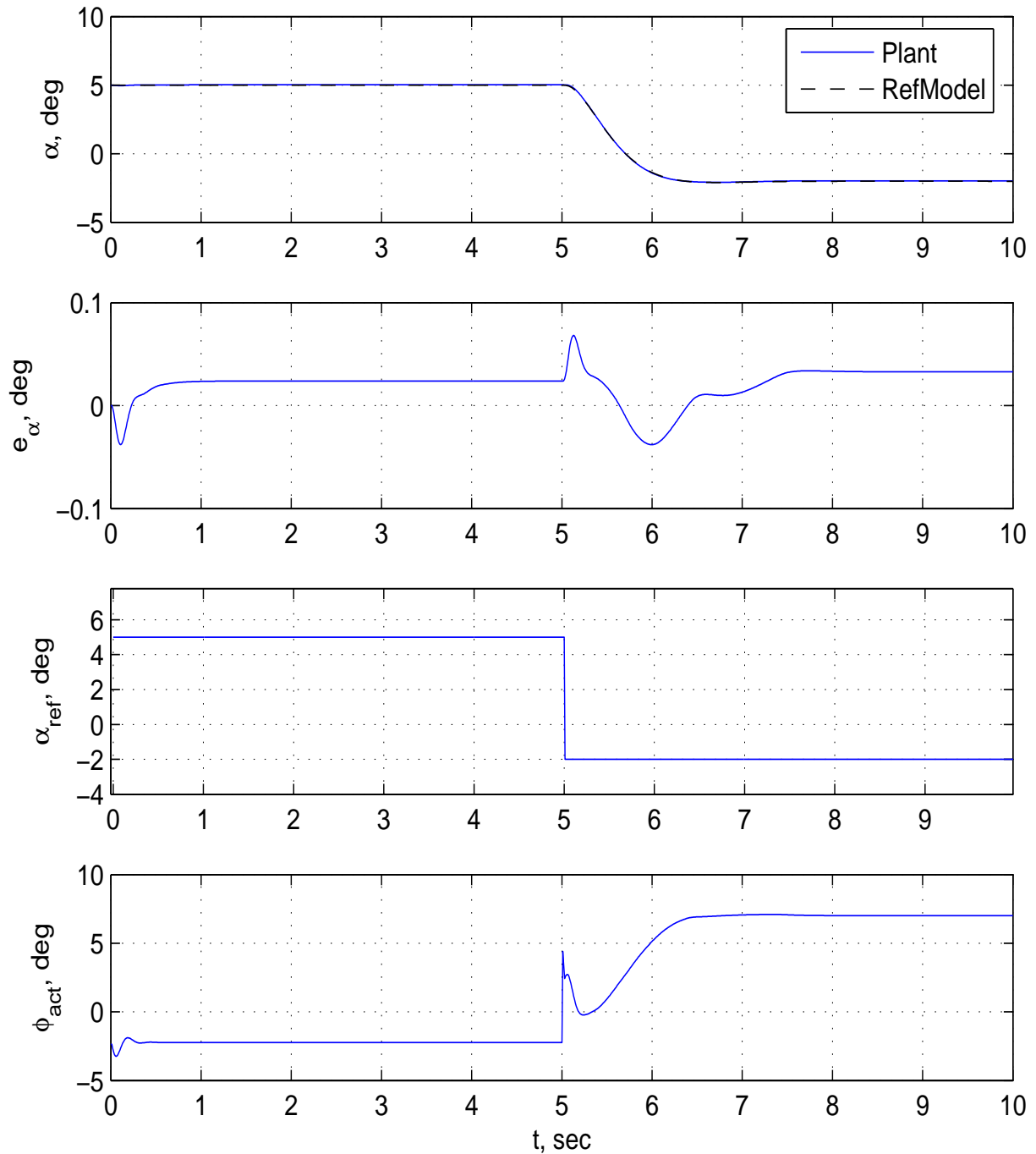


Рис. А.14. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором при ступенчатом изменении задающего сигнала (ГЗЛА Х-43, режим полета $M = 6$, вариант 2). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

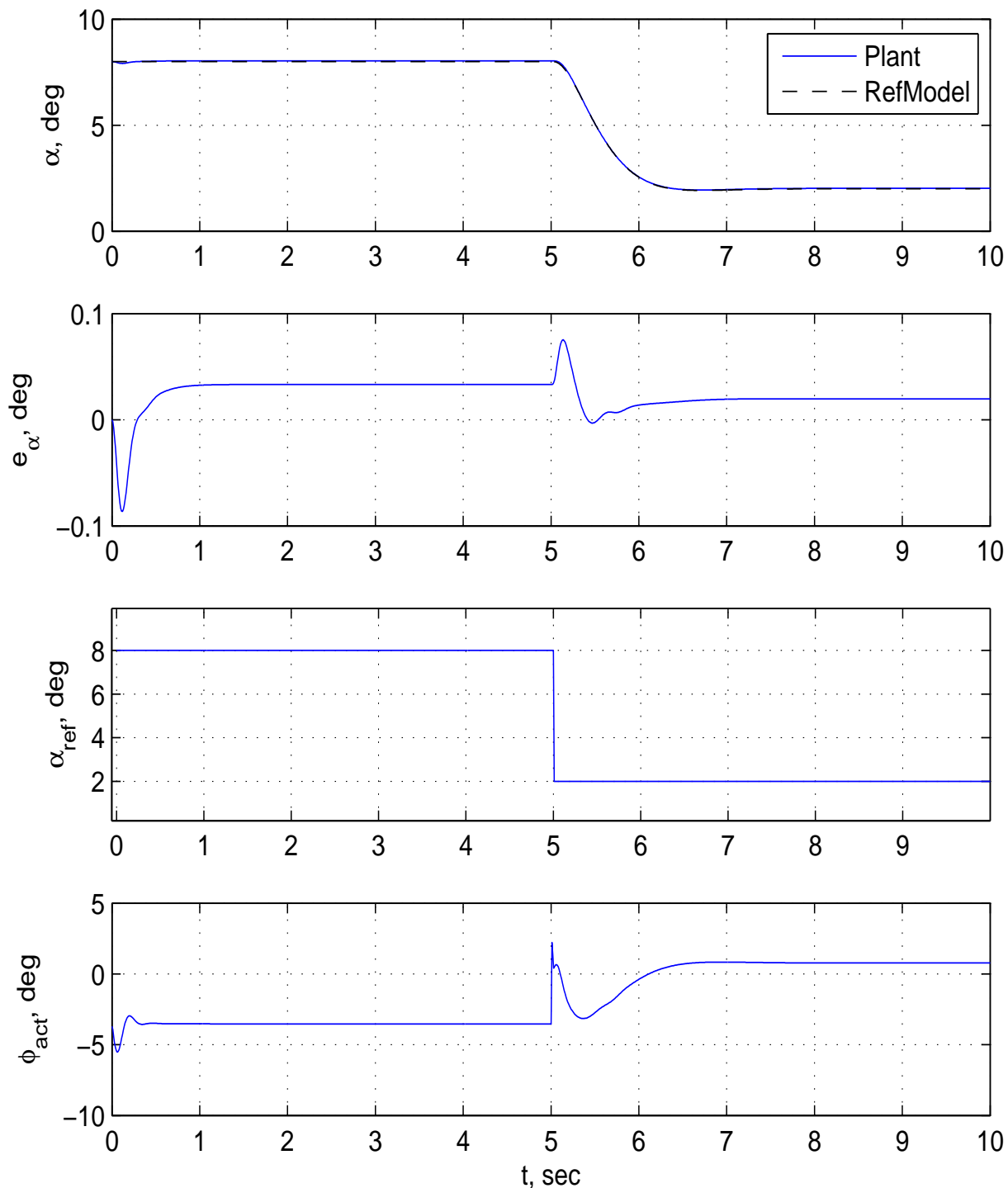


Рис. А.15. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором при ступенчатом изменении задающего сигнала (ГЗЛА Х-43, режим полета $M = 6$, вариант 3). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

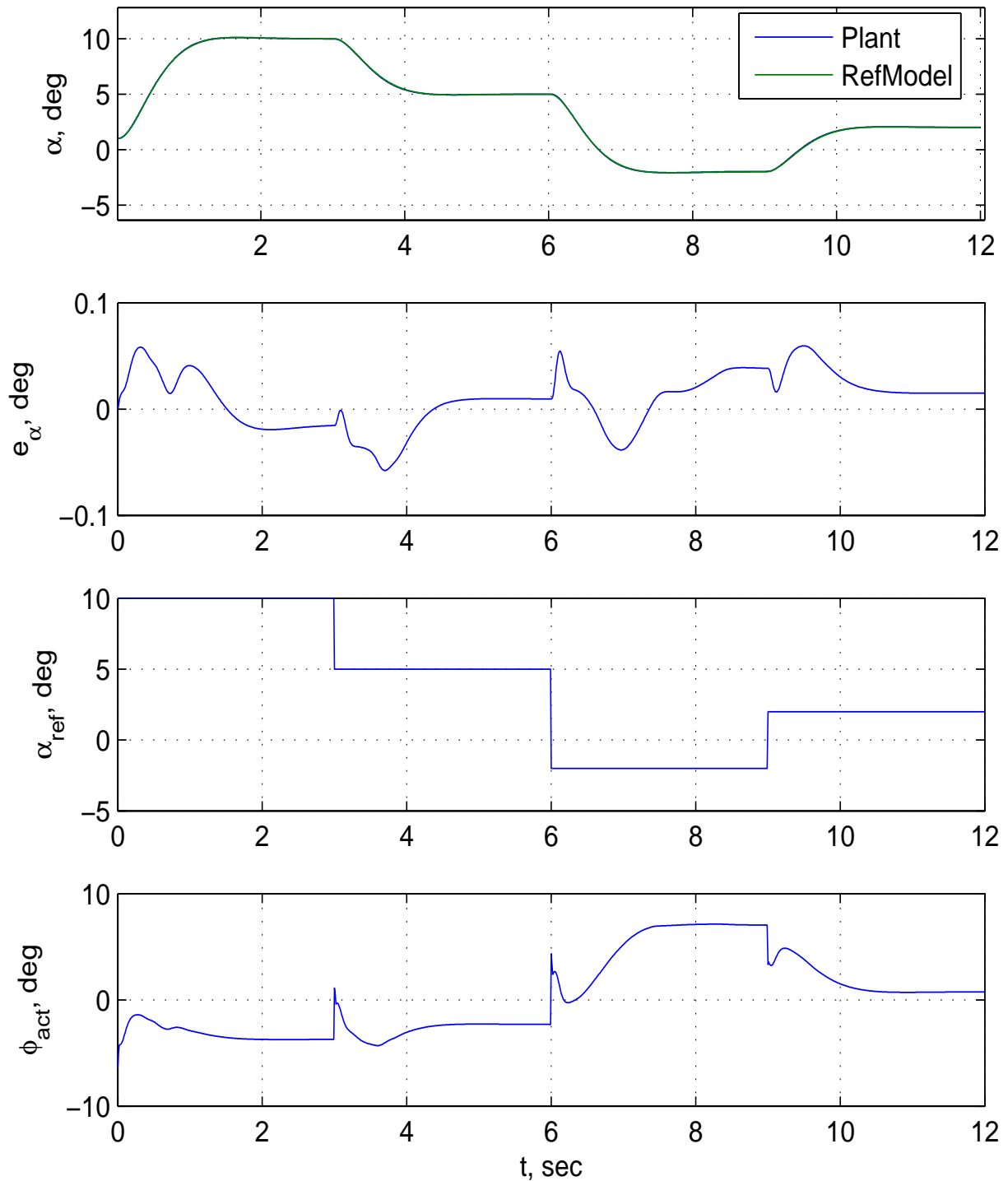


Рис. А.16. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором при последовательности ступенчатых изменений задающего сигнала (ГЗЛА Х-43, режим полета $M = 6$). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

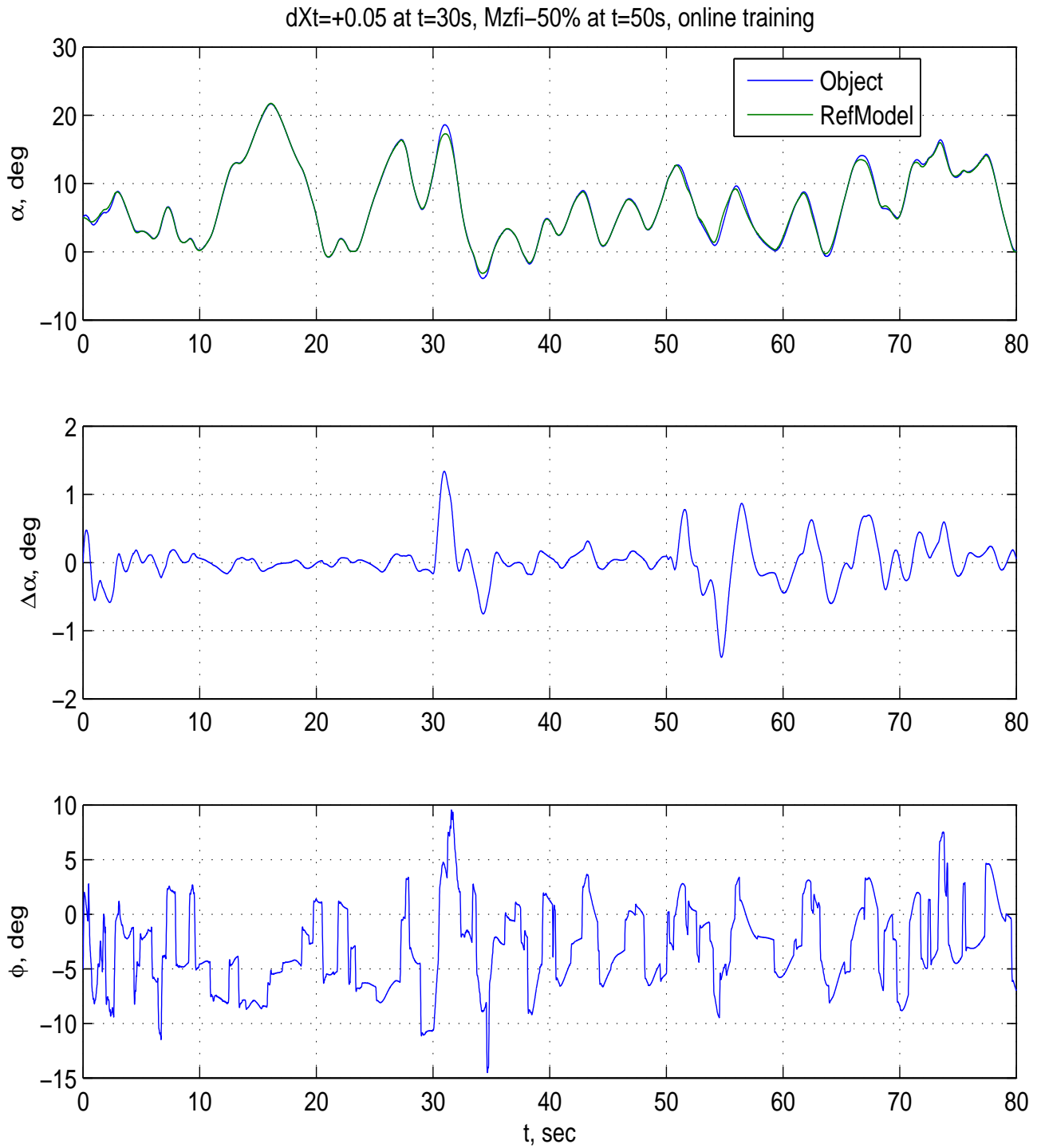


Рис. А.17. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором (самолет F-16, режим полета с индикаторной скоростью $V_{ind} = 300$ км/ч). Адаптация к изменению динамики объекта управления: смещение центровки на 10% назад ($t = 30$ с), 50% уменьшение эффективности органа управления ($t = 50$ с). Обозначения: α — угол атаки, град.; $\Delta\alpha$ — ошибка отслеживания заданного угла атаки, град.; ϕ — угол отклонения стабилизатора, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

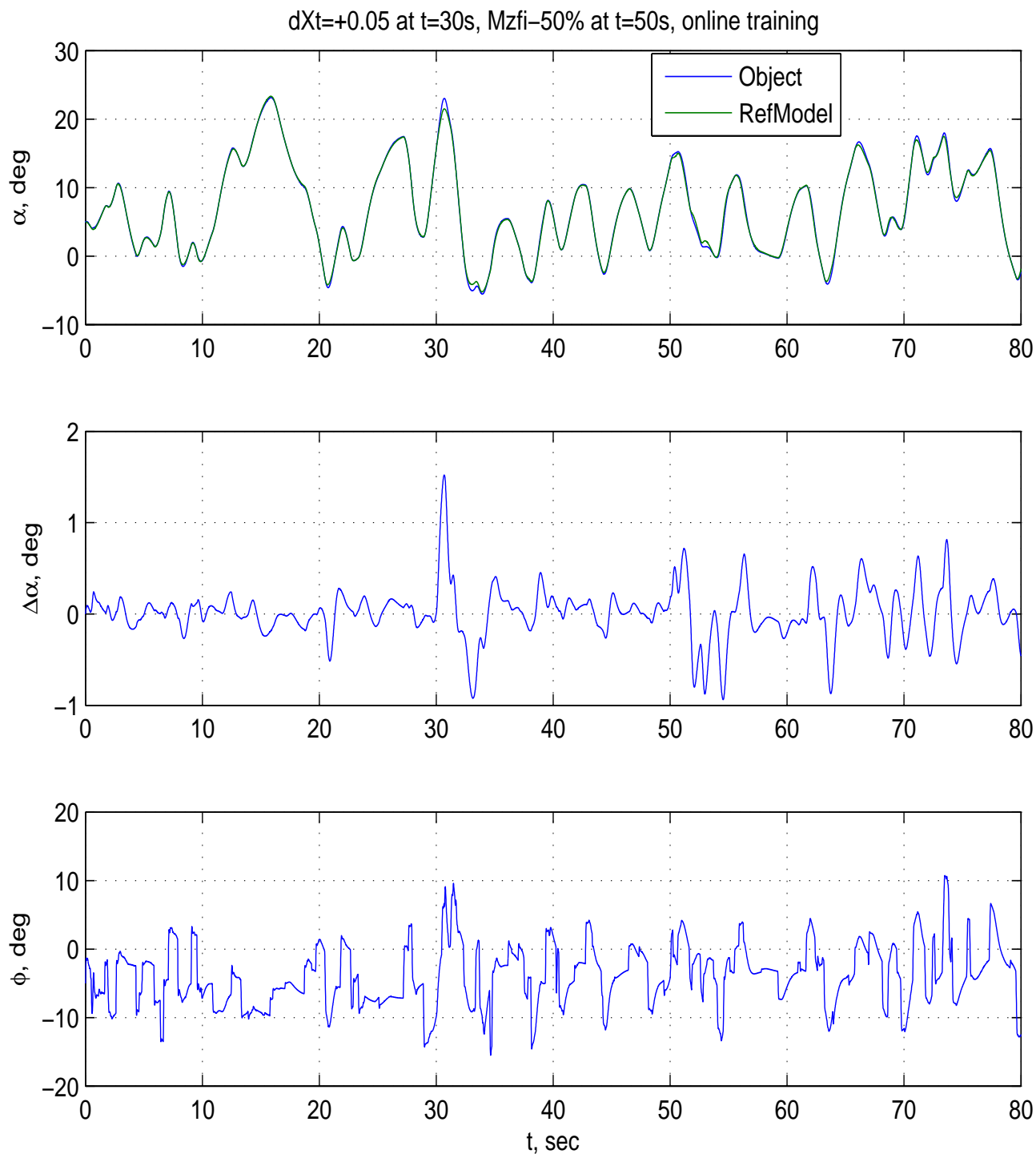


Рис. А.18. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором (самолет F-16, режим полета с индикаторной скоростью $V_{ind} = 400$ км/ч). Адаптация к изменению динамики объекта управления: смещение центровки на 5% назад ($t = 30$ с), 50% уменьшение эффективности органа управления ($t = 50$ с). Обозначения: α — угол атаки, град.; $\Delta\alpha$ — ошибка отслеживания заданного угла атаки, град.; ϕ — угол отклонения стабилизатора, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

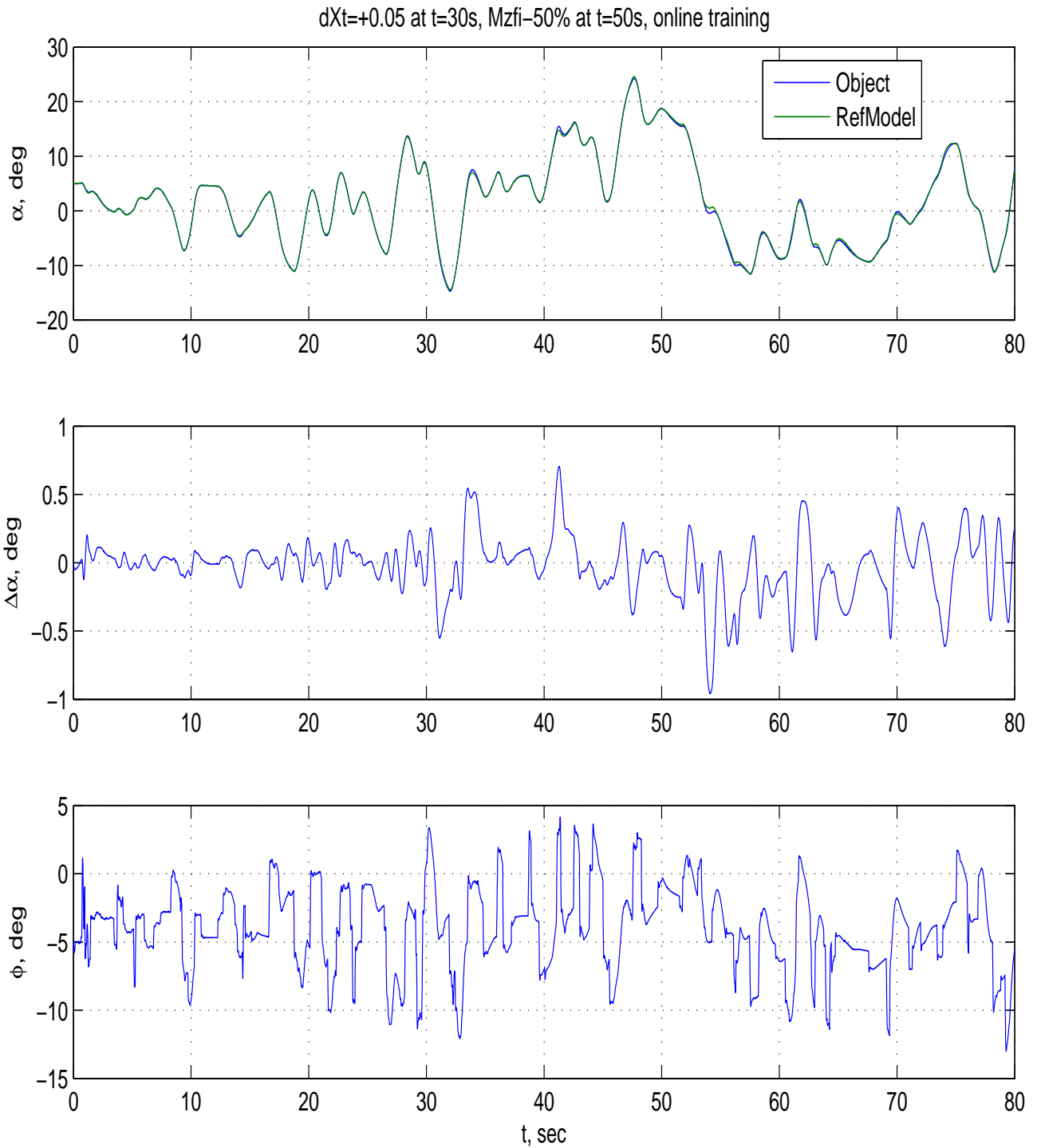


Рис. А.19. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором (самолет F-16, режим полета с индикаторной скоростью $V_{ind} = 500$ км/ч). Адаптация к изменению динамики объекта управления: смещение центровки на 5% назад ($t = 30$ с), 50% уменьшение эффективности органа управления ($t = 50$ с). Обозначения: α — угол атаки, град.; $\Delta\alpha$ — ошибка отслеживания заданного угла атаки, град.; ϕ — угол отклонения стабилизатора, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

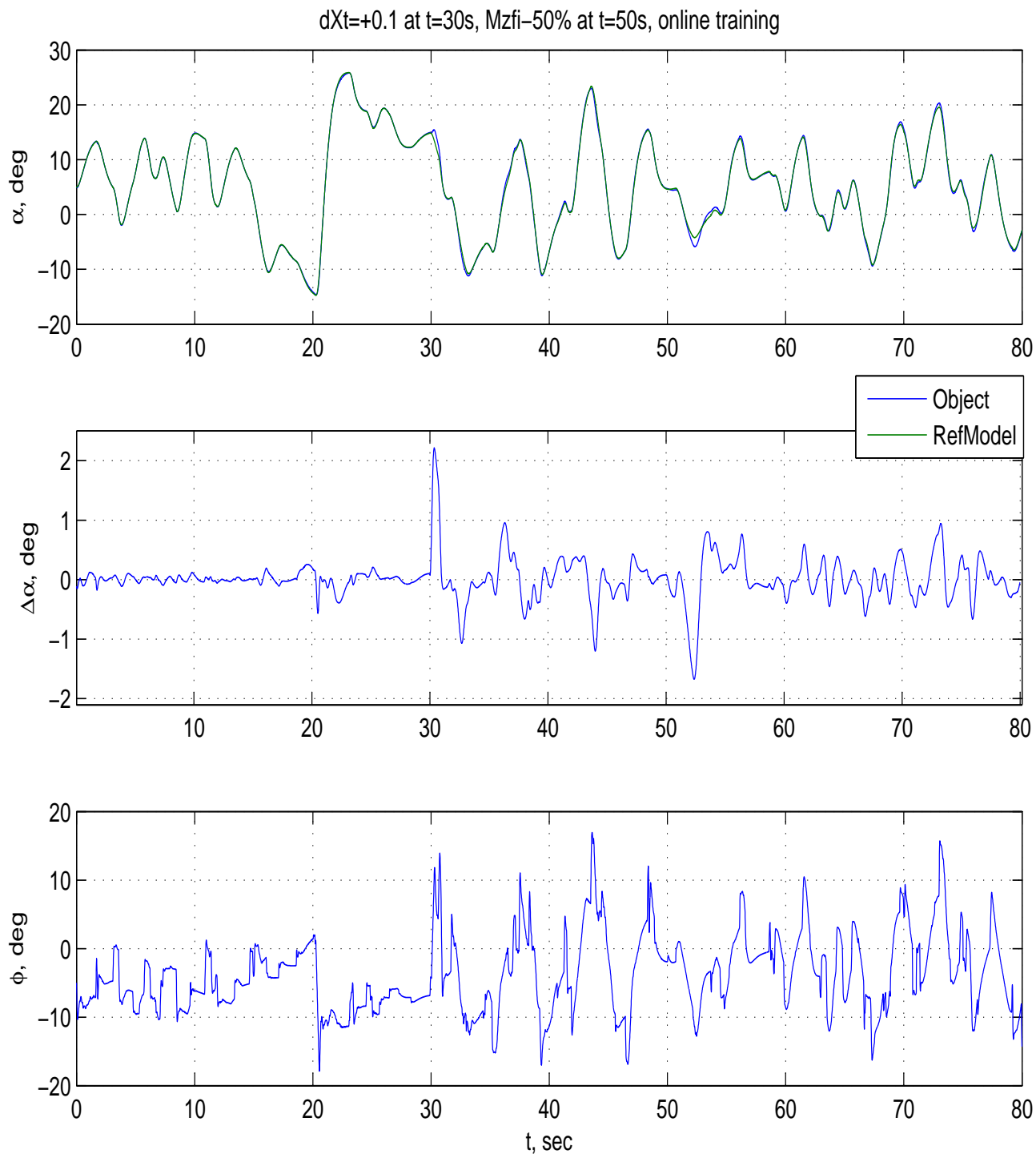


Рис. А.20. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором (самолет F-16, режим полета с индикаторной скоростью $V_{ind} = 600$ км/ч). Адаптация к изменению динамики объекта управления: смещение центровки на 10% назад ($t = 30$ с), 50% уменьшение эффективности органа управления ($t = 50$ с). Обозначения: α — угол атаки, град.; $\Delta\alpha$ — ошибка отслеживания заданного угла атаки, град.; ϕ — угол отклонения стабилизатора, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

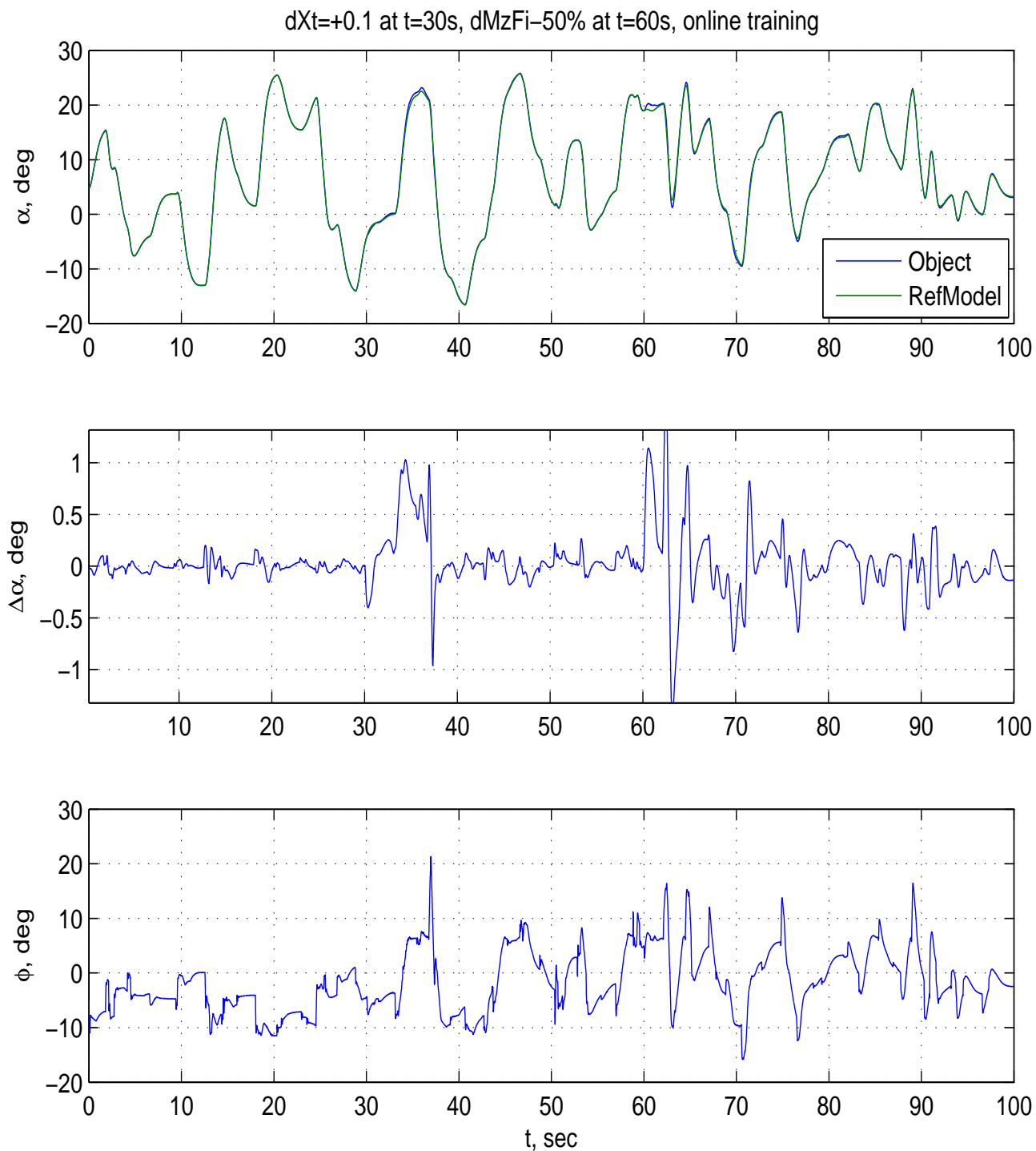


Рис. А.21. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором (самолет F-16, режим полета с индикаторной скоростью $V_{ind} = 700$ км/ч). Адаптация к изменению динамики объекта управления: смещение центровки на 10% назад ($t = 30$ с), 50% уменьшение эффективности органа управления ($t = 60$ с). Обозначения: α — угол атаки, град.; $\Delta\alpha$ — ошибка отслеживания заданного угла атаки, град.; ϕ — угол отклонения стабилизатора, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

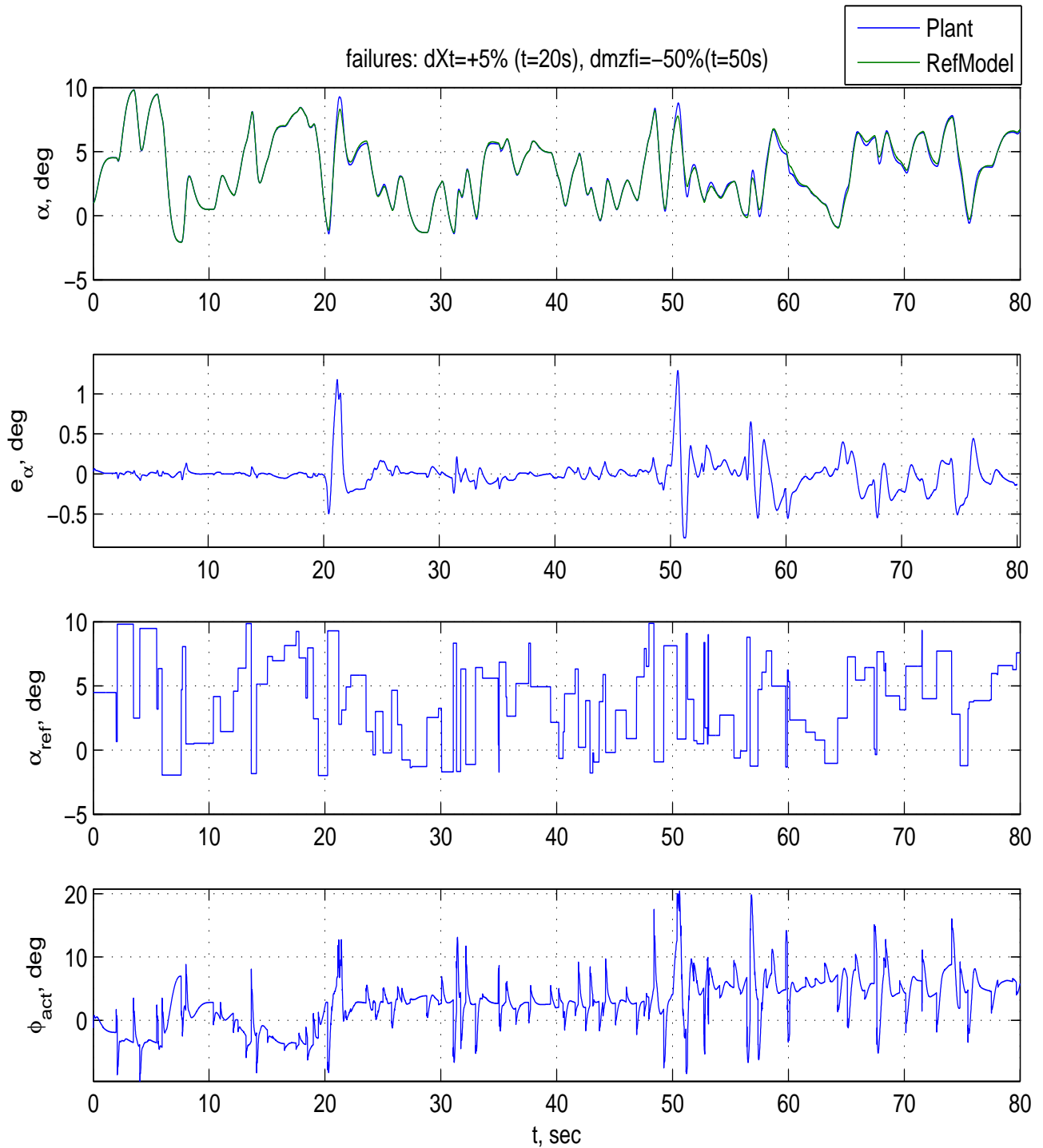


Рис. А.22. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором (ГЗЛА Х-43, режим полета $M = 6$). Адаптация к изменению динамики объекта управления: смещение центровки на 5% назад ($t = 20$ с), 50% уменьшение эффективности органа управления ($t = 50$ с). Обозначения: α — угол атаки, град.; e_{α} — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

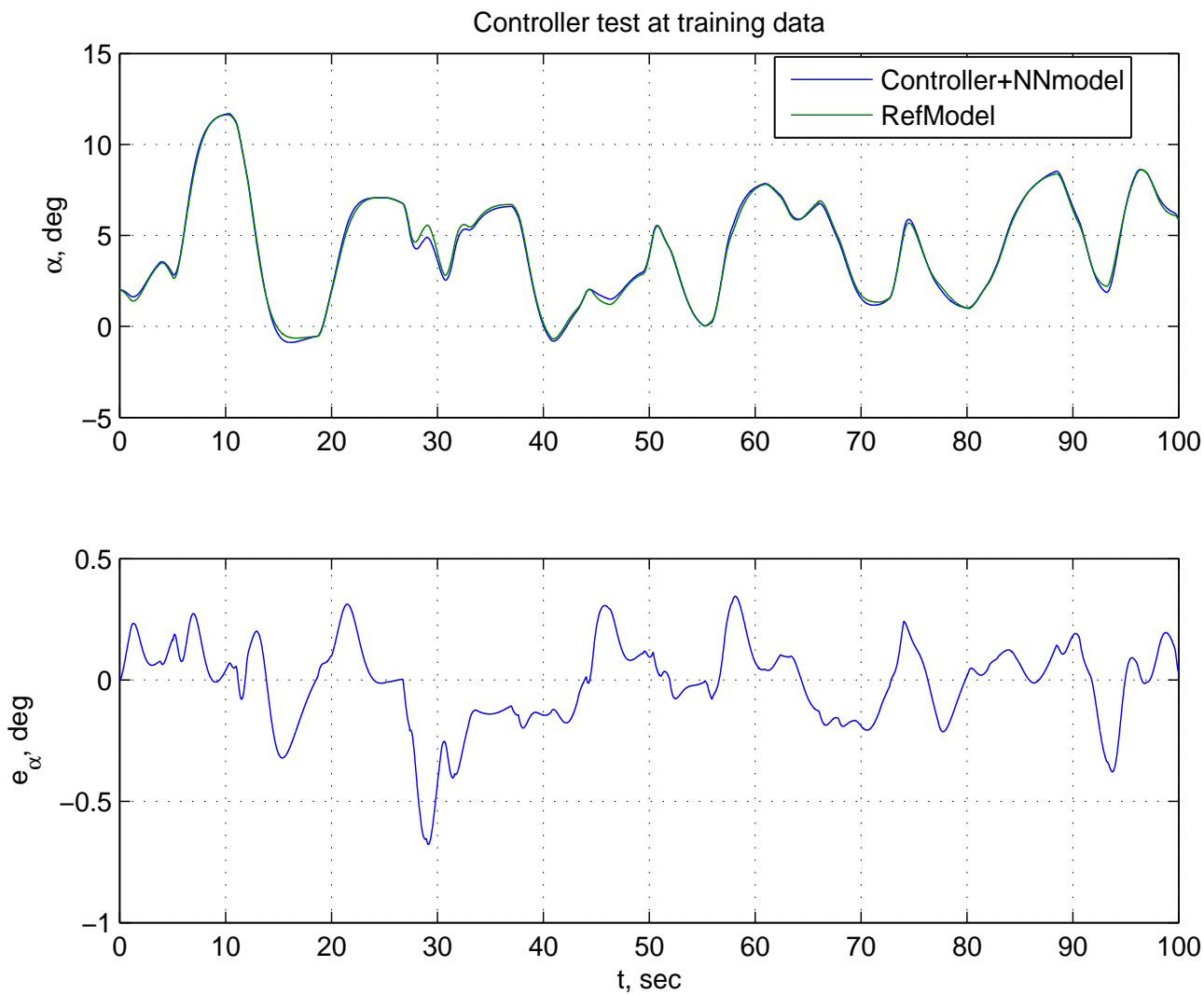


Рис. А.23. Проверка работоспособности нейроконтроллера на тестовом наборе (для ГЗЛА NASP, режим полета $M = 6$). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

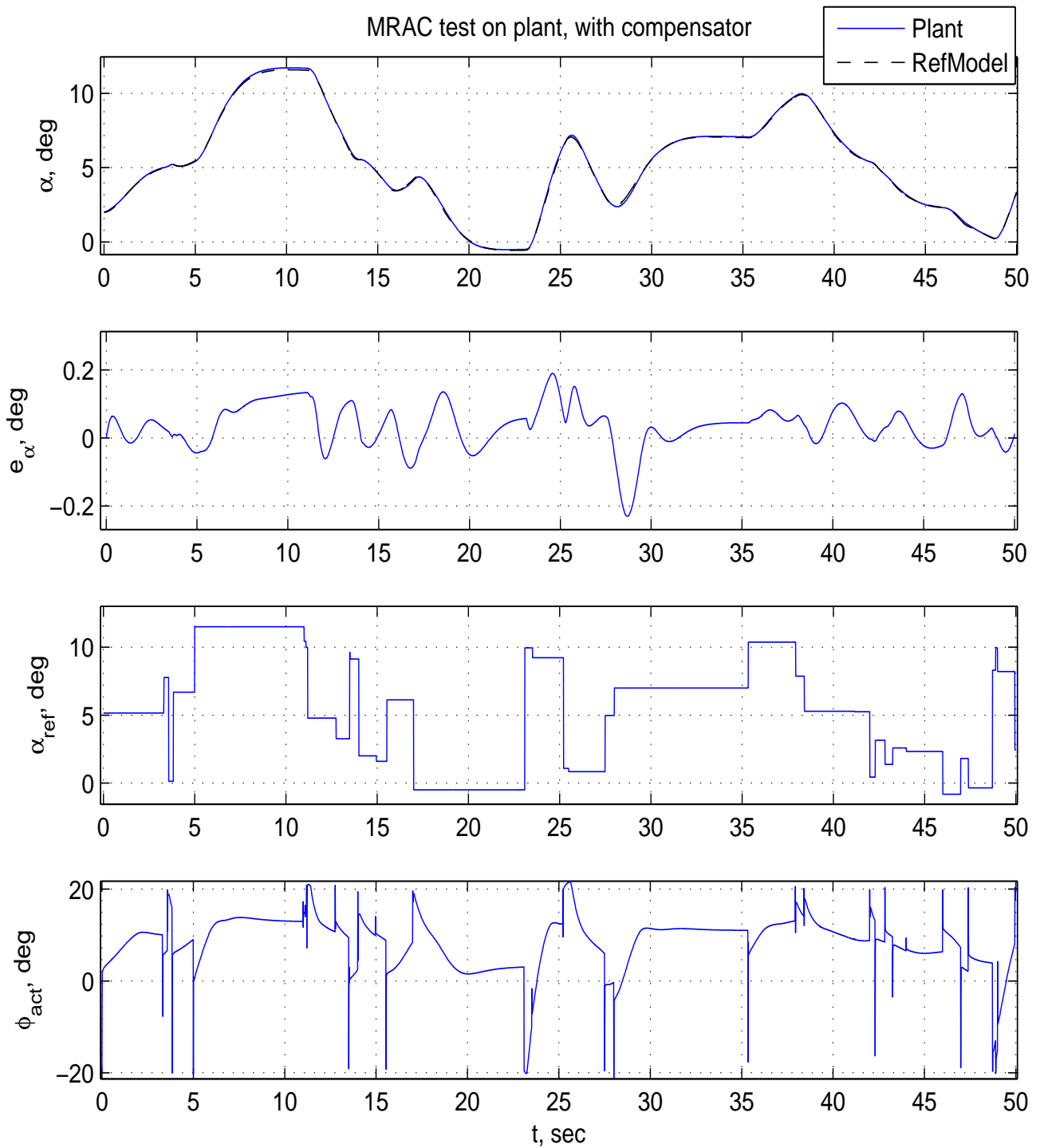


Рис. А.24. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором (для ГЗЛА NASP, режим полета $M = 6$). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

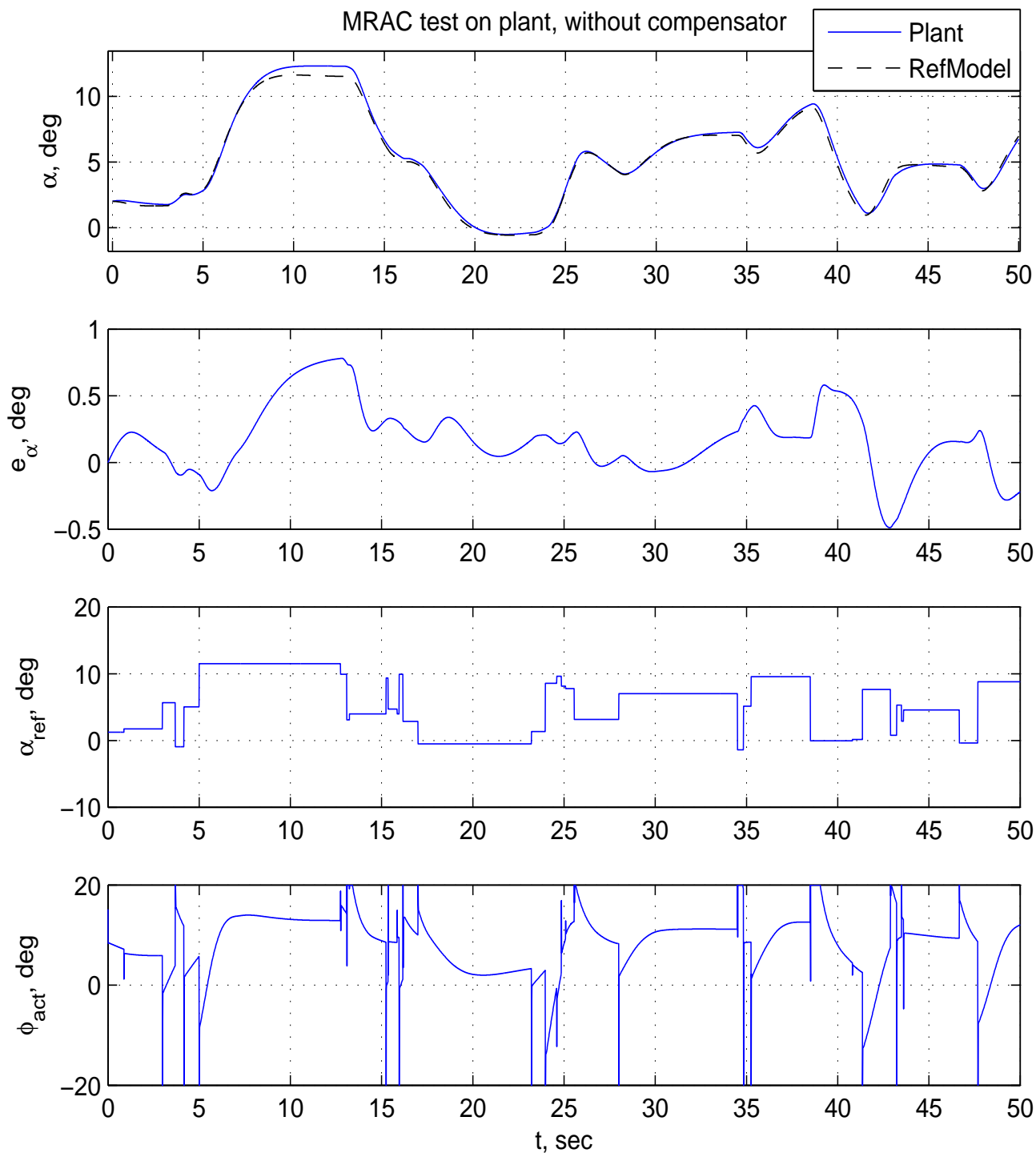


Рис. А.25. Результаты вычислительного эксперимента для системы управления с эталонной моделью, без компенсатора (для ГЗЛА NASP, режим полета $M = 6$). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

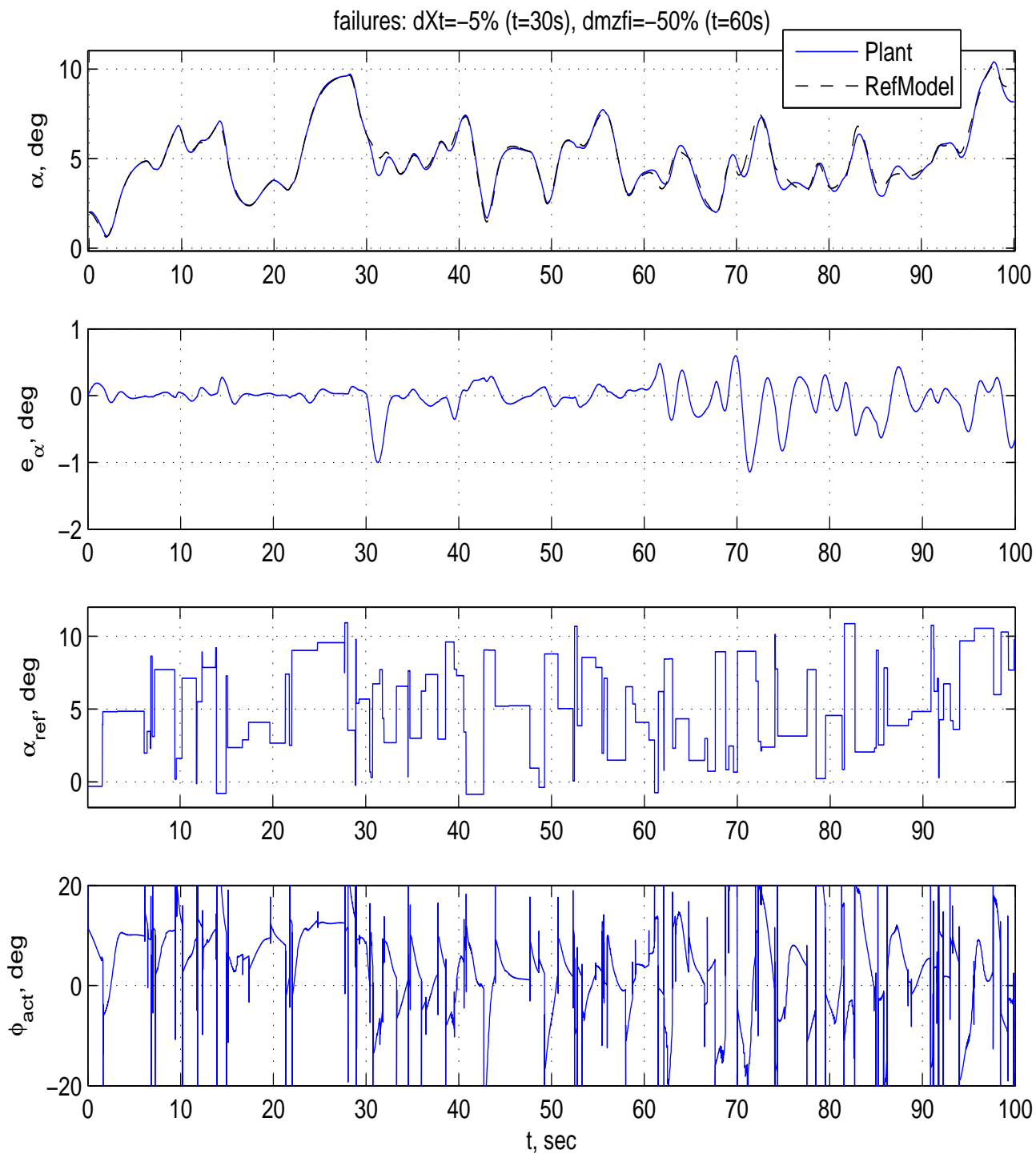


Рис. А.26. Результаты вычислительного эксперимента для системы управления с эталонной моделью и компенсатором (для ГЗЛА NASP, режим полета $M = 6$). Адаптация к изменению динамики объекта управления: смещение центровки на 5% назад ($t = 30$ с), 50% уменьшение эффективности органа управления ($t = 60$ с). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

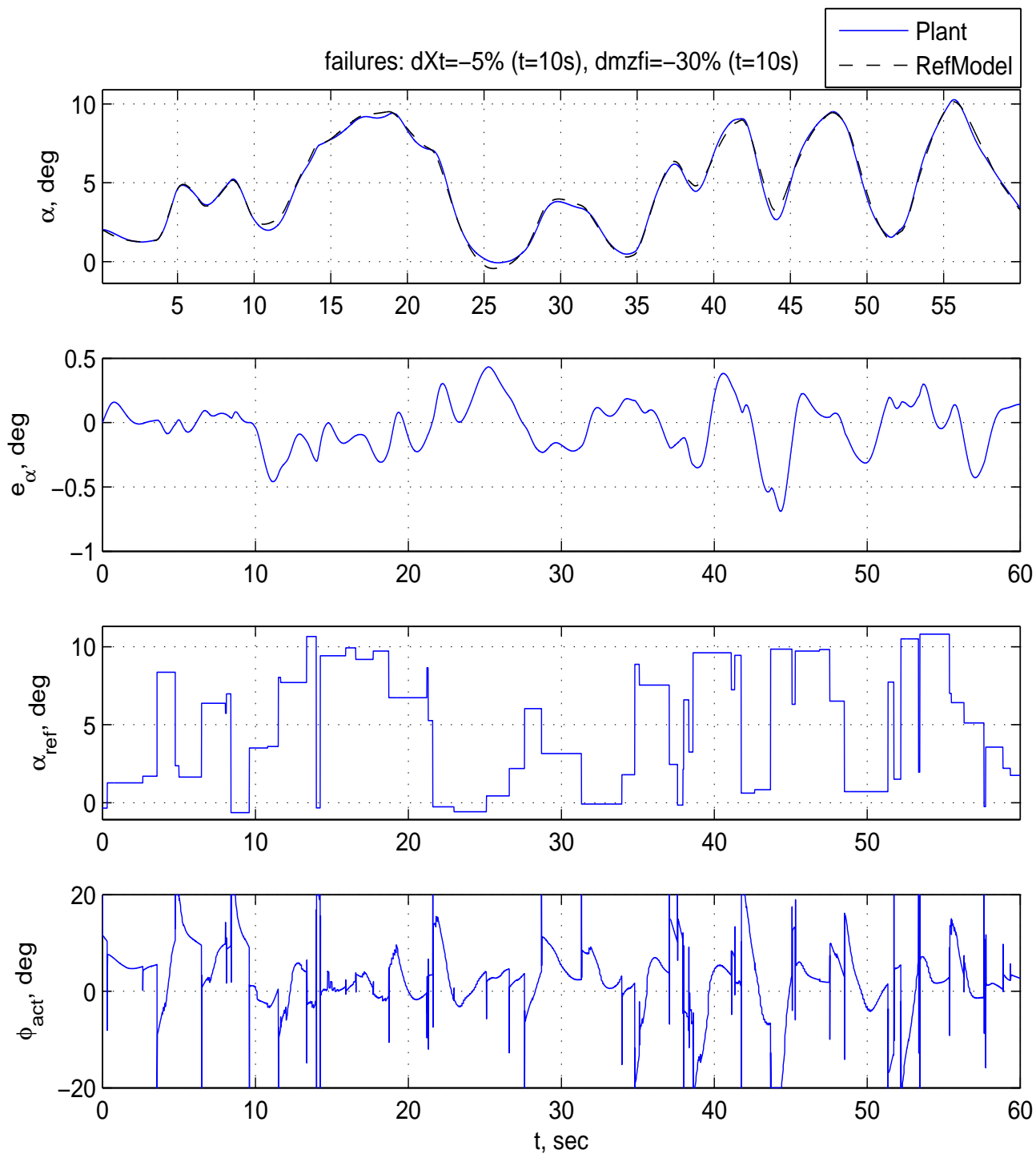


Рис. А.27. Результаты вычислительного эксперимента для системы управления с эталонной моделью с компенсатором (для ГЗЛА NASP, режим полета $M = 6$). Адаптация к двум одновременным отказам: смещение центровки на 5% назад и 50% уменьшение эффективности органа управления при $t = 10$ с. Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

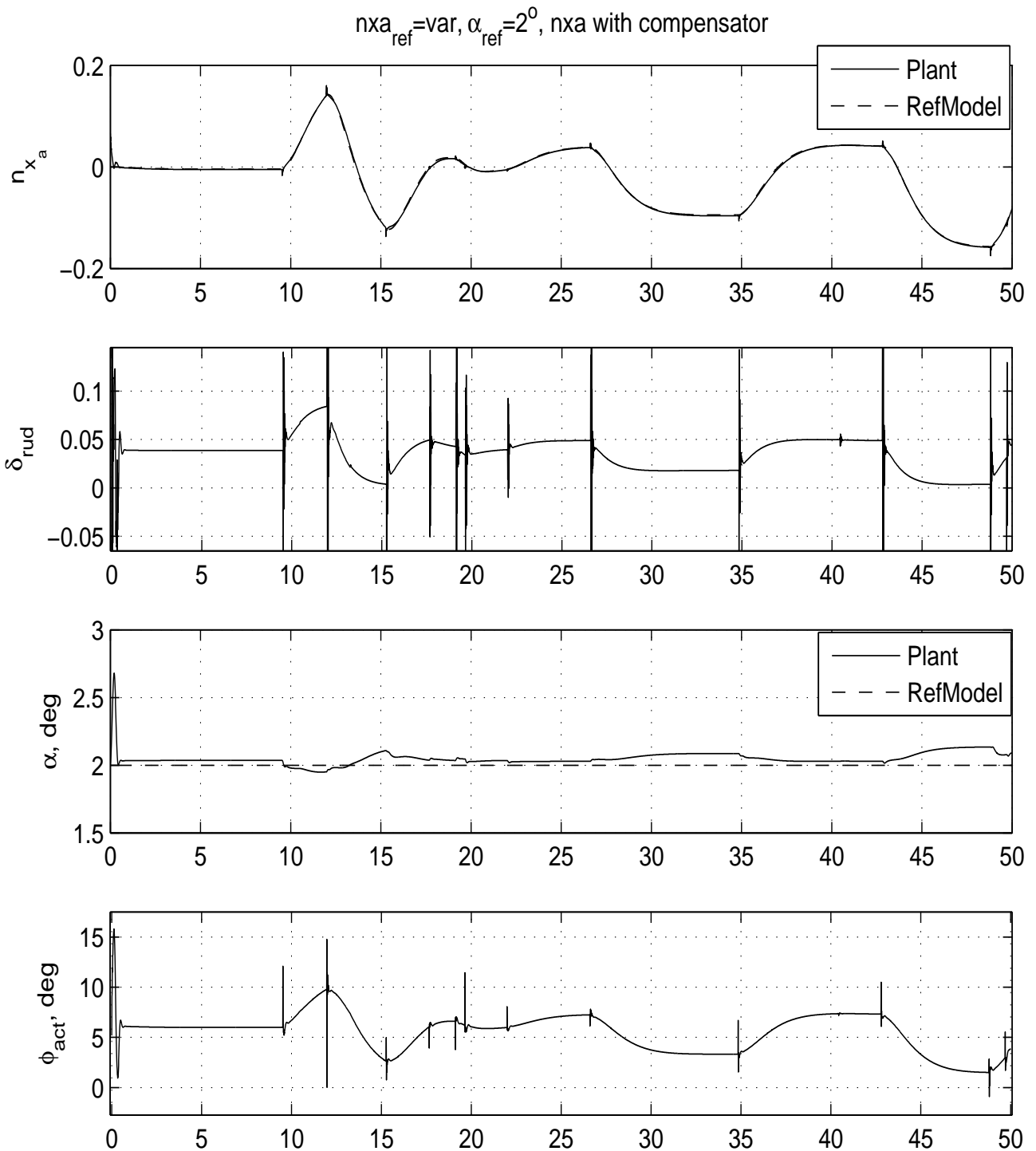


Рис. А.28. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА Х-43, режим полета $M = 6$). Постоянная перегрузка $n_{x_a} = 0$, заданный угол атаки — случайный. Компенсаторы в обоих каналах. I — Поведение объекта и эталонной модели. Обозначения: n_{x_a} — тангенциальная перегрузка; δ_{rud} — командный сигнал управления двигателем; α — угол атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

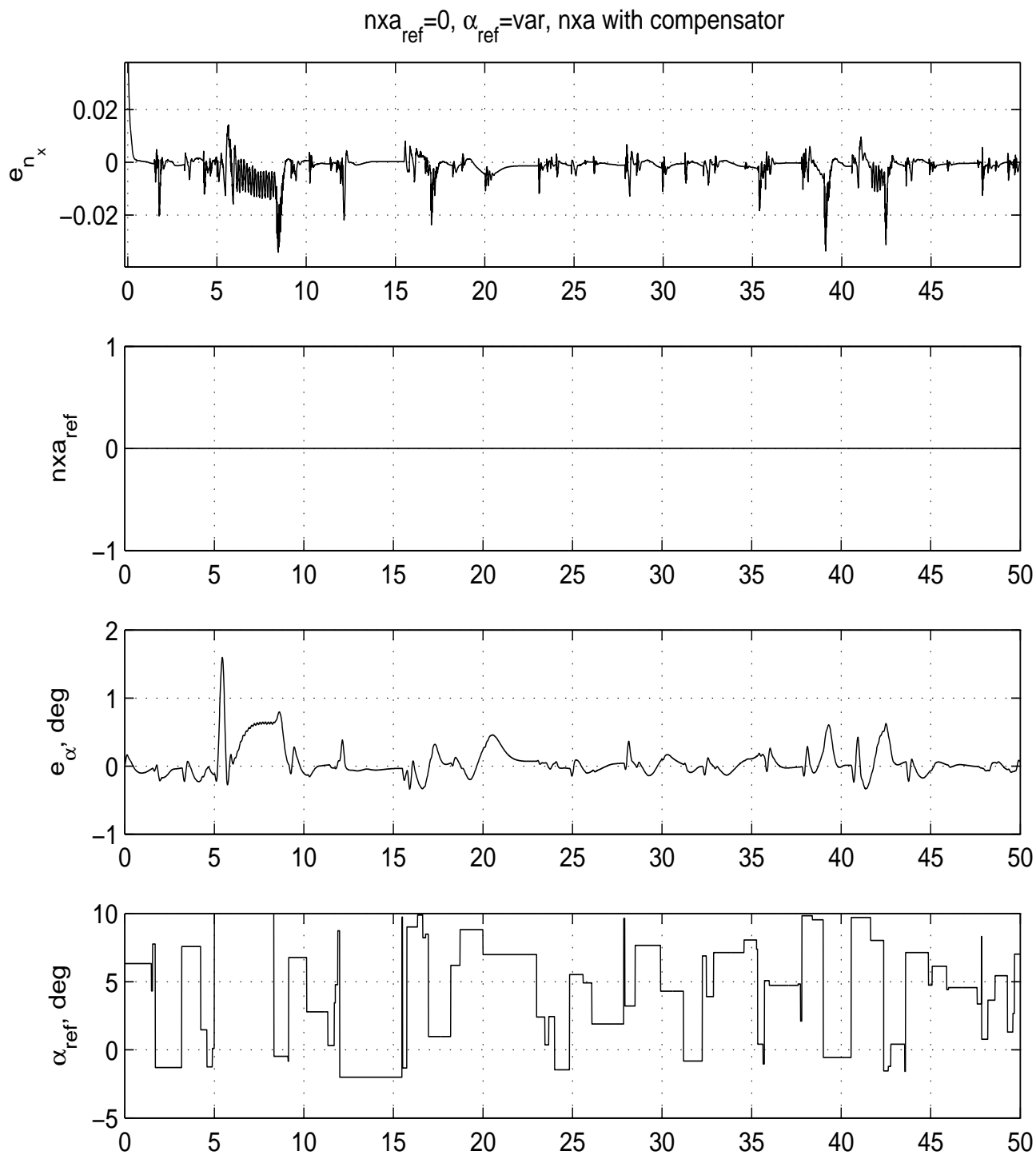


Рис. А.29. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА Х-43, режим полета $M = 6$). Постоянная перегрузка $n_{x_a} = 0$, заданный угол атаки — случайный. Компенсаторы в обоих каналах. II — Задающие сигналы и ошибки отслеживания. Обозначения: $e_{n_{x_a}}$ — ошибка по тангенциальной перегрузке; $n_{x_a,ref}$ — задающий сигнал по тангенциальной перегрузке; e_{α} — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

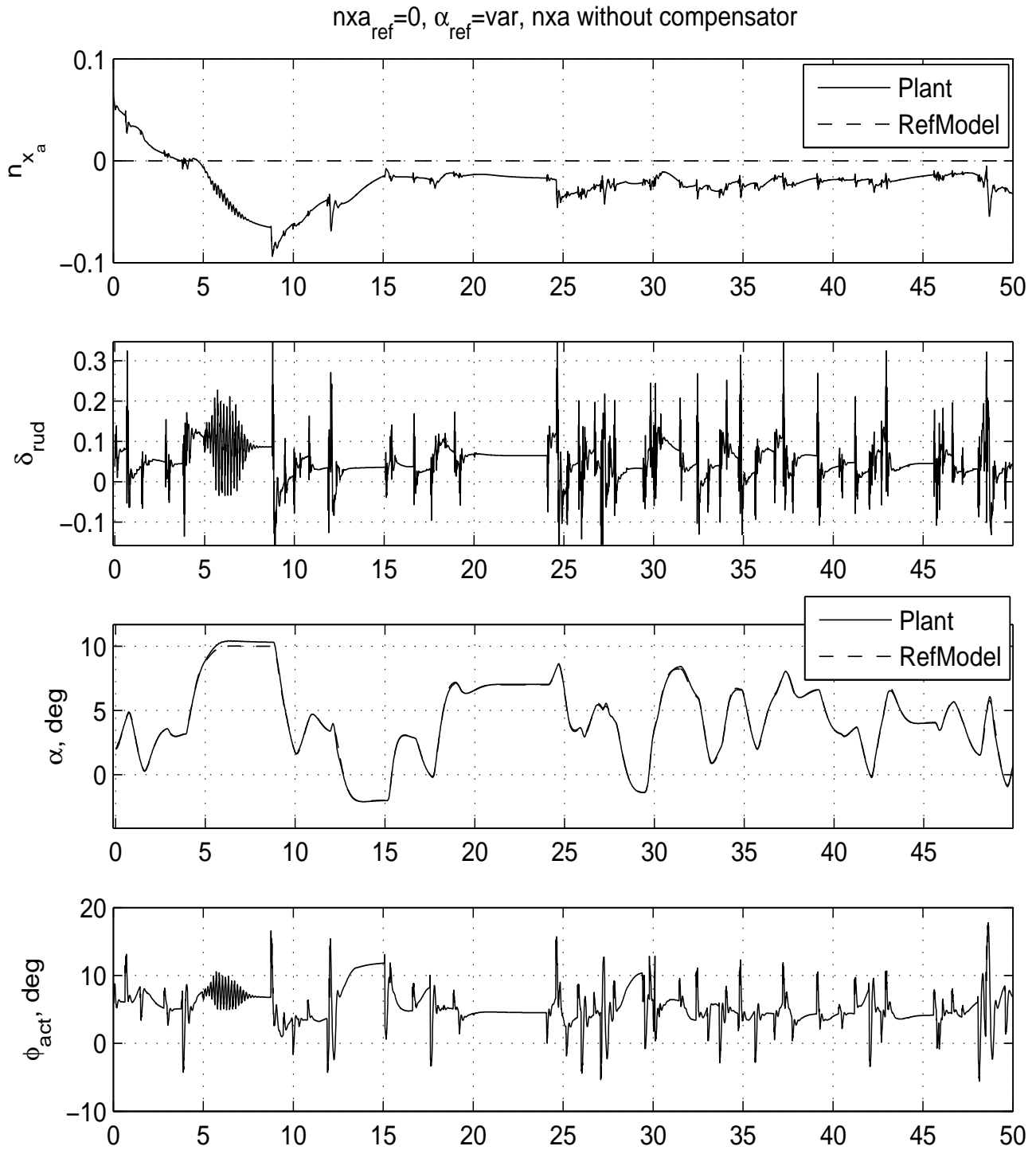


Рис. А.30. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА Х-43, режим полета $M = 6$). Постоянная перегрузка $n_{x_a} = 0$. В канале перегрузки нет компенсатора. I — Поведение объекта и эталонной модели. Обозначения: n_{x_a} — тангенциальная перегрузка; δ_{rud} — командный сигнал управления двигателем; α — угол атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

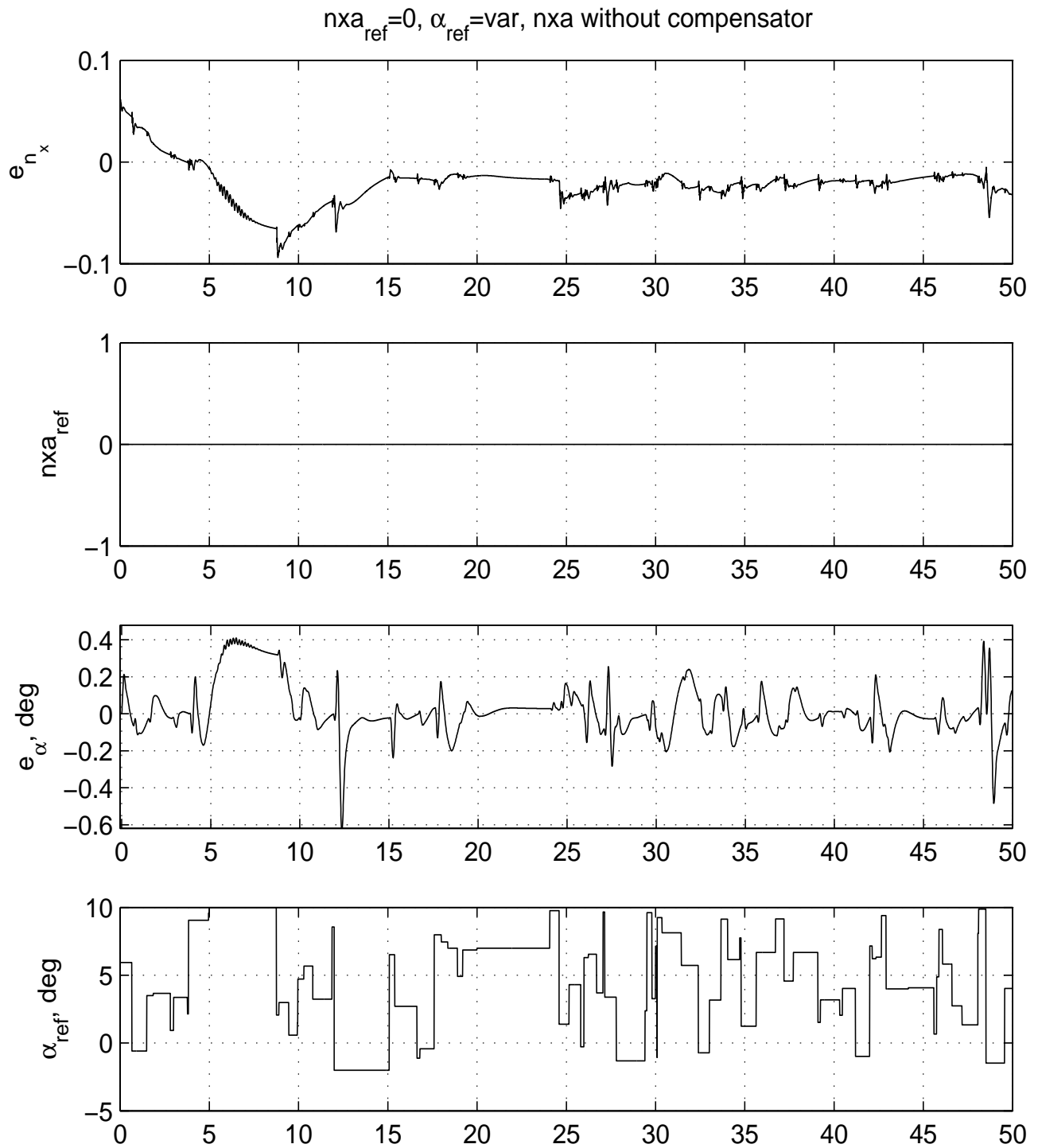


Рис. А.31. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА X-43, режим полета $M = 6$). Постоянная перегрузка $n_{x_a} = 0$. В канале перегрузки нет компенсатора. II — Задающие сигналы и ошибки отслеживания. Обозначения: $e_{n_{x_a}}$ — ошибка по тангенциальной перегрузке; $n_{x_a,ref}$ — задающий сигнал по тангенциальной перегрузке; e_{α} — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

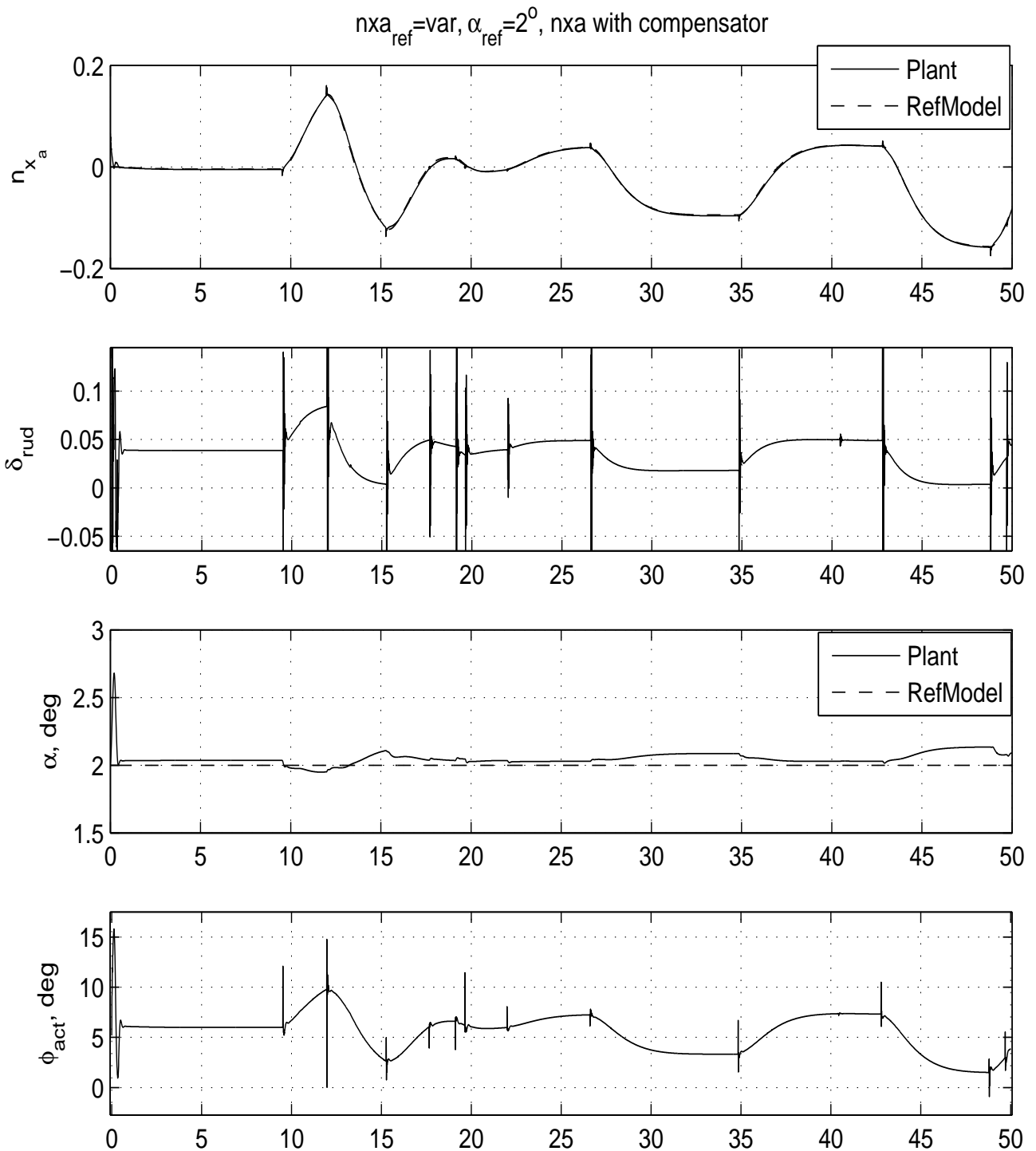


Рис. А.32. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА X-43, режим полета $M = 6$). Постоянный заданный угол атаки $\alpha = 2$ град., случайная заданная перегрузка. Компенсаторы в обоих каналах. I — Поведение объекта и эталонной модели. Обозначения: n_{x_a} — тангенциальная перегрузка; δ_{rud} — командный сигнал управления двигателем; α — угол атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

$n_{x_{ref}} = \text{var}, \alpha_{ref} = 2^\circ, n_x$ with compensator

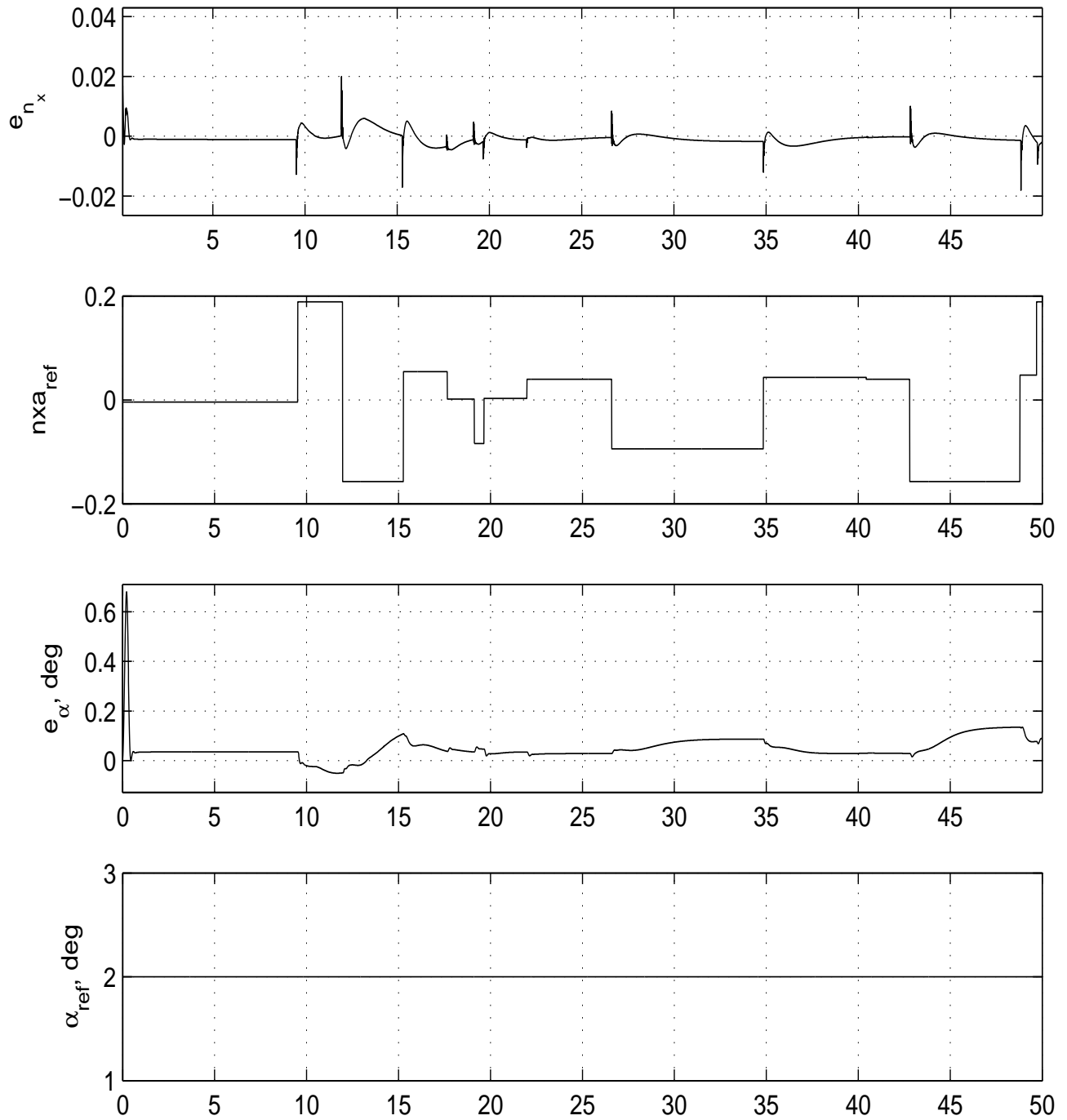


Рис. А.33. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА Х-43, режим полета $M = 6$). Постоянный заданный угол атаки $\alpha = 2$ град., случайная заданная перегрузка. Компенсаторы в обоих каналах. II — Задающие сигналы и ошибки отслеживания. Обозначения: $e_{n_{x_a}}$ — ошибка по тангенциальной перегрузке; $n_{x_{a,ref}}$ — задающий сигнал по тангенциальной перегрузке; e_{α} — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

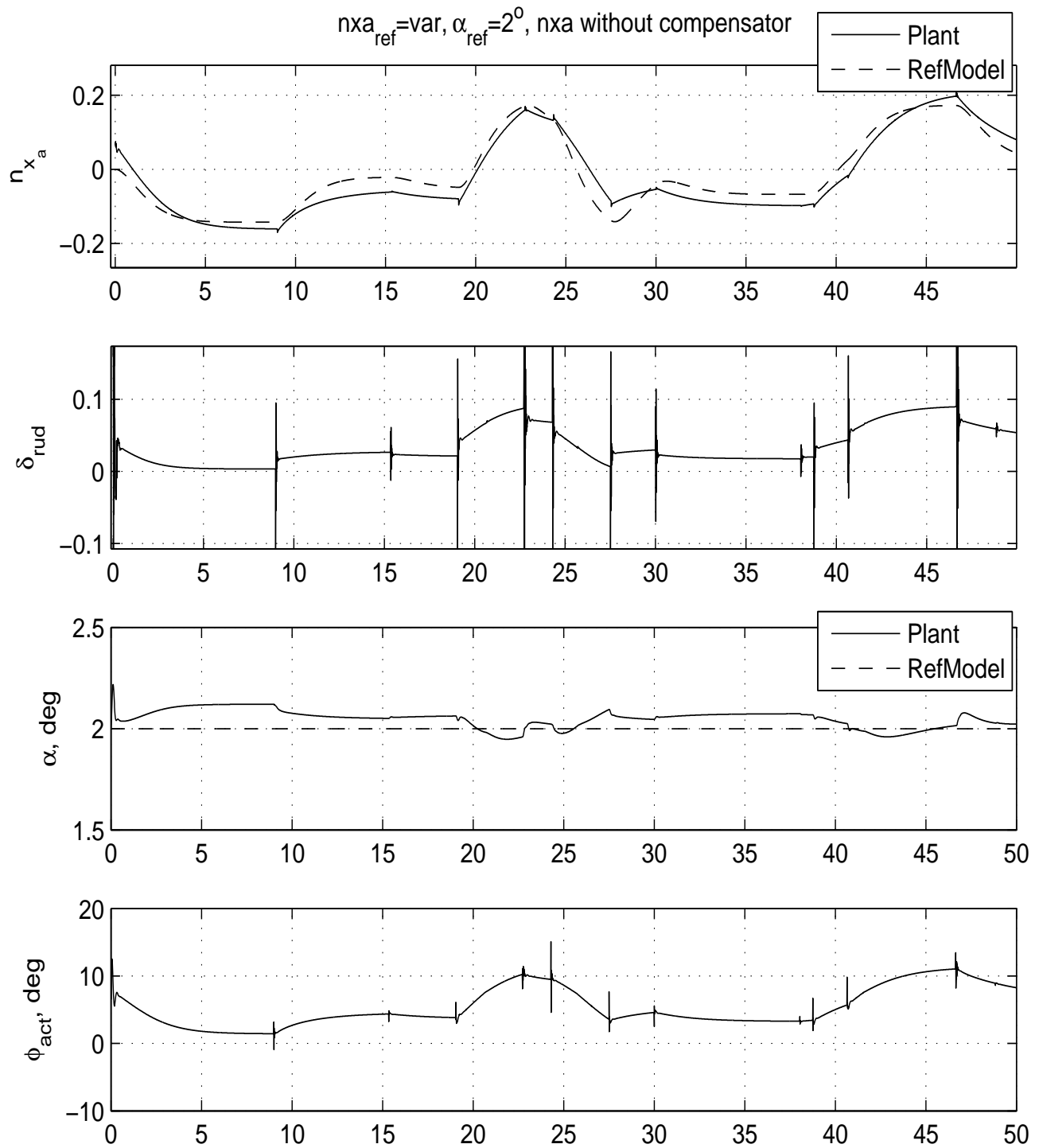


Рис. А.34. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА X-43, режим полета $M = 6$). Постоянный заданный угол атаки $\alpha = 2$ град., случайная заданная перегрузка. Нет компенсатора в канале перегрузки. I — Поведение объекта и эталонной модели. Обозначения: n_{x_a} — тангенциальная перегрузка; δ_{rud} — командный сигнал управления двигателем; α — угол атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

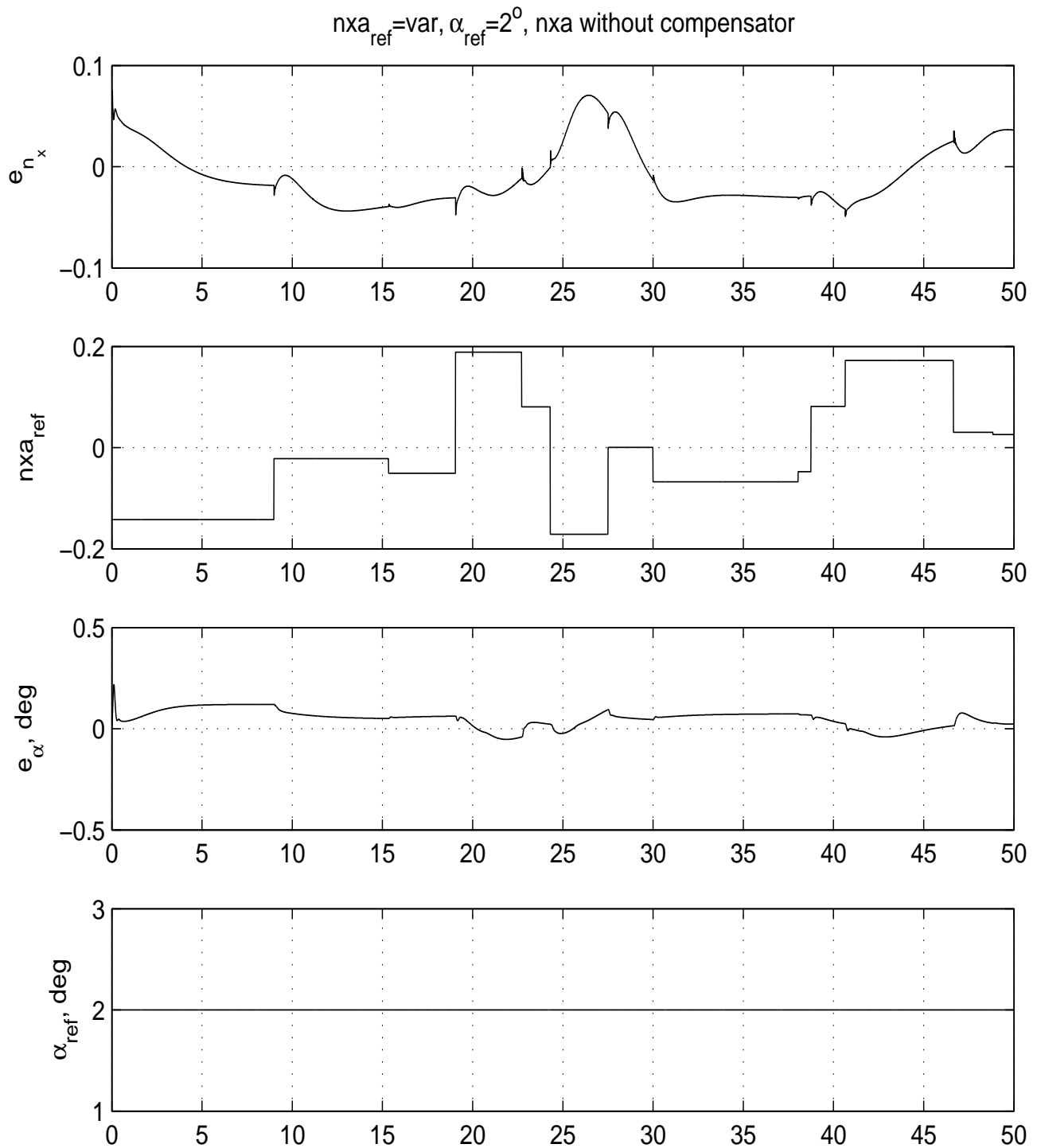


Рис. А.35. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА Х-43, режим полета $M = 6$). Постоянный заданный угол атаки $\alpha = 2$ град., случайная заданная перегрузка. Нет компенсатора в канале перегрузки. II — Задающие сигналы и ошибки отслеживания. Обозначения: $e_{n_{xa}}$ — ошибка по тангенциальной перегрузке; $n_{x_{a,ref}}$ — задающий сигнал по тангенциальной перегрузке; e_{α} — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

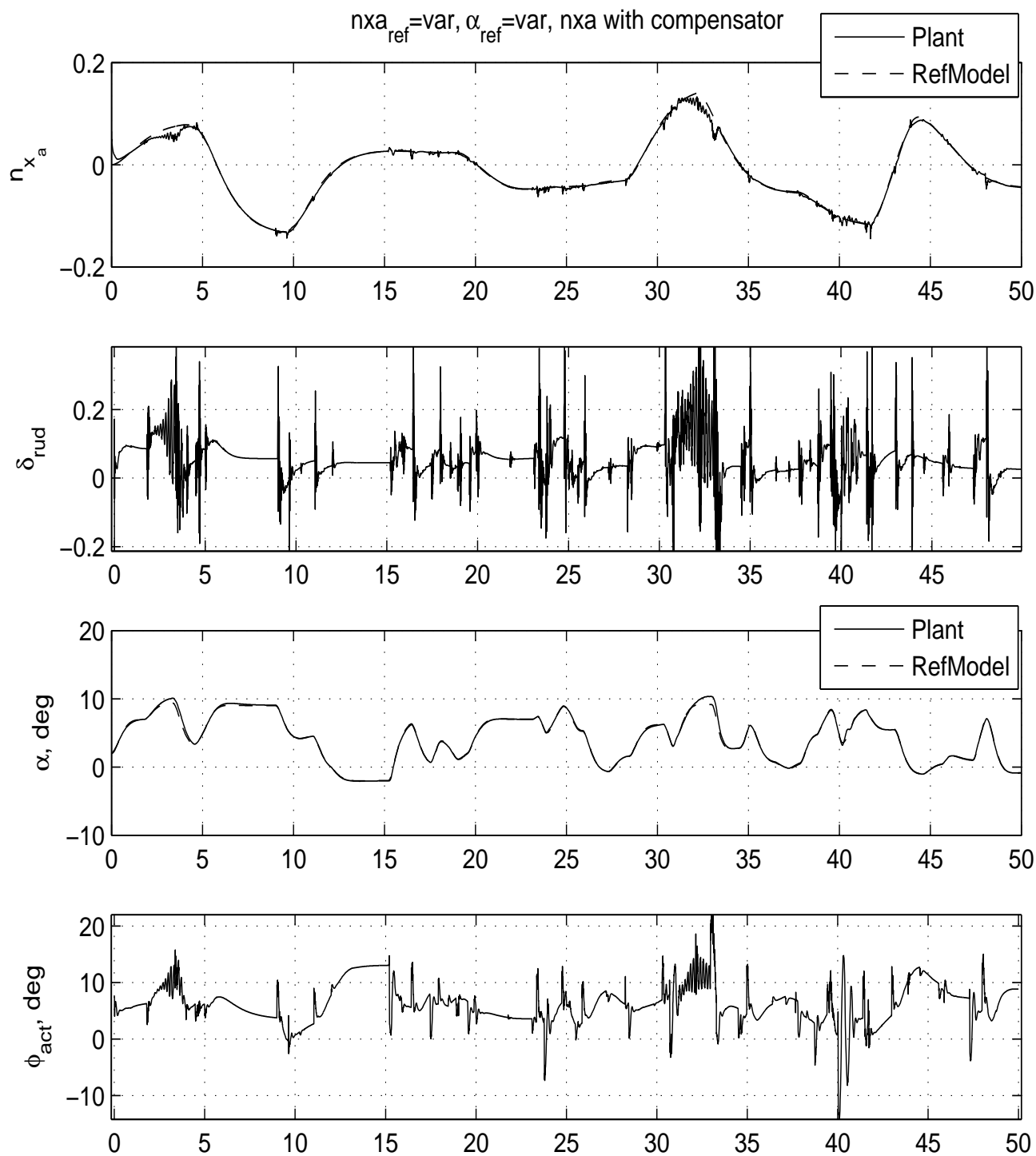


Рис. А.36. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА X-43, режим полета $M = 6$). Оба задающих сигнала — случайные ступенчатые. Компенсаторы в обоих каналах. I — Поведение объекта и эталонной модели. Обозначения: n_{x_a} — тангенциальная перегрузка; δ_{rud} — командный сигнал управления двигателем; α — угол атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

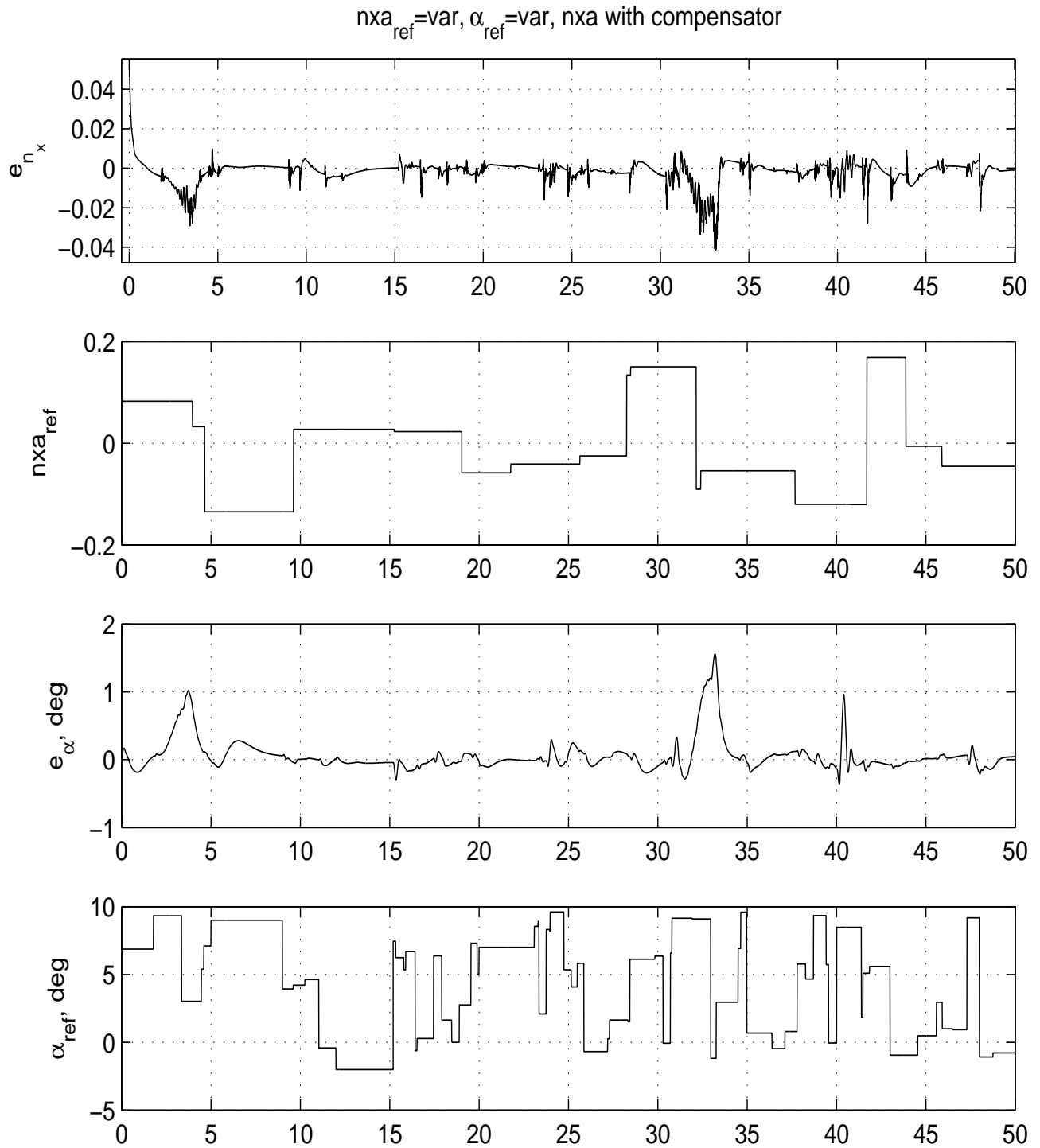


Рис. А.37. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА Х-43, режим полета $M = 6$). Оба задающих сигнала — случайные ступенчатые. Компенсаторы в обоих каналах. II — Задающие сигналы и ошибки отслеживания. Обозначения: $e_{n_{xa}}$ — ошибка по тангенциальной перегрузке; $n_{x_{a,ref}}$ — задающий сигнал по тангенциальной перегрузке; e_{α} — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

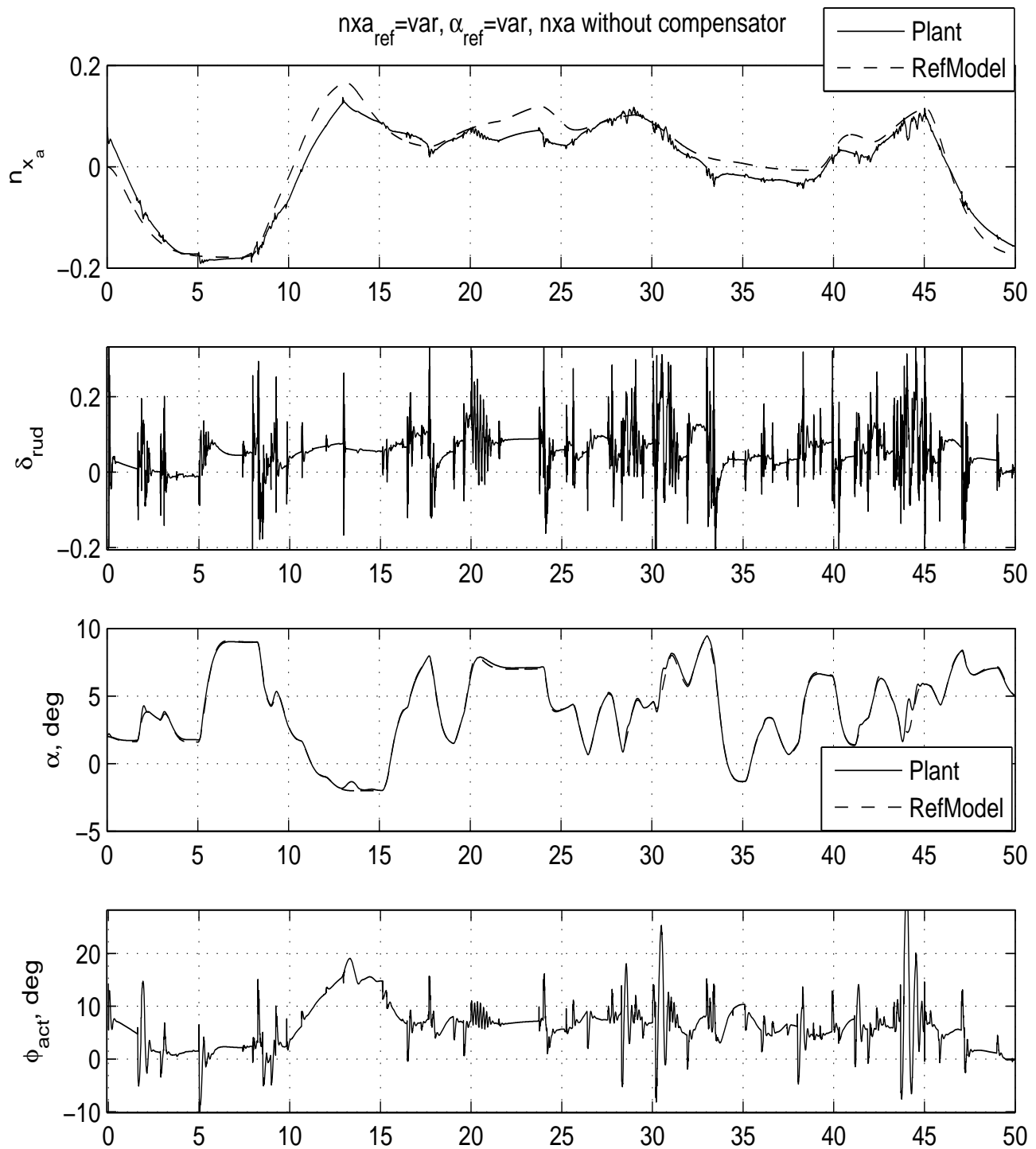


Рис. А.38. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА Х-43, режим полета $M = 6$). Оба задающих сигнала — случайные ступенчатые. Нет компенсатора в канале перегрузки. I — Поведение объекта и эталонной модели. Обозначения: n_{x_a} — тангенциальная перегрузка; δ_{rud} — командный сигнал управления двигателем; α — угол атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

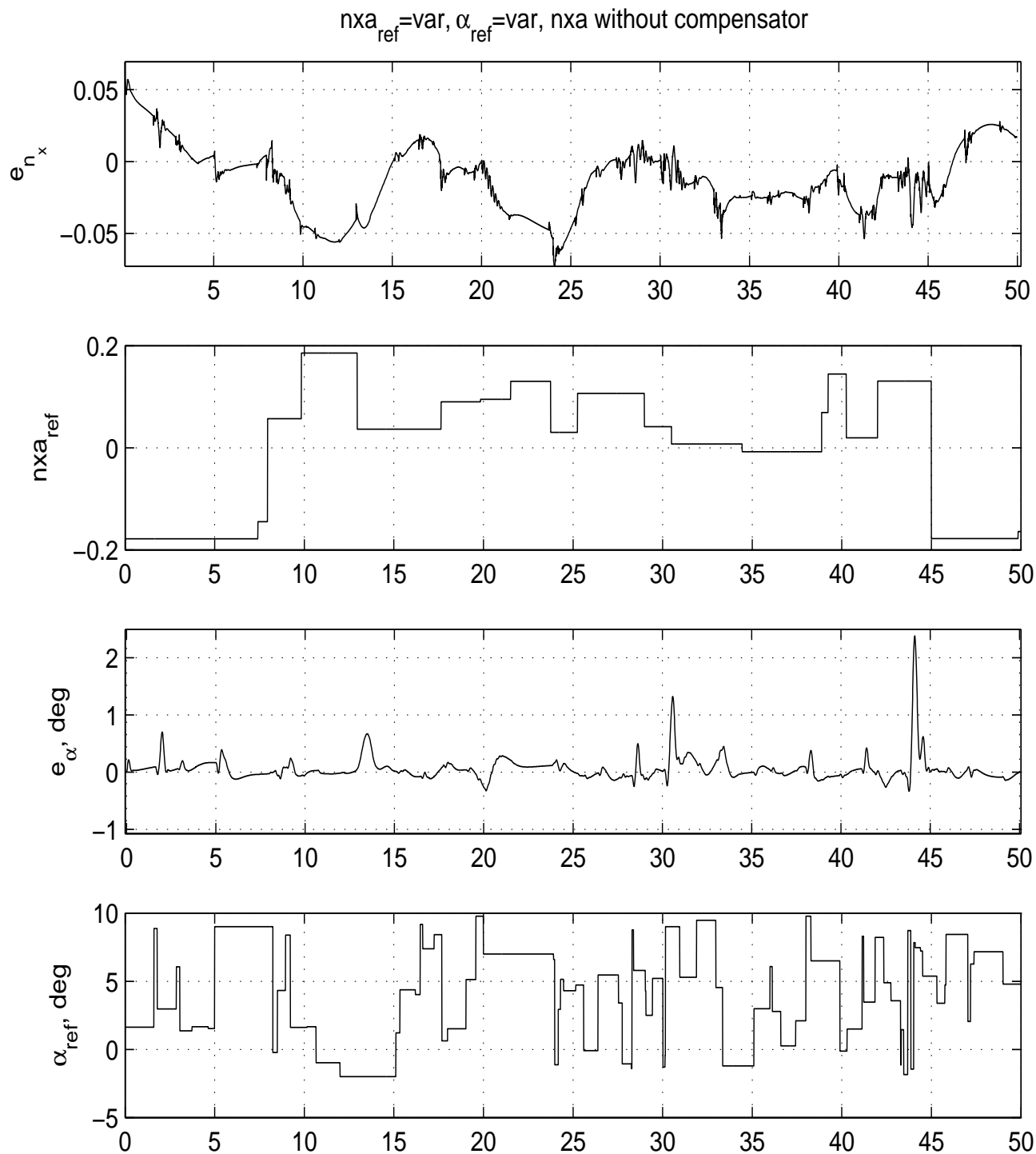


Рис. А.39. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА X-43, режим полета $M = 6$). Оба задающих сигнала — случайные ступенчатые. Нет компенсатора в канале перегрузки. II — Задающие сигналы и ошибки отслеживания. Обозначения: $e_{n_{xa}}$ — ошибка по тангенциальной перегрузке; $n_{x_{a,ref}}$ — задающий сигнал по тангенциальной перегрузке; e_{α} — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

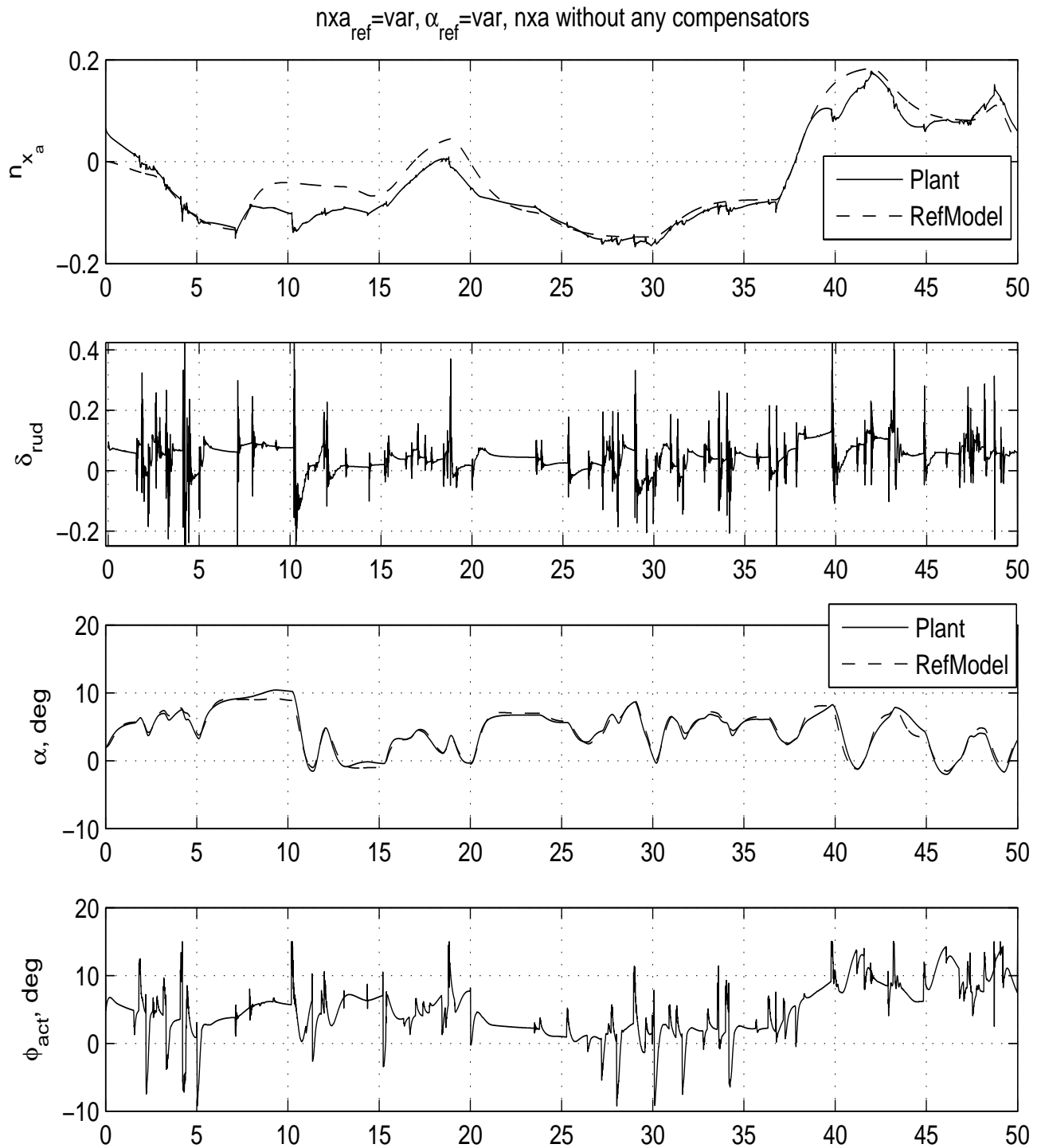


Рис. А.40. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА Х-43, режим полета $M = 6$). Оба задающих сигнала — случайные ступенчатые. В обоих каналах отсутствуют компенсаторы. I — Поведение объекта и эталонной модели. Обозначения: n_{x_a} — тангенциальная перегрузка; δ_{rud} — командный сигнал управления двигателем; α — угол атаки, град.; ϕ_{act} — командный сигнал для привода элементов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

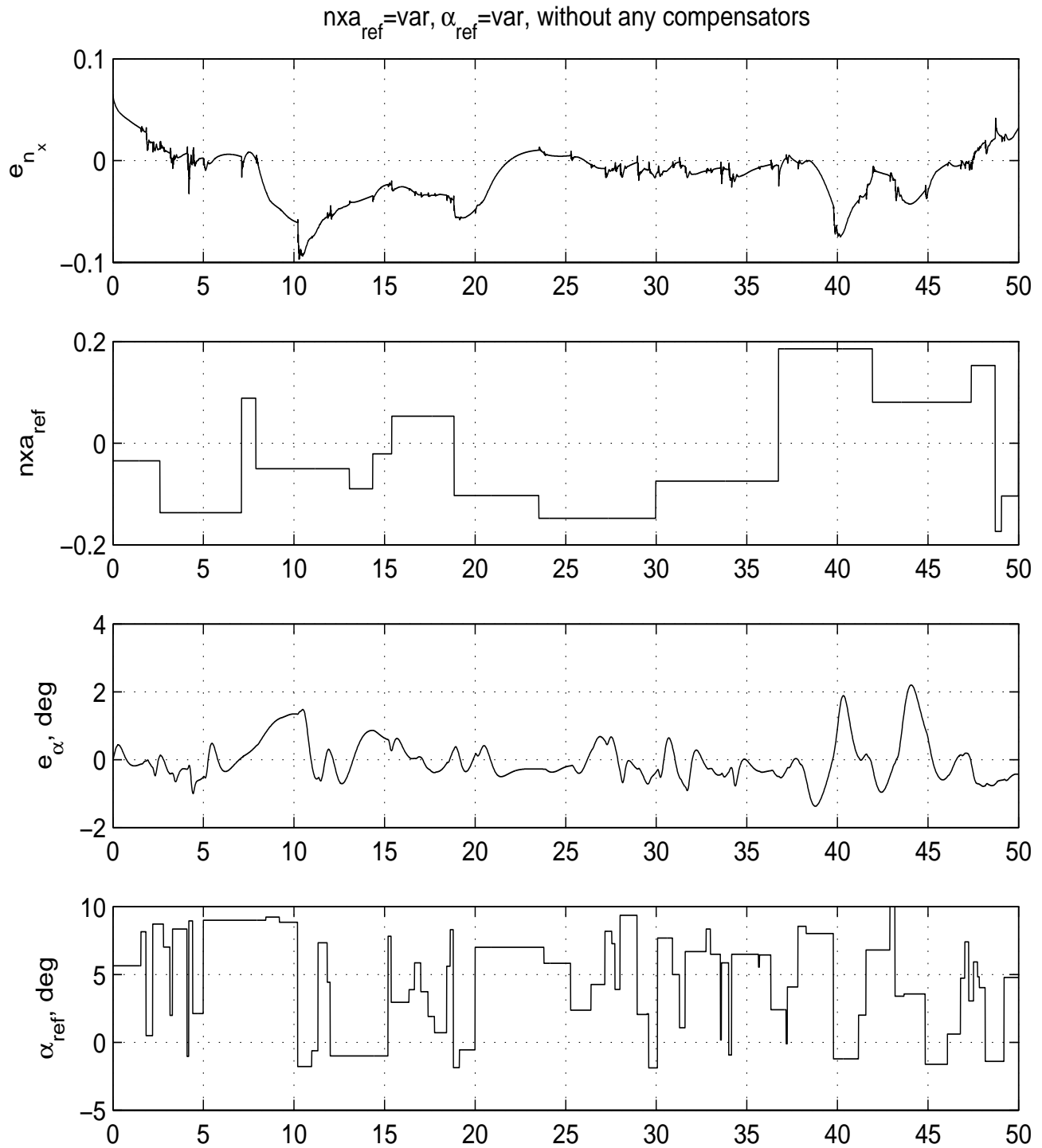


Рис. А.41. Результаты вычислительного эксперимента для системы управления с эталонной моделью (ГЗЛА Х-43, режим полета $M = 6$). Оба задающих сигнала — случайные ступенчатые. В обоих каналах отсутствуют компенсаторы. II — Задающие сигналы и ошибки отслеживания. Обозначения: $e_{n_{x_a}}$ — ошибка по тангенциальной перегрузке; $n_{x_a,ref}$ — задающий сигнал по тангенциальной перегрузке; e_{α} — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

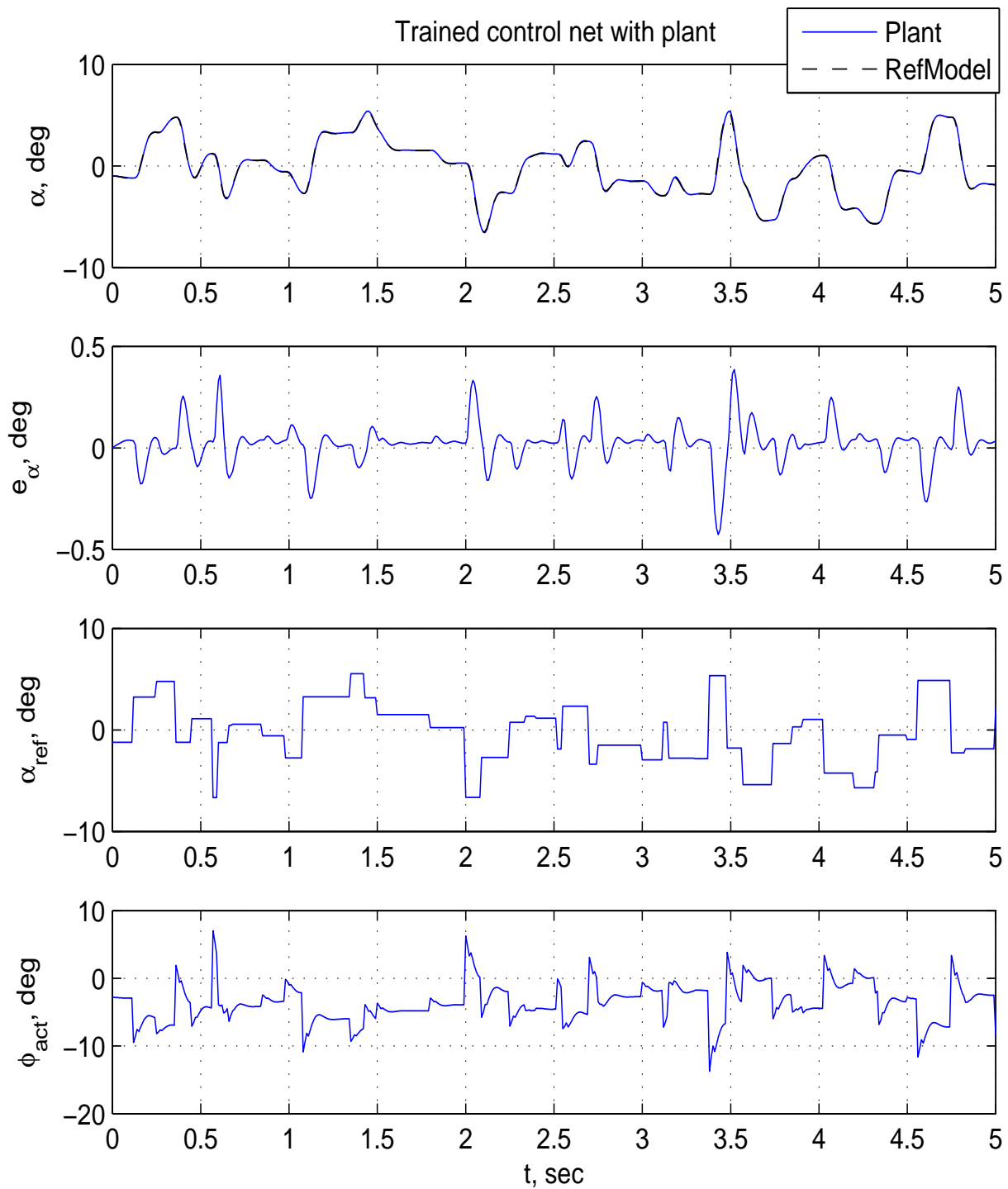


Рис. А.42. Результаты вычислительного эксперимента для системы управления с эталонной моделью (микро-БПЛА «003»), полет с индикаторной скоростью $V_{ind} = 30$ км/ч в штатных условиях функционирования. Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Object — объект управления; RefModel — эталонная модель

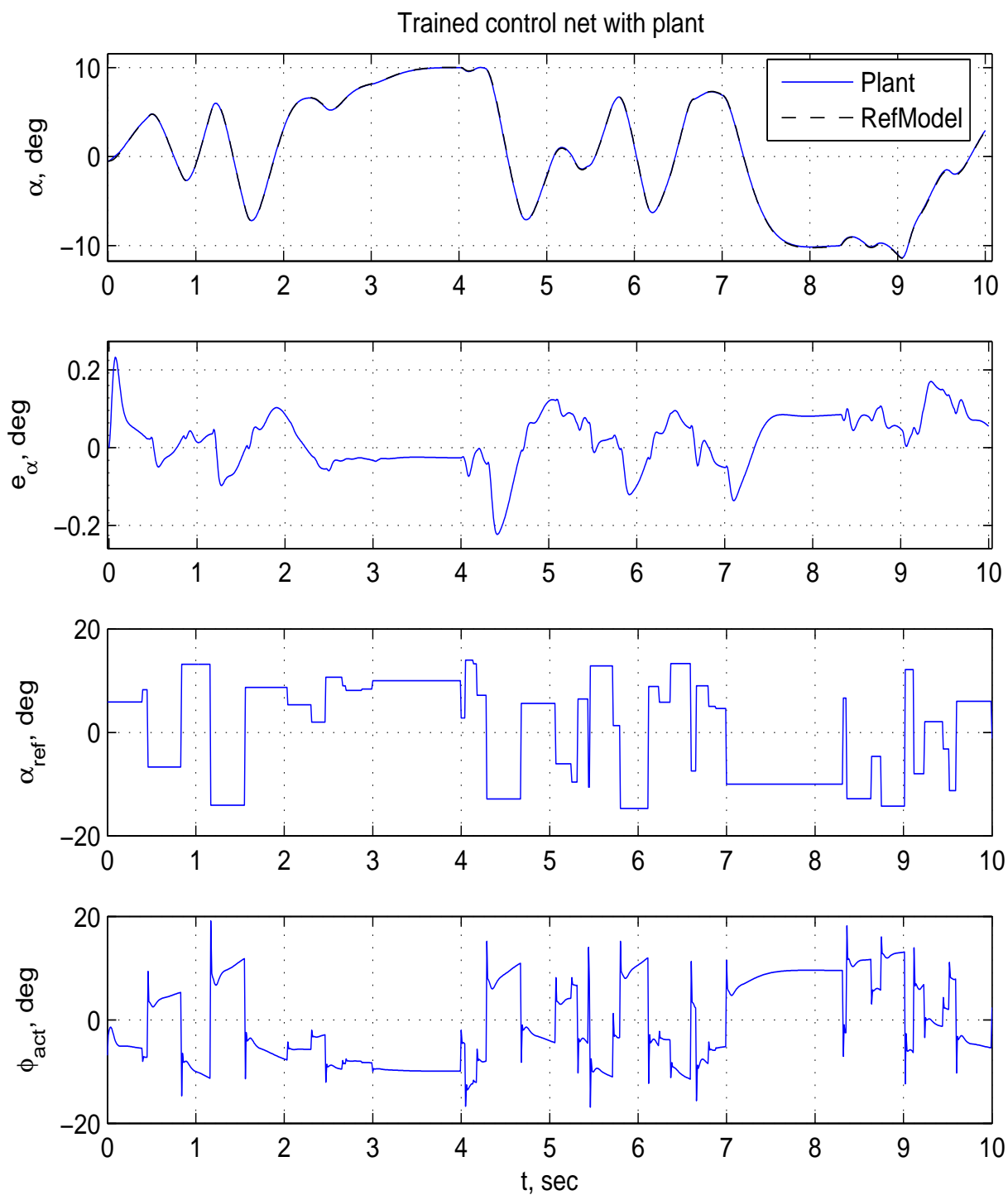


Рис. А.43. Результаты вычислительного эксперимента для системы управления с эталонной моделью (мини-БПЛА X-04), полет с индикаторной скоростью $V_{ind} = 70$ км/ч в штатных условиях функционирования. Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Object — объект управления; RefModel — эталонная модель

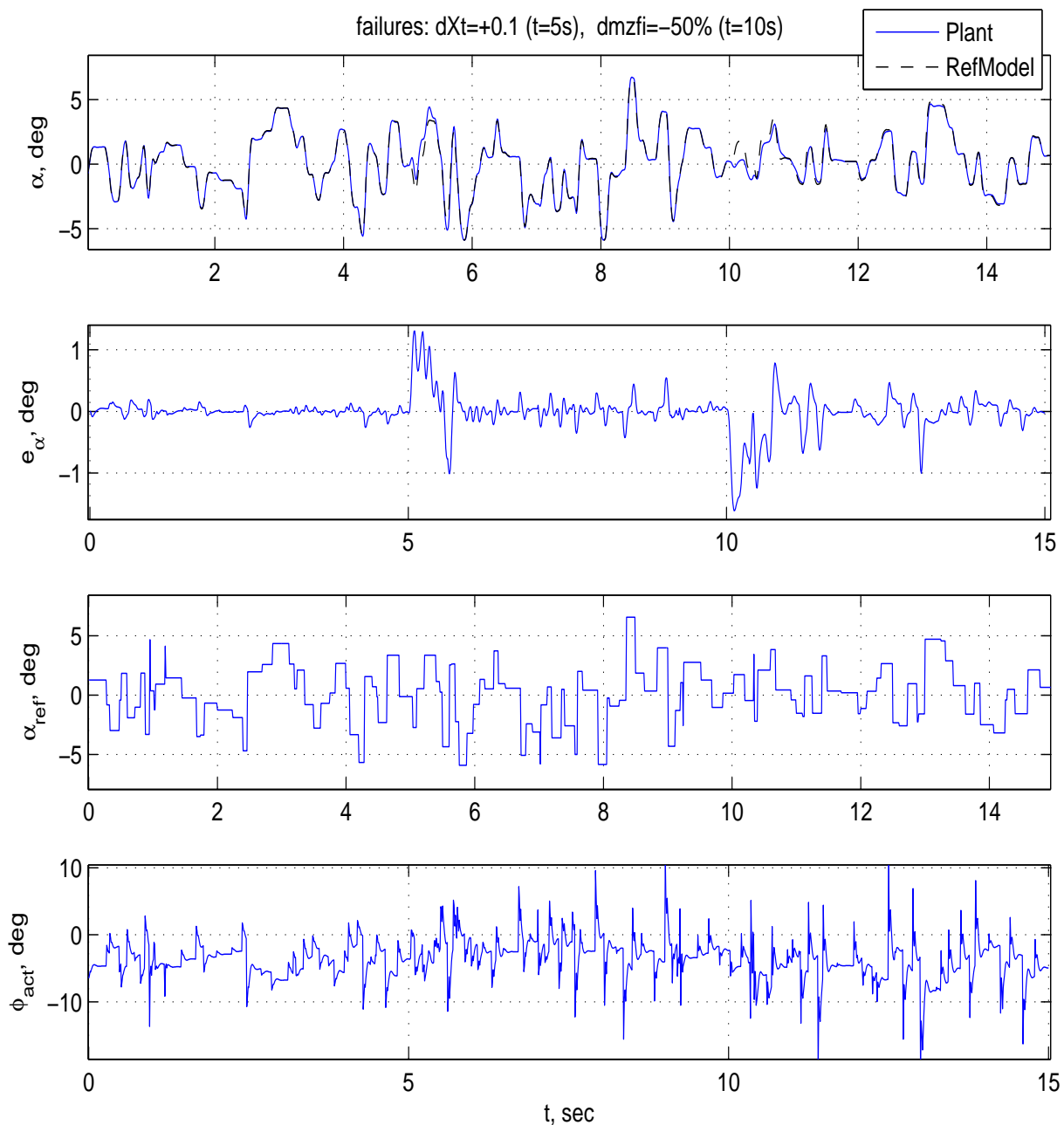


Рис. А.44. Результаты вычислительного эксперимента для системы управления с эталонной моделью (микро-БПЛА «003»), полет с индикаторной скоростью $V_{ind} = 30$ км/ч при возникновении отказных ситуаций в момент времени $t = 5$ с (смещение центровки на 10% назад) и затем в момент времени $t = 10$ с (уменьшение на 50% эффективности органа продольного управления). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Object — объект управления; RefModel — эталонная модель

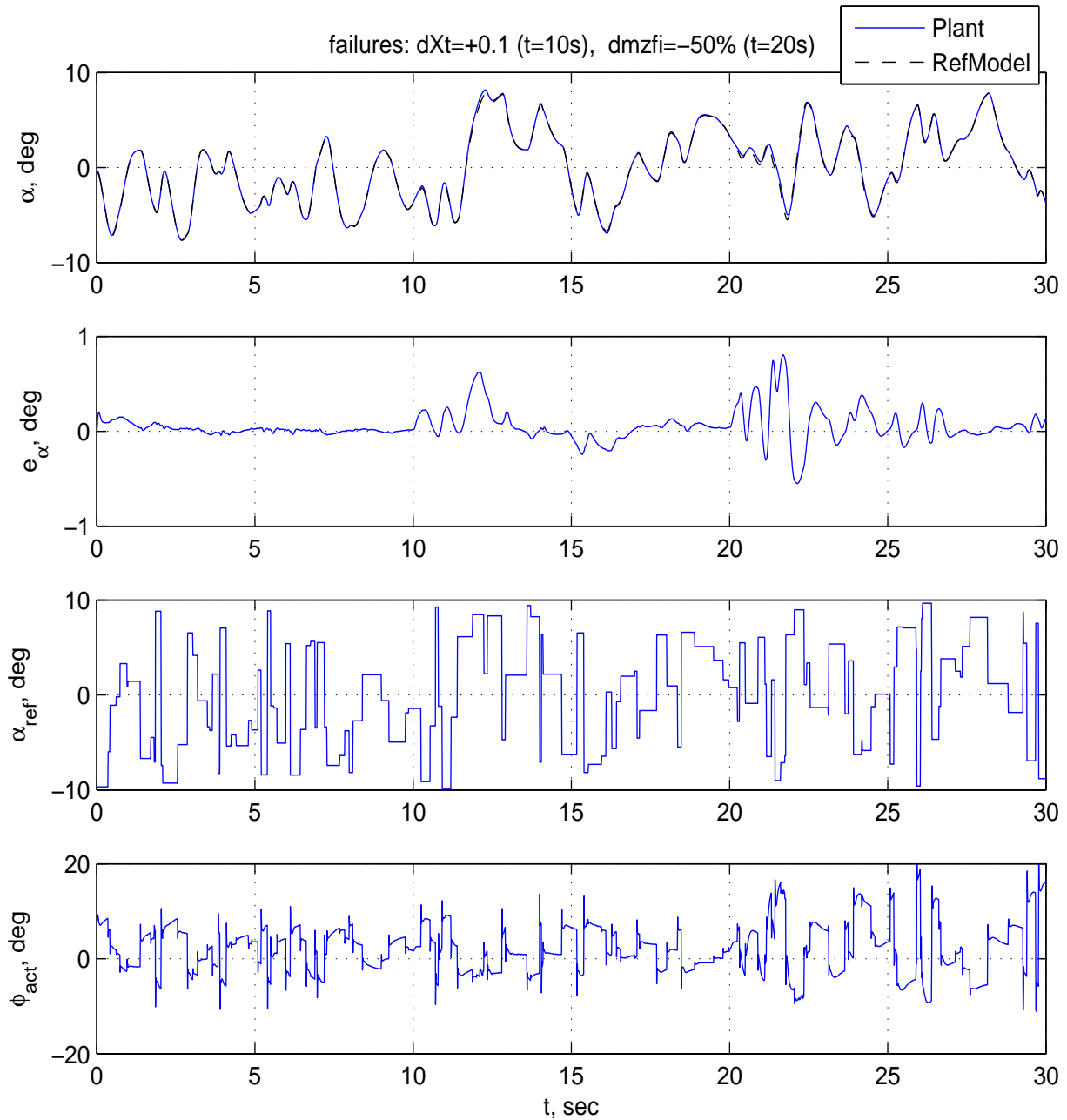


Рис. А.45. Результаты вычислительного эксперимента для системы управления с эталонной моделью (мини-БПЛА X-04), полет с индикаторной скоростью $V_{ind} = 70$ км/ч при возникновении отказных ситуаций в момент времени $t = 10$ с (смещение центровки на 10% назад) и затем в момент времени $t = 20$ с (уменьшение на 50% эффективности органа продольного управления). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Object — объект управления; RefModel — эталонная модель

А.2. Адаптивное управление с прогнозирующей моделью (ПМ)

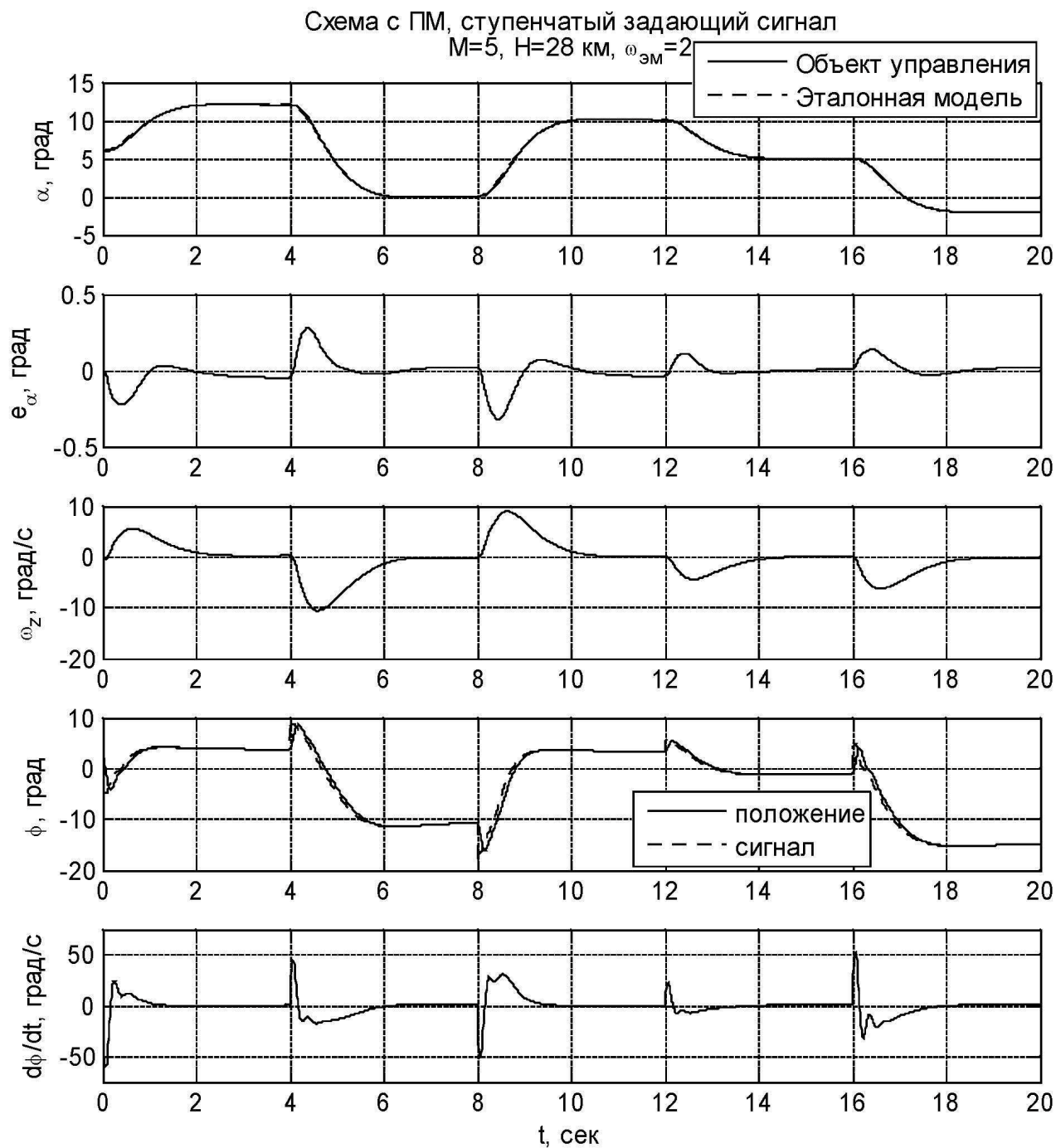


Рис. А.46. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА при ступенчатом задающем сигнале по углу атаки (режим полета $M = 5, H = 28 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град;; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

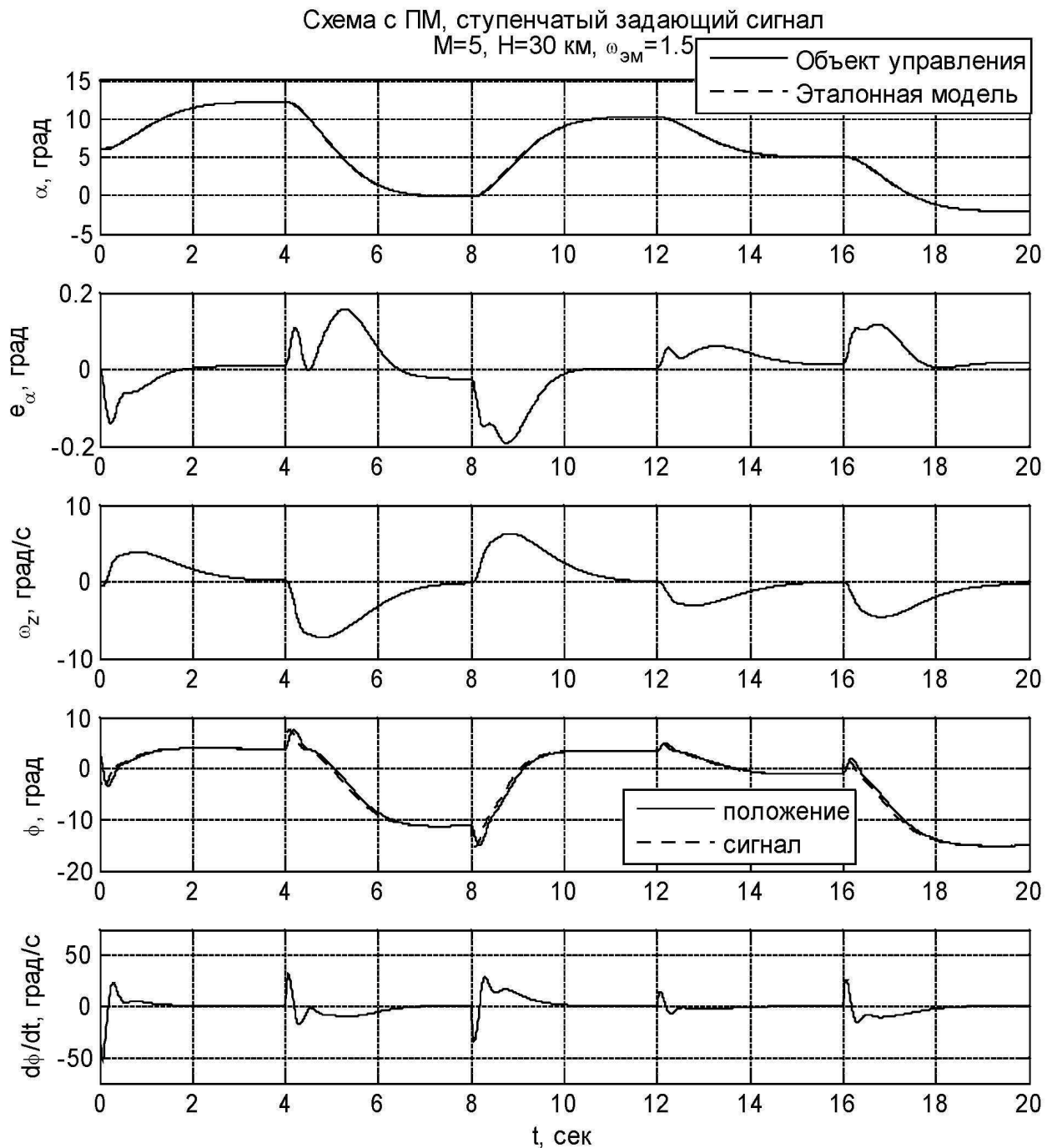


Рис. А.47. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА при ступенчатом задающем сигнале по углу атаки (режим полета $M = 5, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град;; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

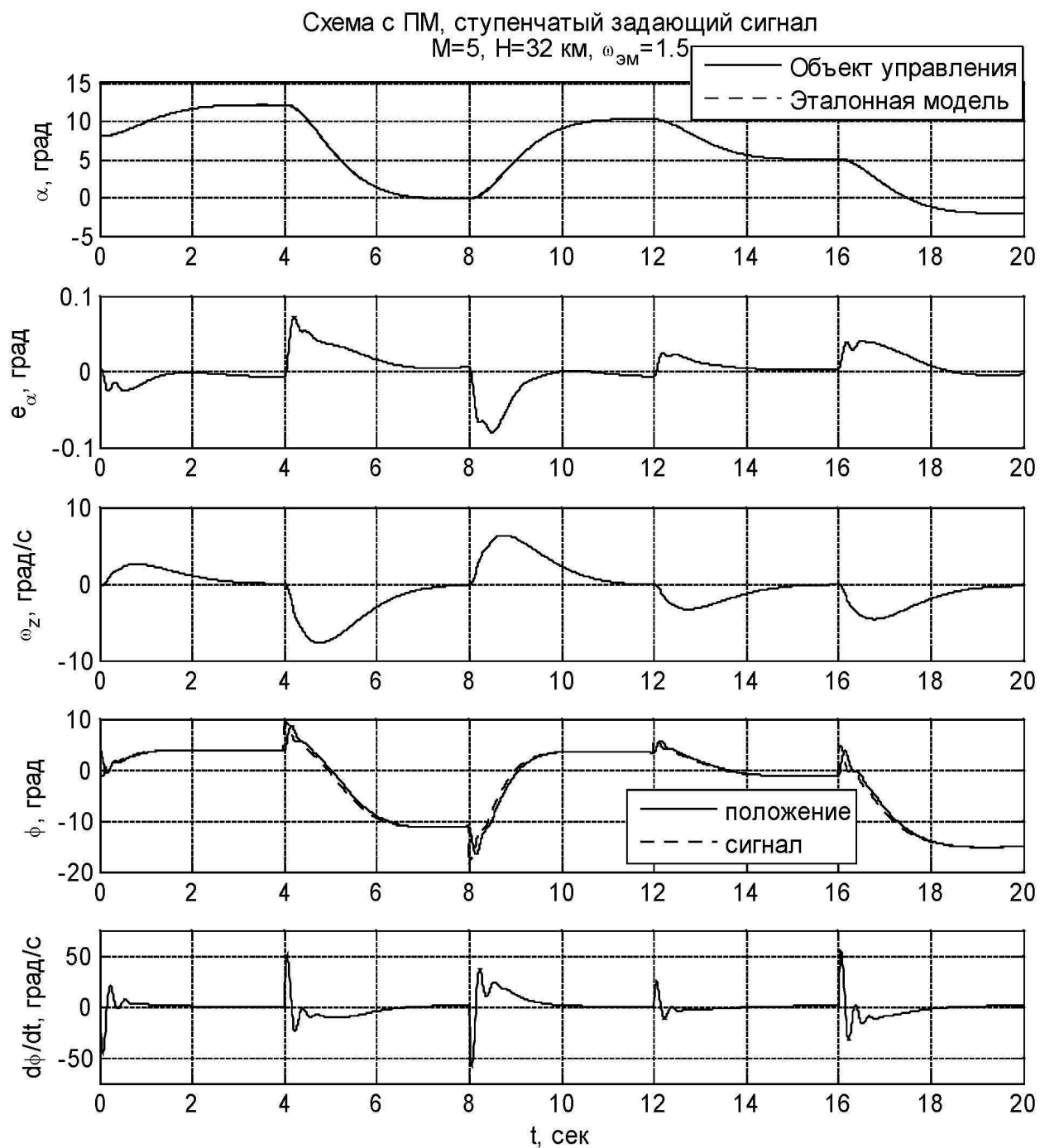


Рис. А.48. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА при ступенчатом задающем сигнале по углу атаки (режим полета $M = 5, H = 32 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град;; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

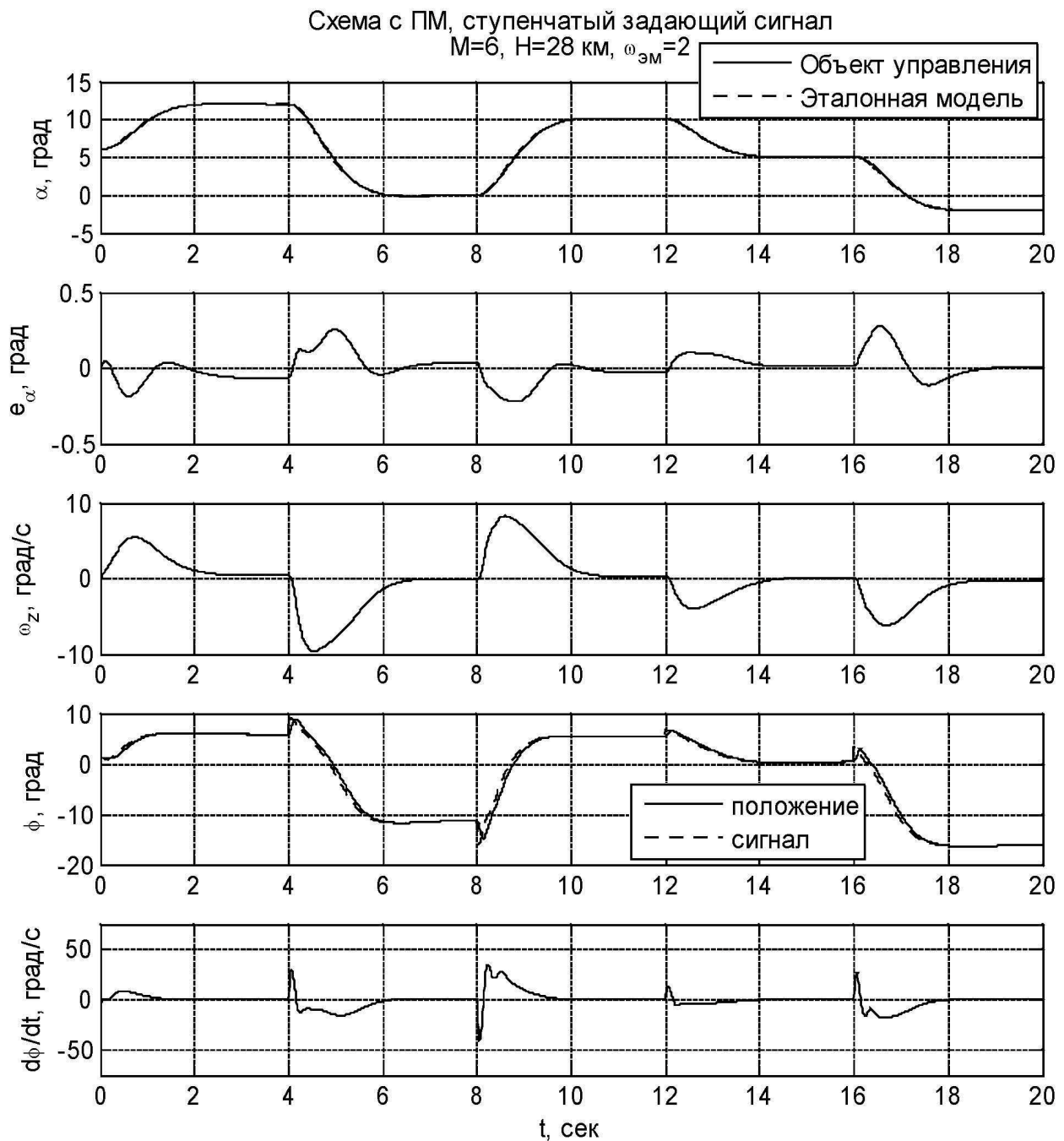


Рис. А.49. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА при ступенчатом задающем сигнале по углу атаки (режим полета $M = 6, H = 28 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град;; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

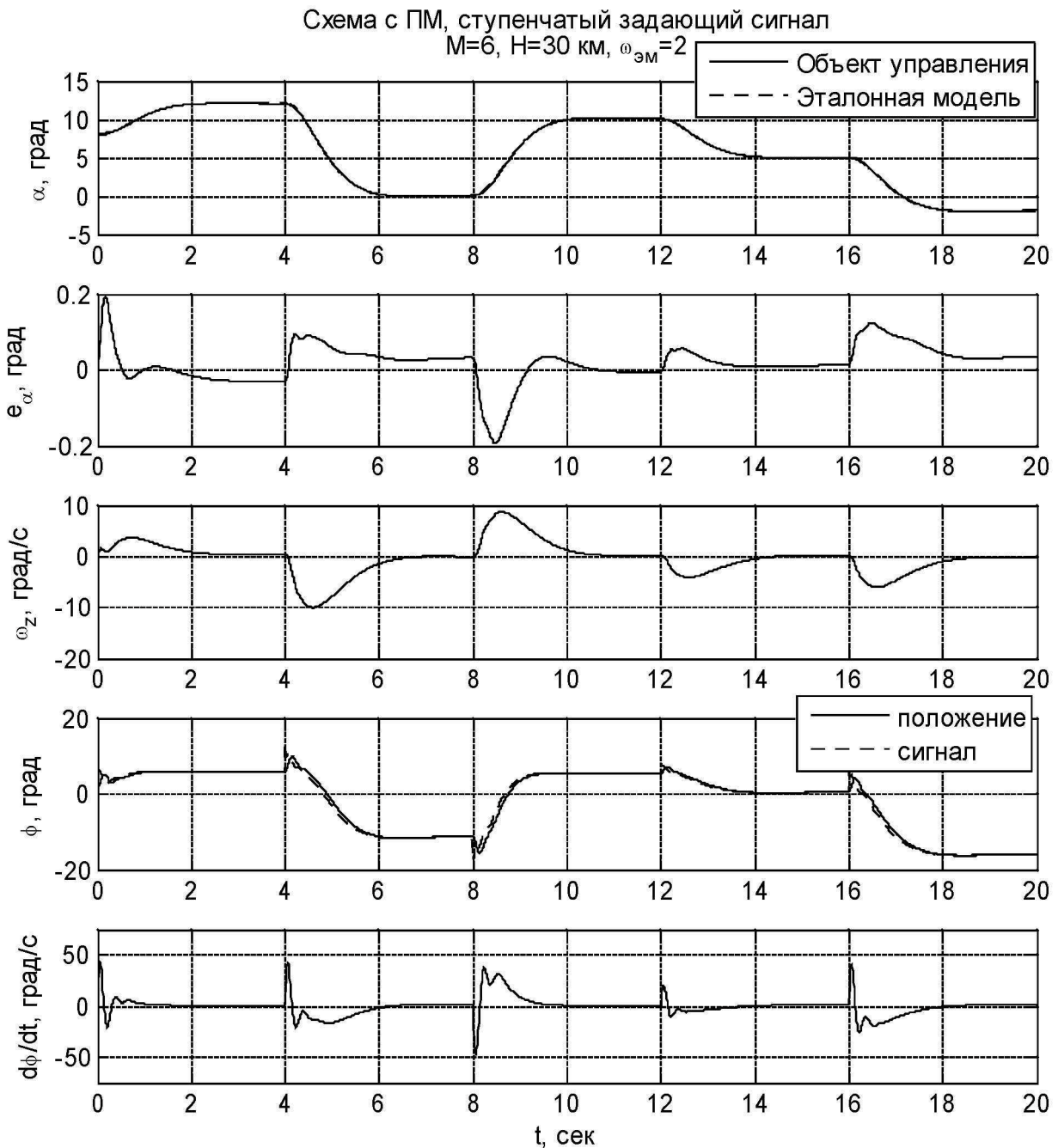


Рис. А.50. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА при ступенчатом задающем сигнале по углу атаки (режим полета $M = 6, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град;; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

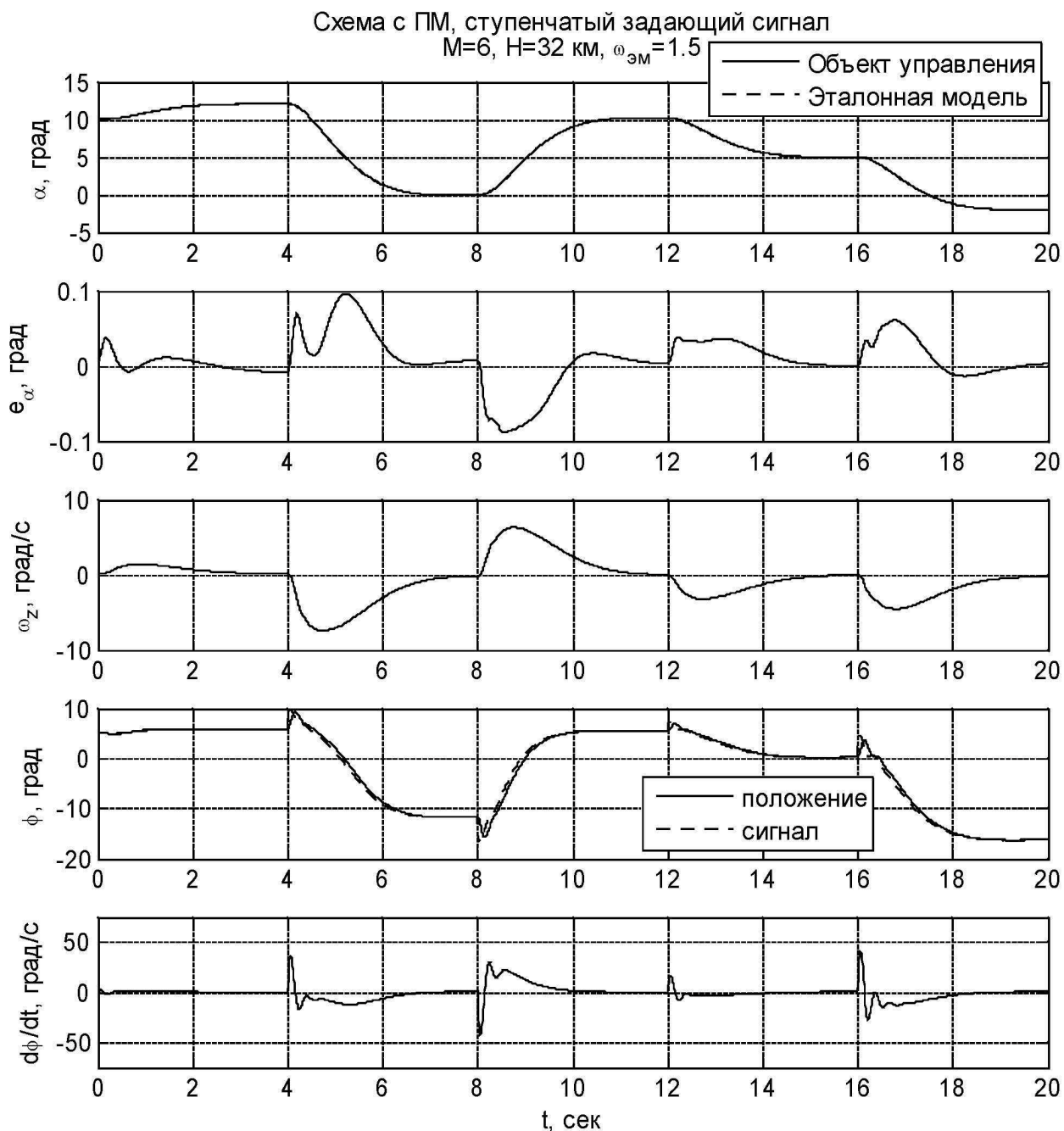


Рис. А.51. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА при ступенчатом задающем сигнале по углу атаки (режим полета $M = 6, H = 32 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град;; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

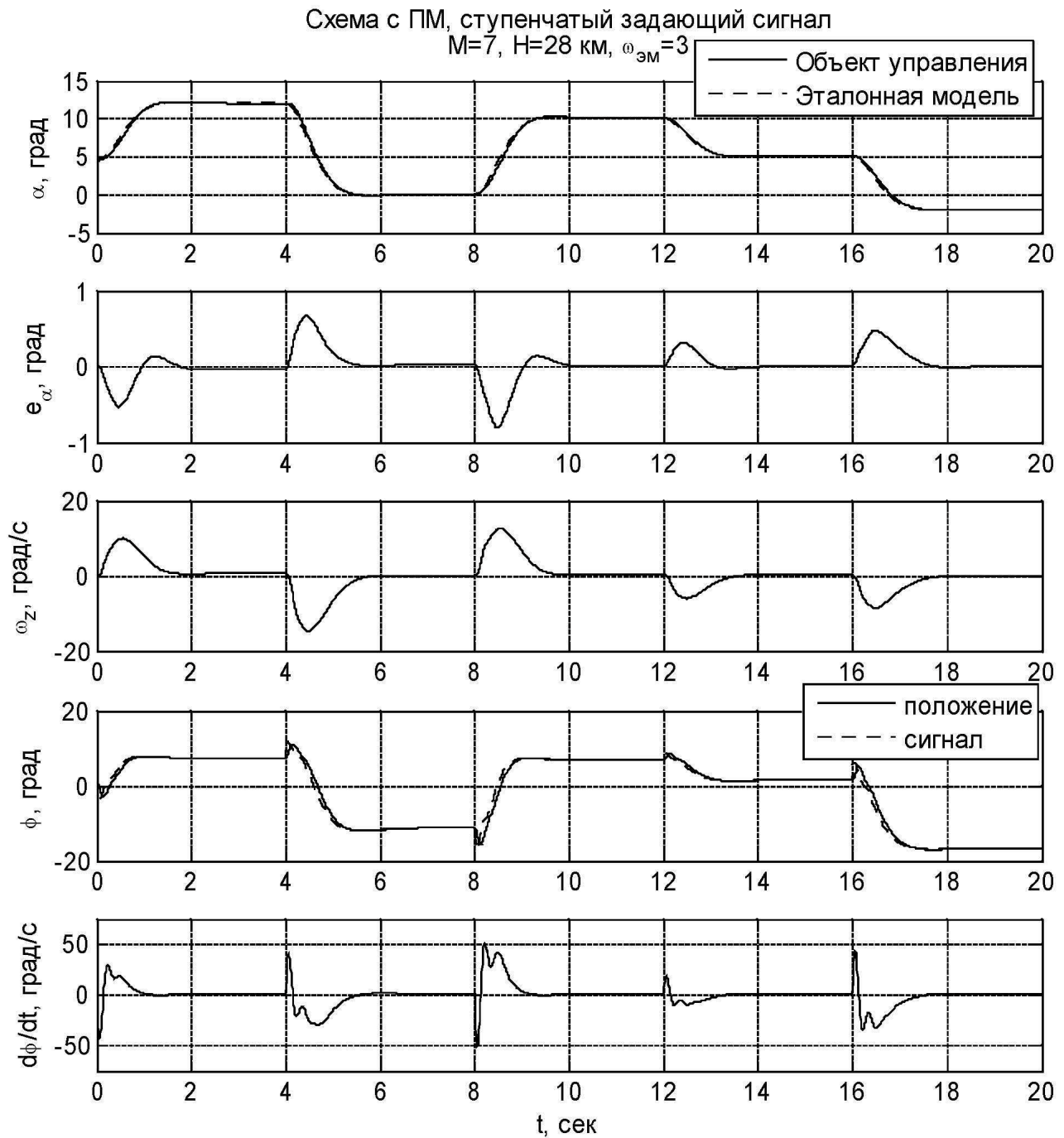


Рис. А.52. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА при ступенчатом задающем сигнале по углу атаки (режим полета $M = 7$, $H = 28$ км). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град;; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

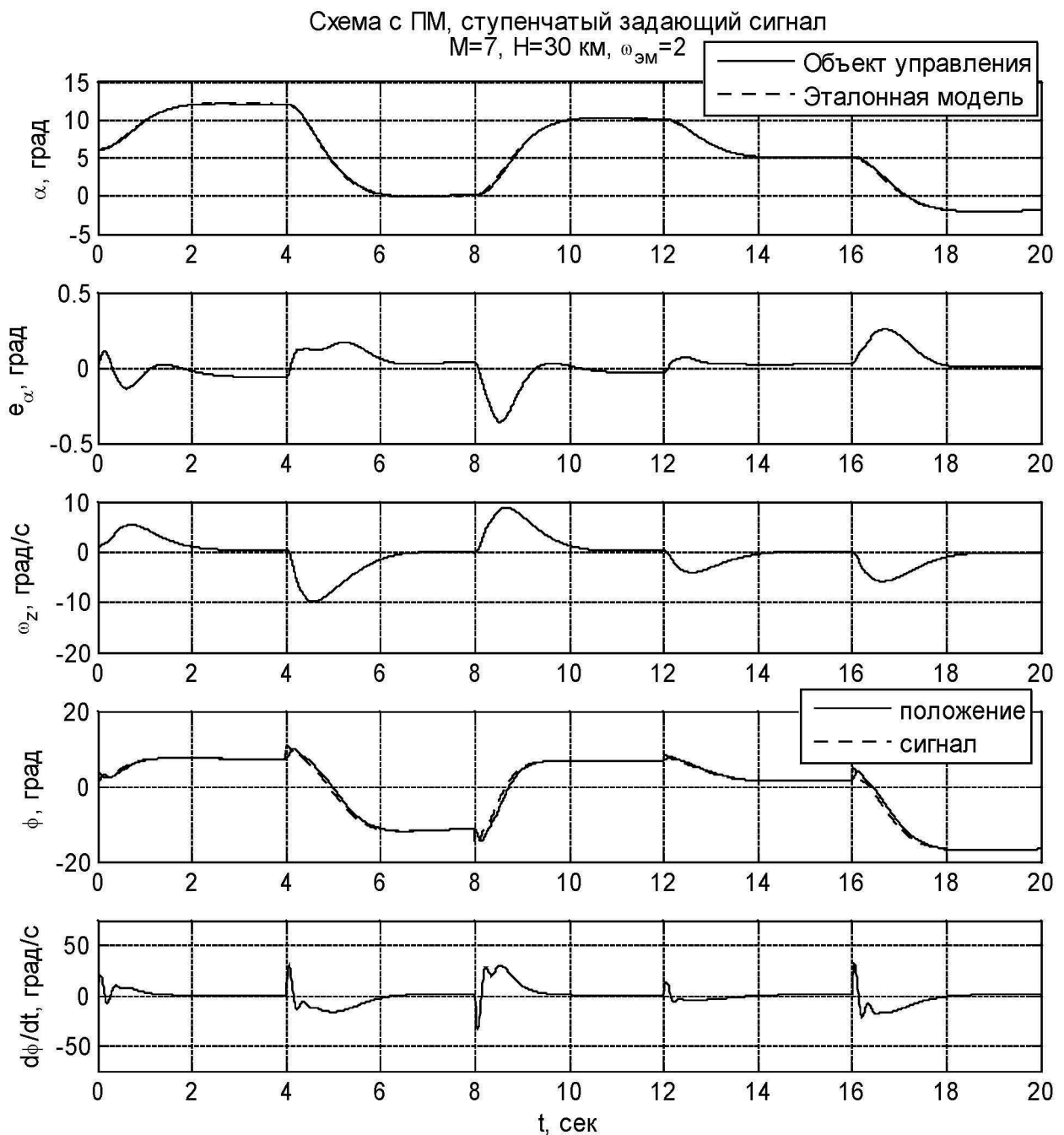


Рис. А.53. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА при ступенчатом задающем сигнале по углу атаки (режим полета $M = 7$, $H = 30$ км). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град;; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

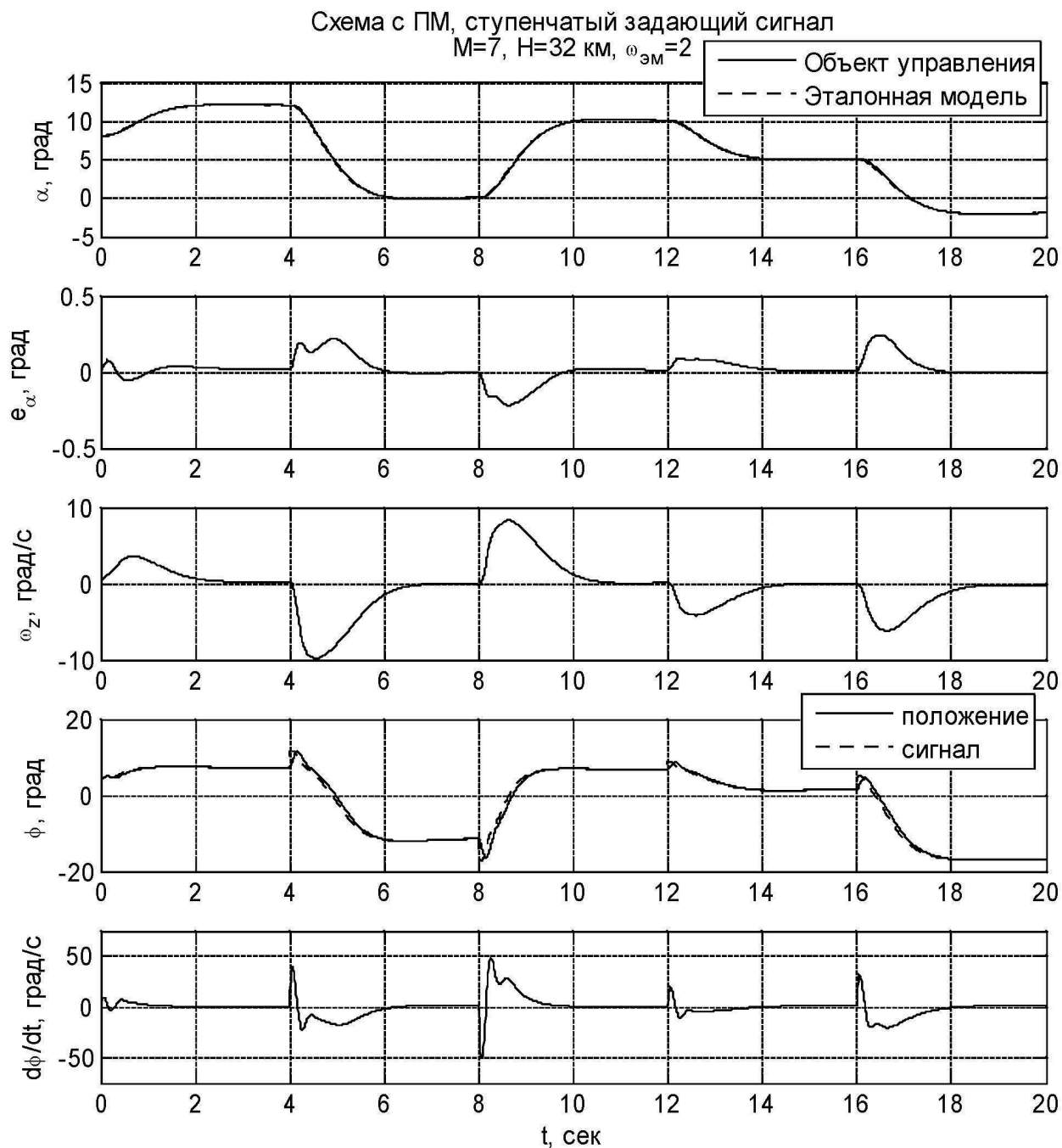


Рис. А.54. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА при ступенчатом задающем сигнале по углу атаки (режим полета $M = 7, H = 32 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; ω_z — угловая скорость тангажа, град/с; ϕ — командный сигнал привода («сигнал») и угол отклонения элевонов («положение»), град;; $d\phi/dt$ — угловая скорость отклонения элевонов, град/с; t — время, с

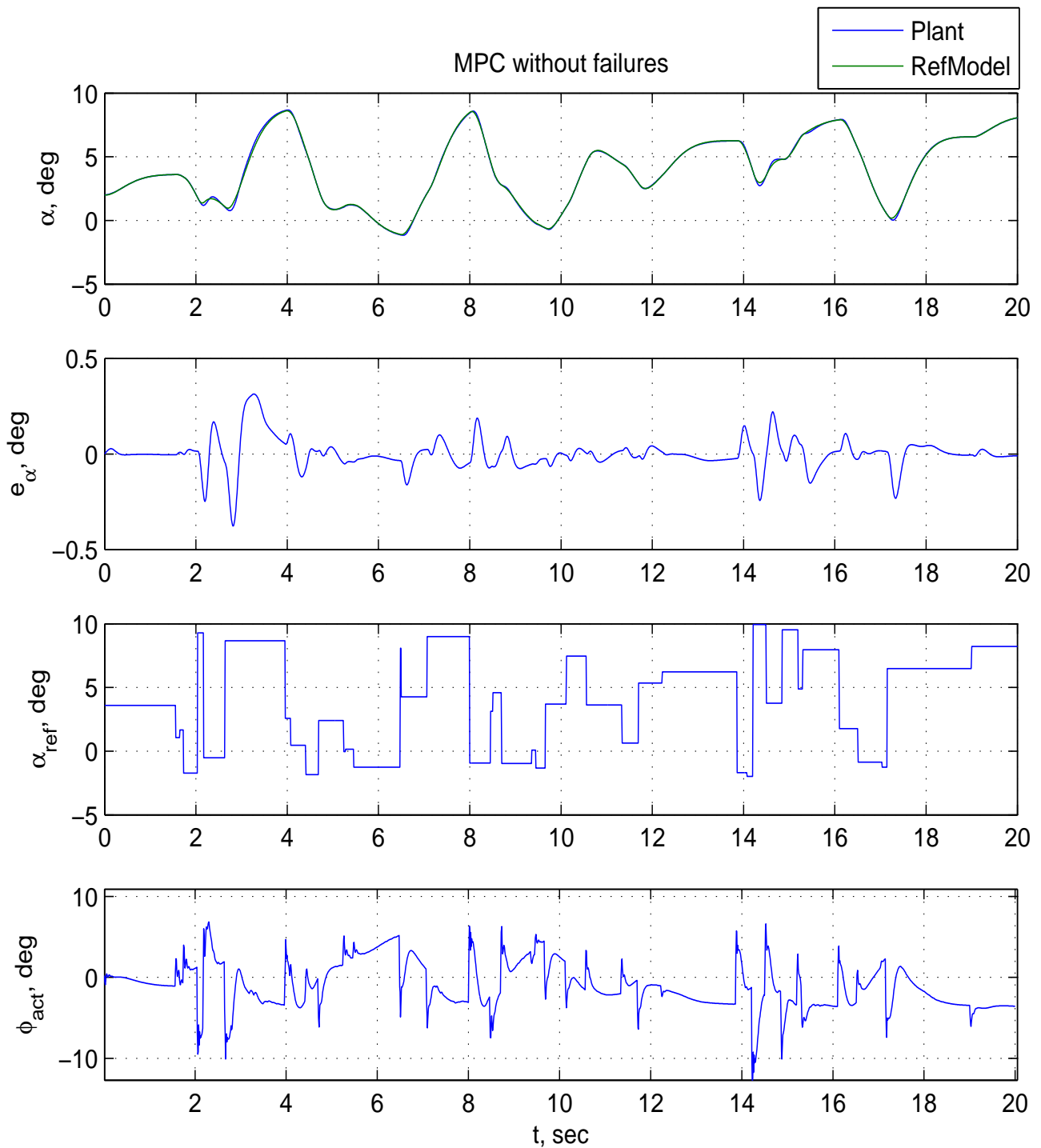


Рис. А.55. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью и компенсатором, без отказов (ГЗЛА Х-43, режим полета с числом Маха $M = 6$). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

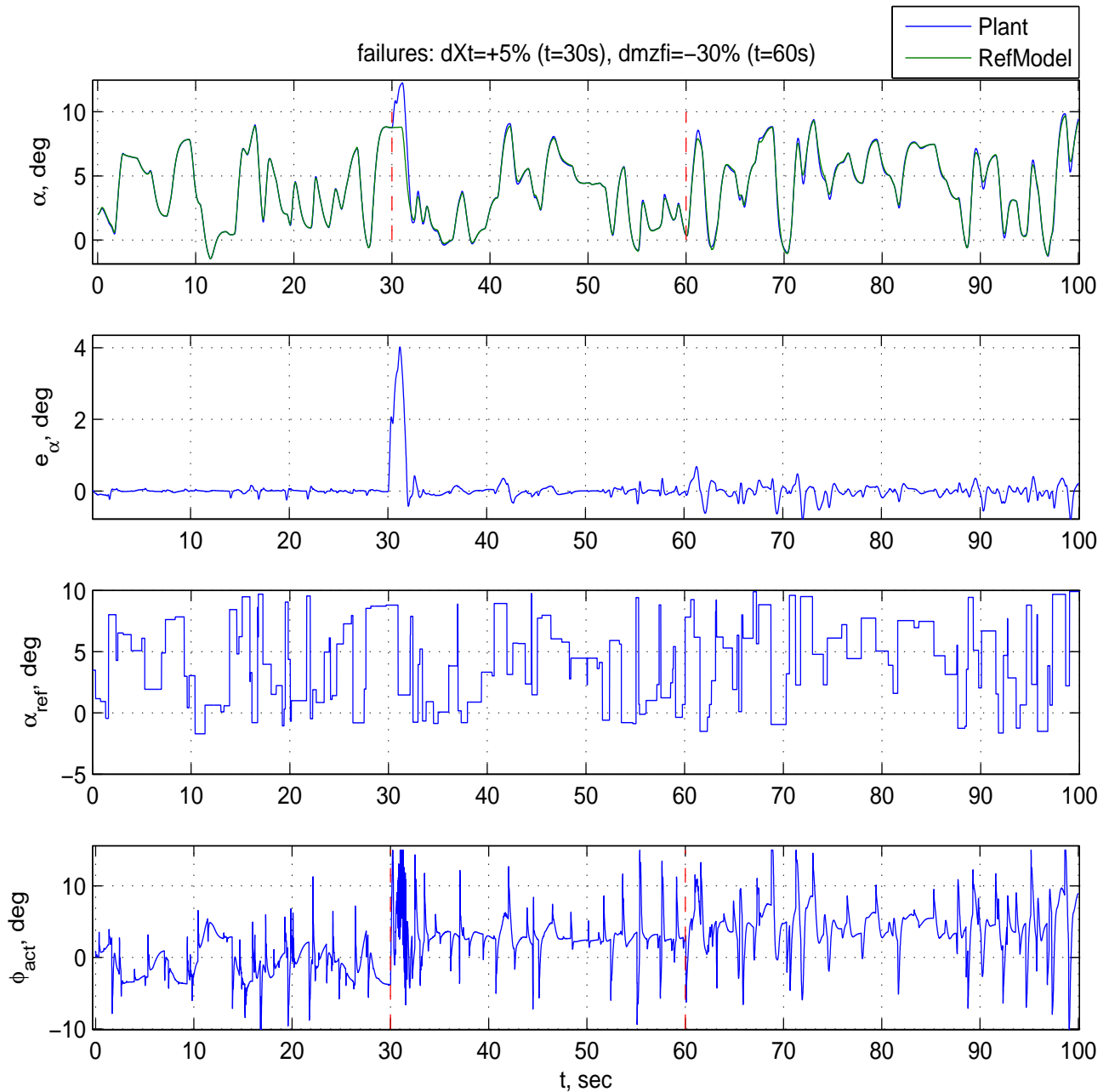


Рис. А.56. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью и компенсатором (ГЗЛА X-43, режим полета с числом Маха $M = 6$). Адаптация к изменению динамики объекта управления вследствие отказов: смещение центровки на 5% назад ($t = 30$ с), 30% уменьшение эффективности органа управления ($t = 60$ с). Обозначения: α — угол атаки, град.; e_{α} — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

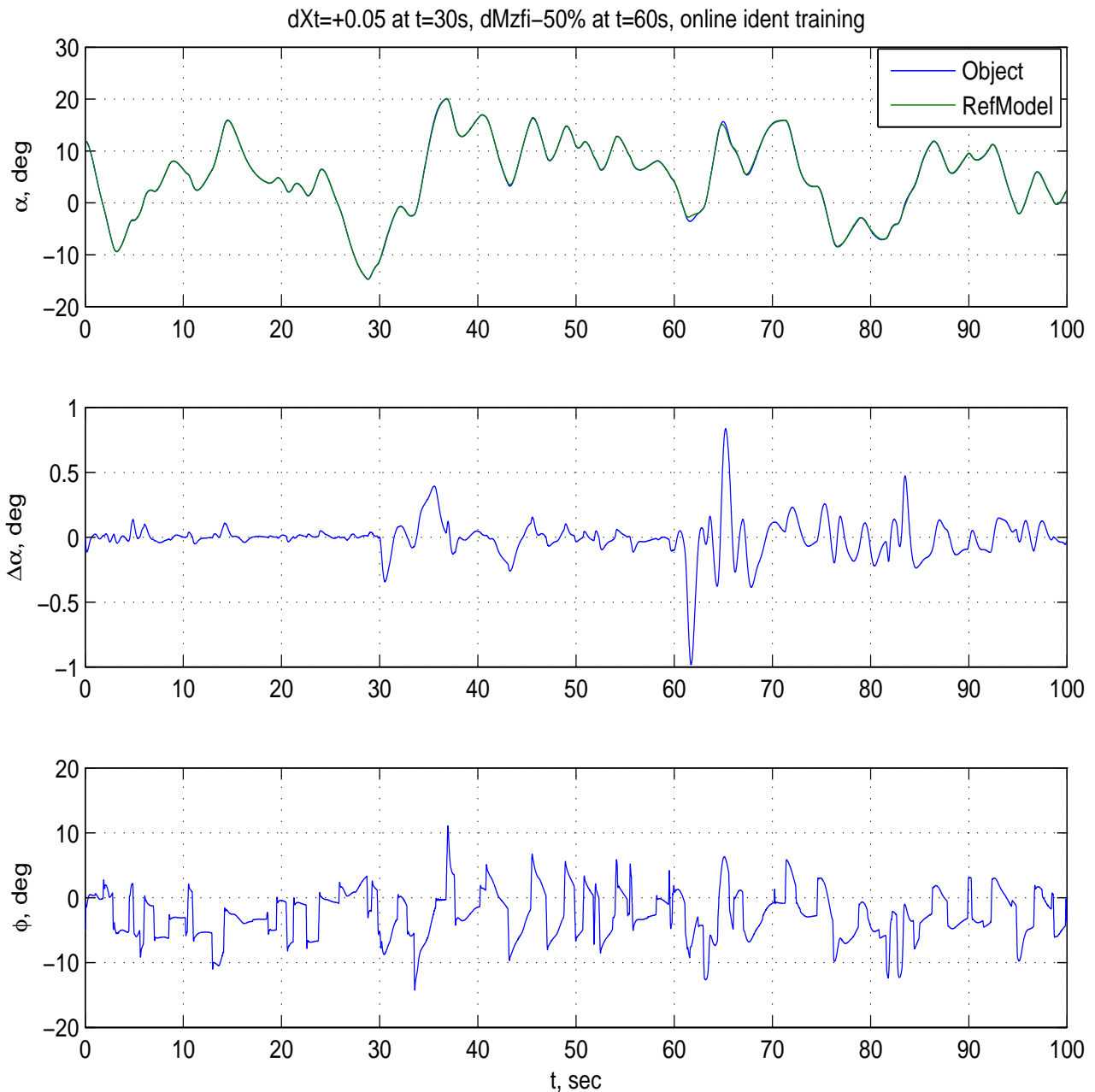


Рис. А.57. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью и компенсатором (самолет F-16, режим полета с индикаторной скоростью $V_{ind} = 300$ км/ч). Адаптация к изменению динамики объекта управления вследствие отказов: смещение центровки на 5% назад ($t = 30$ с), 50% уменьшение эффективности органа управления ($t = 60$ с). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

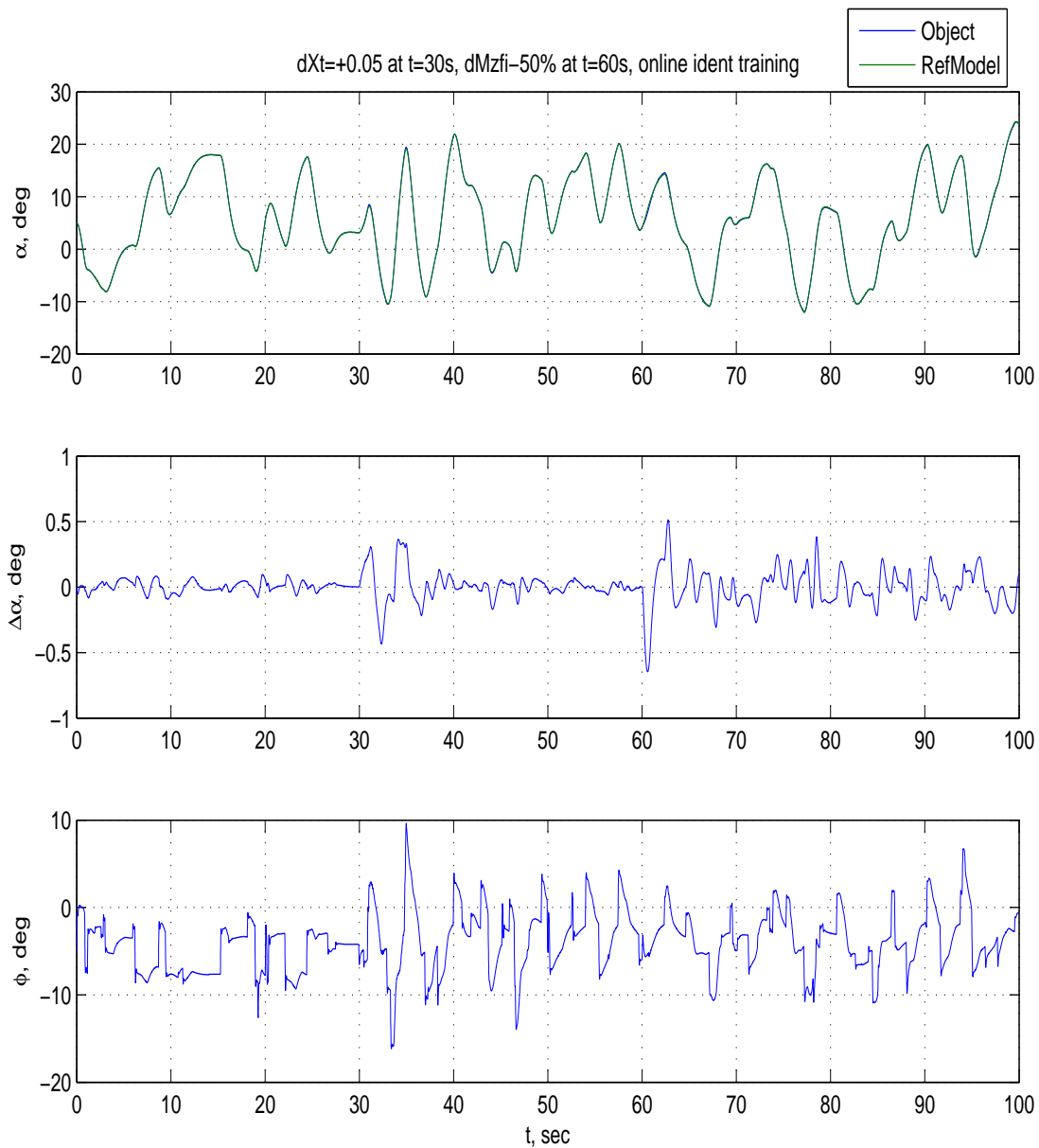


Рис. А.58. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью и компенсатором (самолет F-16, режим полета с индикаторной скоростью $V_{ind} = 500$ км/ч). Адаптация к изменению динамики объекта управления вследствие отказов: смещение центровки на 5% назад ($t = 30$ с), 50% уменьшение эффективности органа управления ($t = 60$ с). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

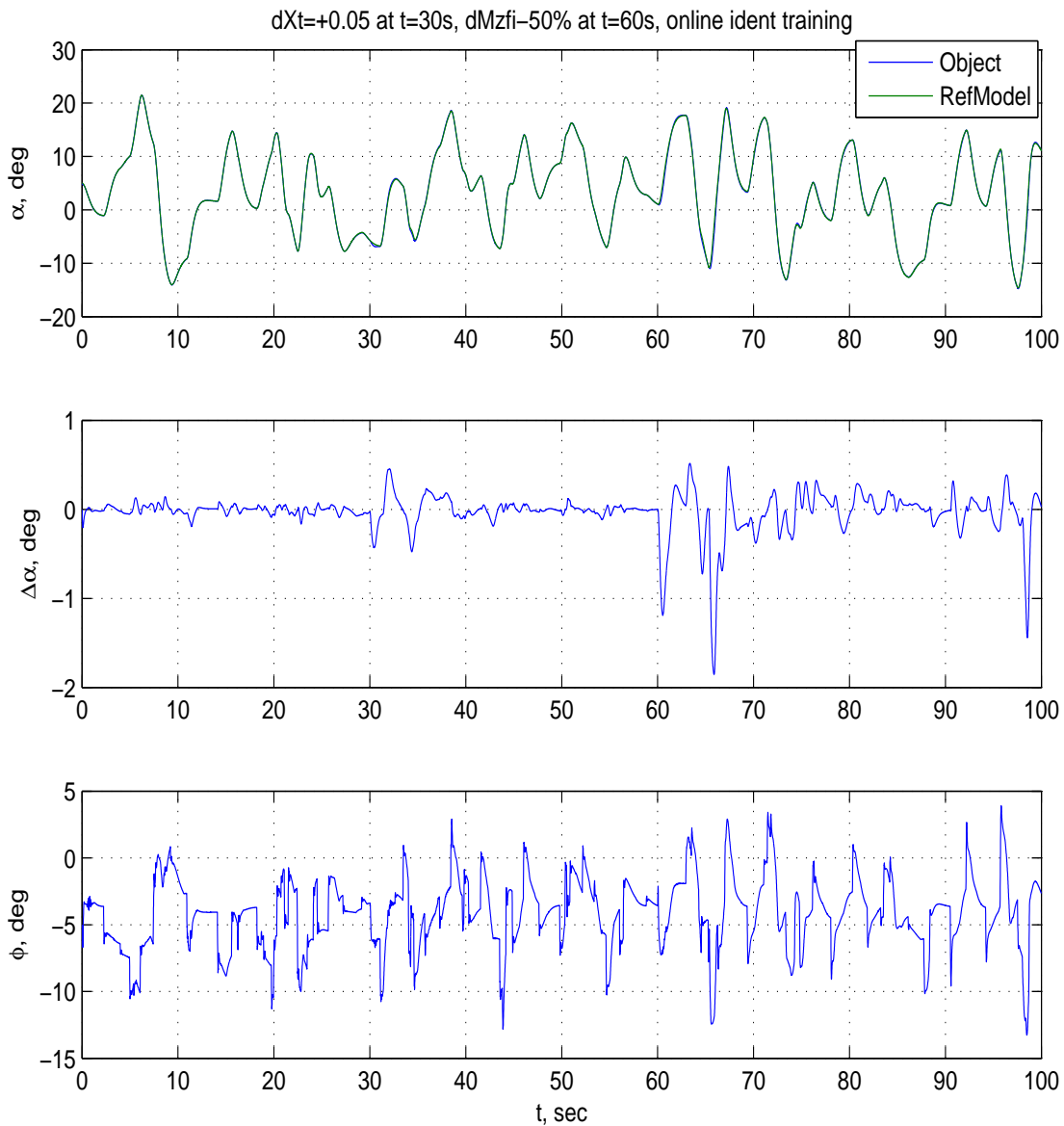


Рис. А.59. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью и компенсатором (самолет F-16, режим полета с индикаторной скоростью $V_{ind} = 700$ км/ч). Адаптация к изменению динамики объекта управления вследствие отказов: смещение центровки на 5% назад ($t = 30$ с), 50% уменьшение эффективности органа управления ($t = 60$ с). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Plant — объект управления; RefModel — эталонная модель

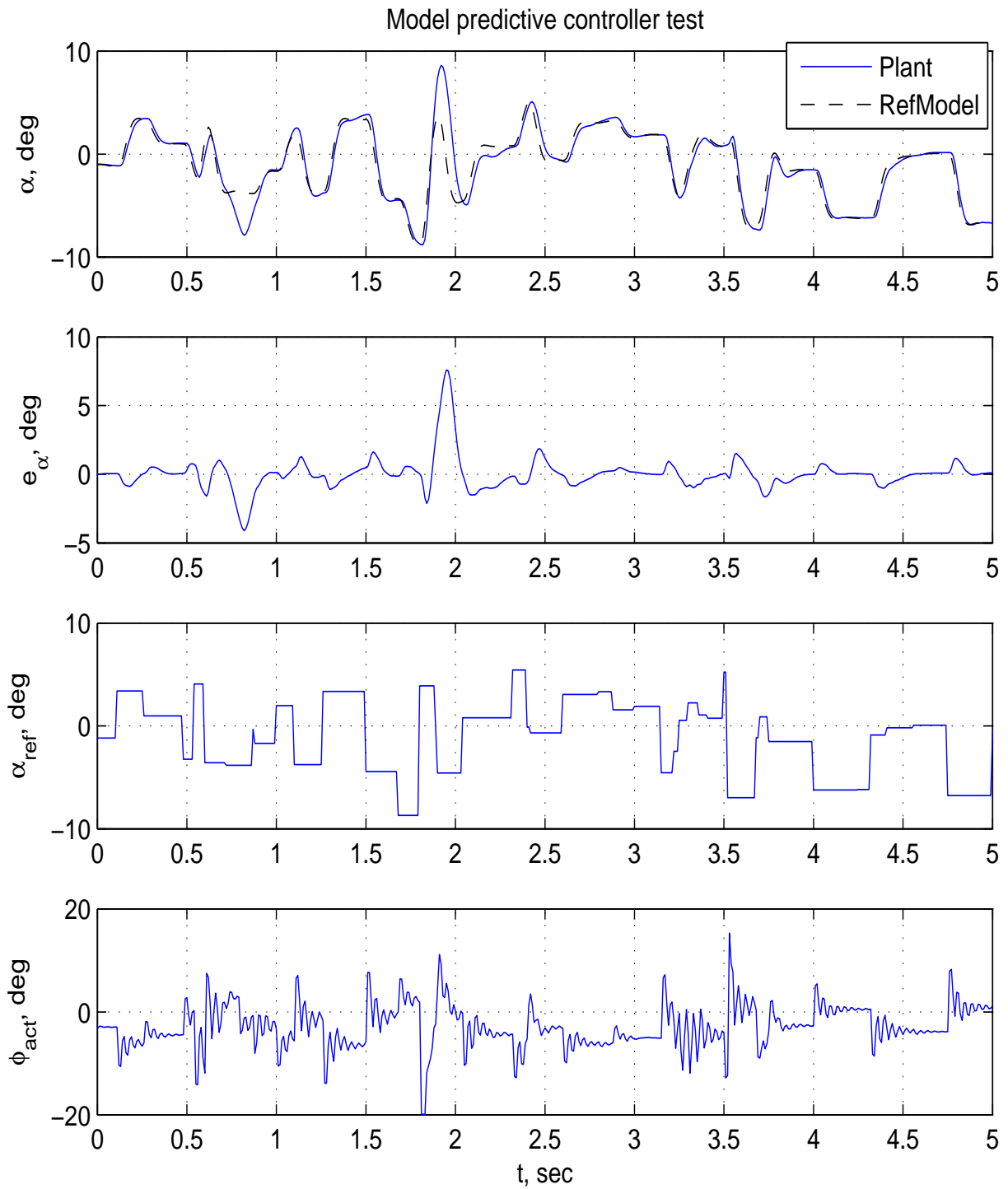


Рис. А.60. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью (микро-БПЛА «003»), полет с индикаторной скоростью $V_{ind} = 30$ км/ч в штатных условиях функционирования. Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Object — объект управления; RefModel — эталонная модель

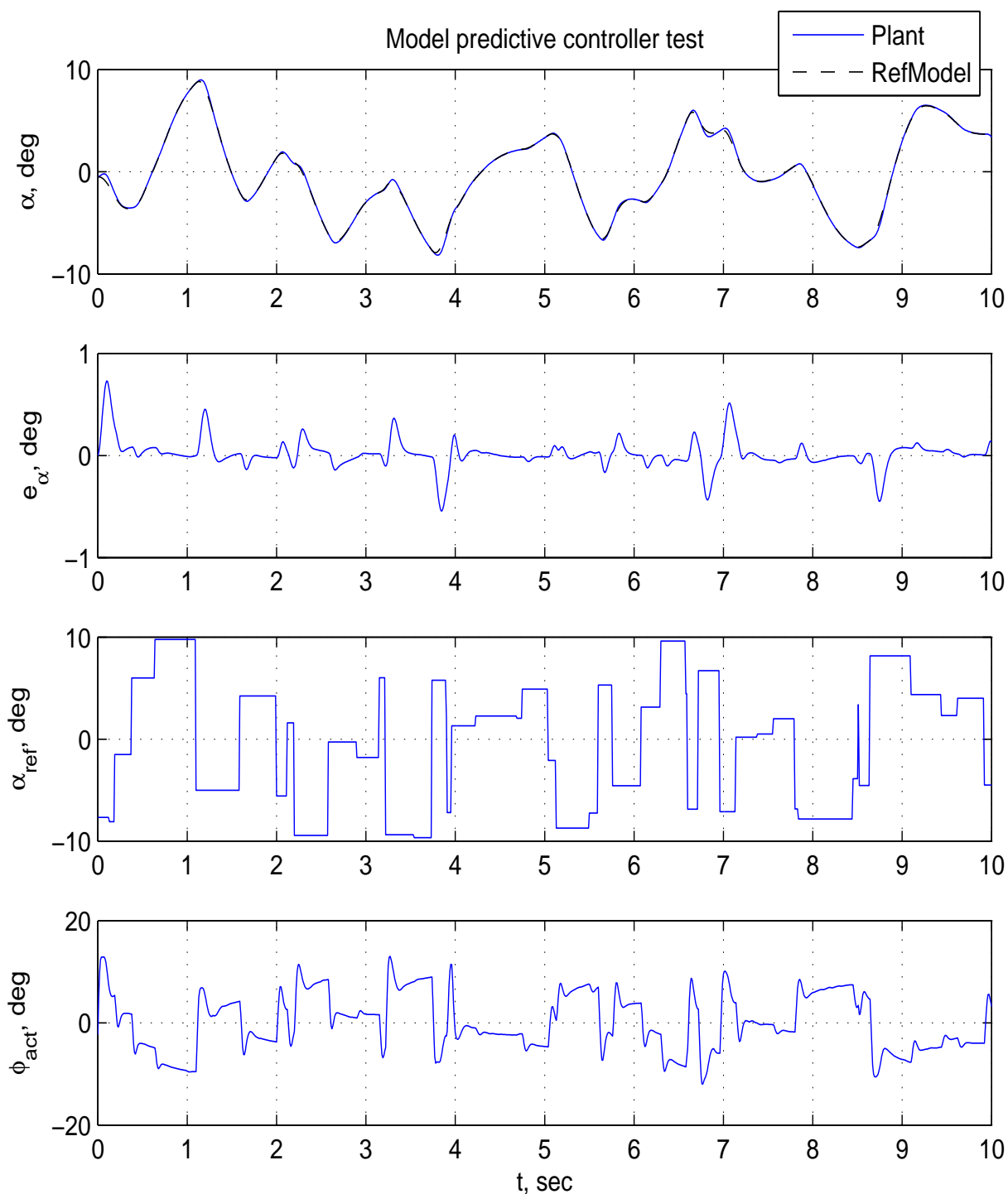


Рис. А.61. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью (мини-БПЛА X-04), полет с индикаторной скоростью $V_{ind} = 70$ км/ч в штатных условиях функционирования. Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Object — объект управления; RefModel — эталонная модель

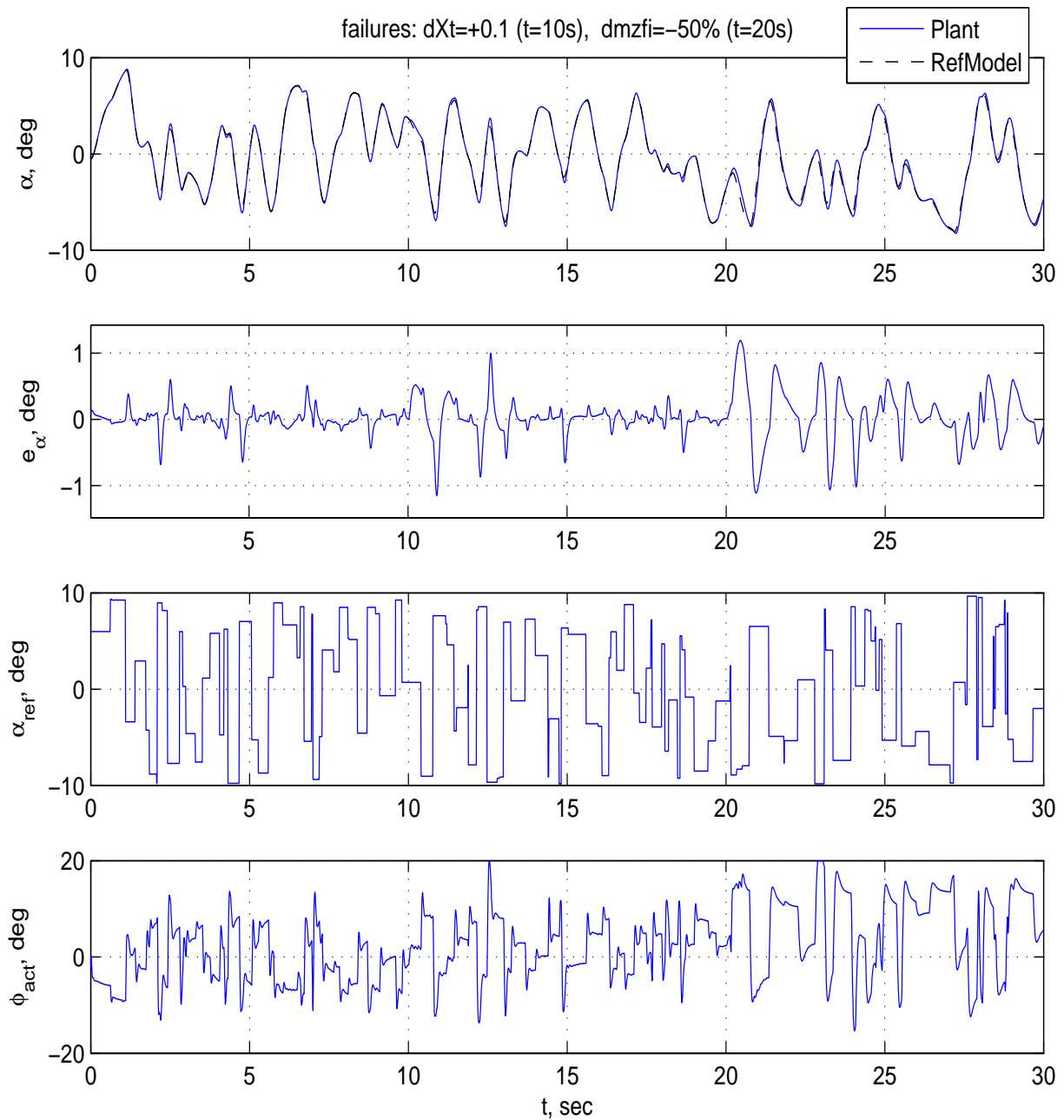


Рис. А.62. Результаты вычислительного эксперимента для системы управления с прогнозирующей моделью (мини-БПЛА X-04), полет с индикаторной скоростью $V_{ind} = 70$ км/ч при возникновении отказных ситуаций в момент времени $t = 10$ с (смещение центровки на 10% назад) и затем в момент времени $t = 20$ с (уменьшение на 50% эффективности органа продольного управления). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; α_{ref} — задающий сигнал по углу атаки, град.; ϕ_{act} — командный сигнал для привода элевонов, град.; t — время, с; Object — объект управления; RefModel — эталонная модель

А.3. Адаптивное управление с обратной динамикой

для продольного углового движения ЛА

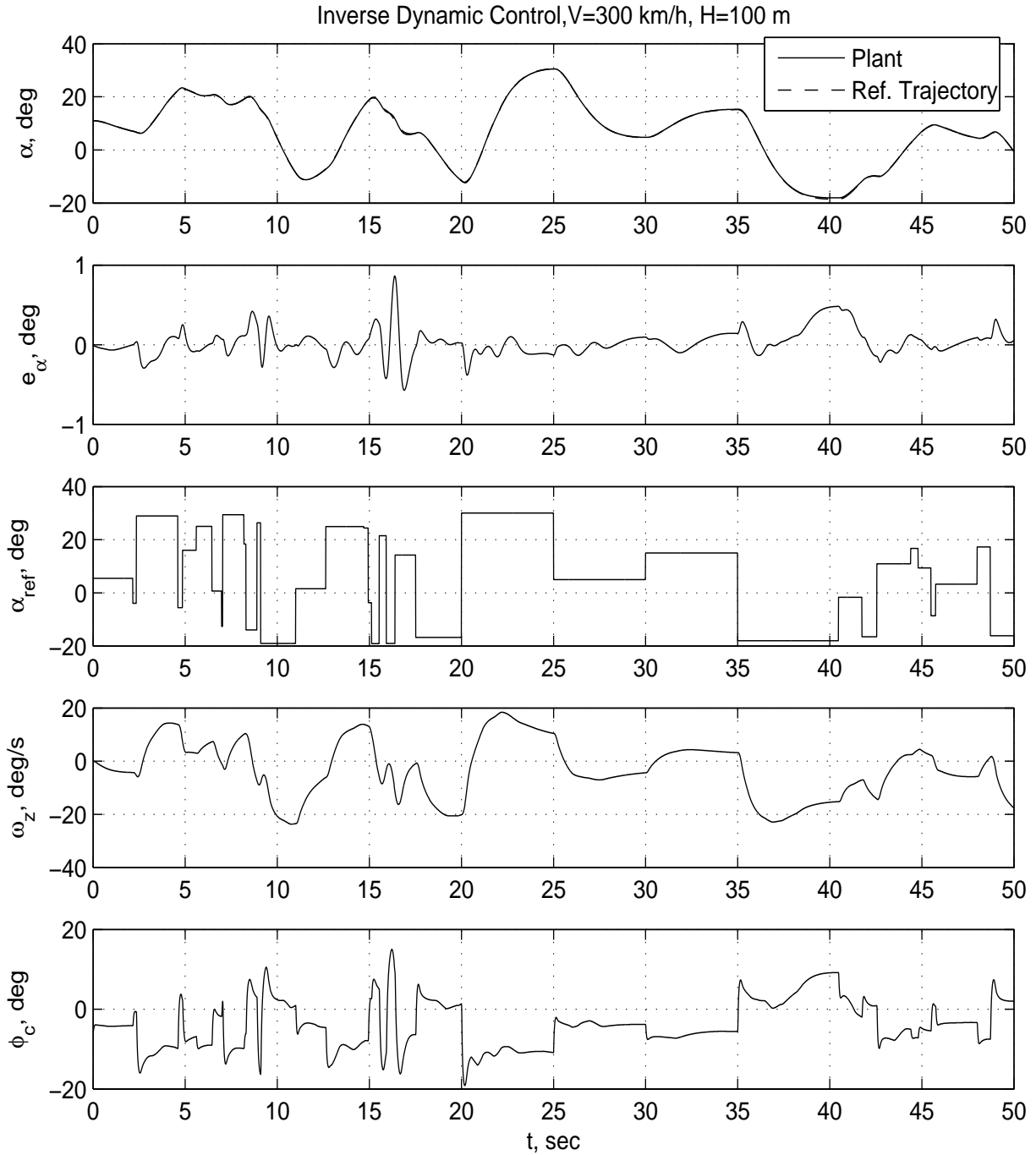


Рис. А.63. Результаты вычислительного эксперимента для системы управления с обратной динамикой (режим полета $V = 300$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; α_{ref} — задающий сигнал по углу атаки, град; ϕ_c — командный сигнал для привода управляемого стабилизатора, град; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

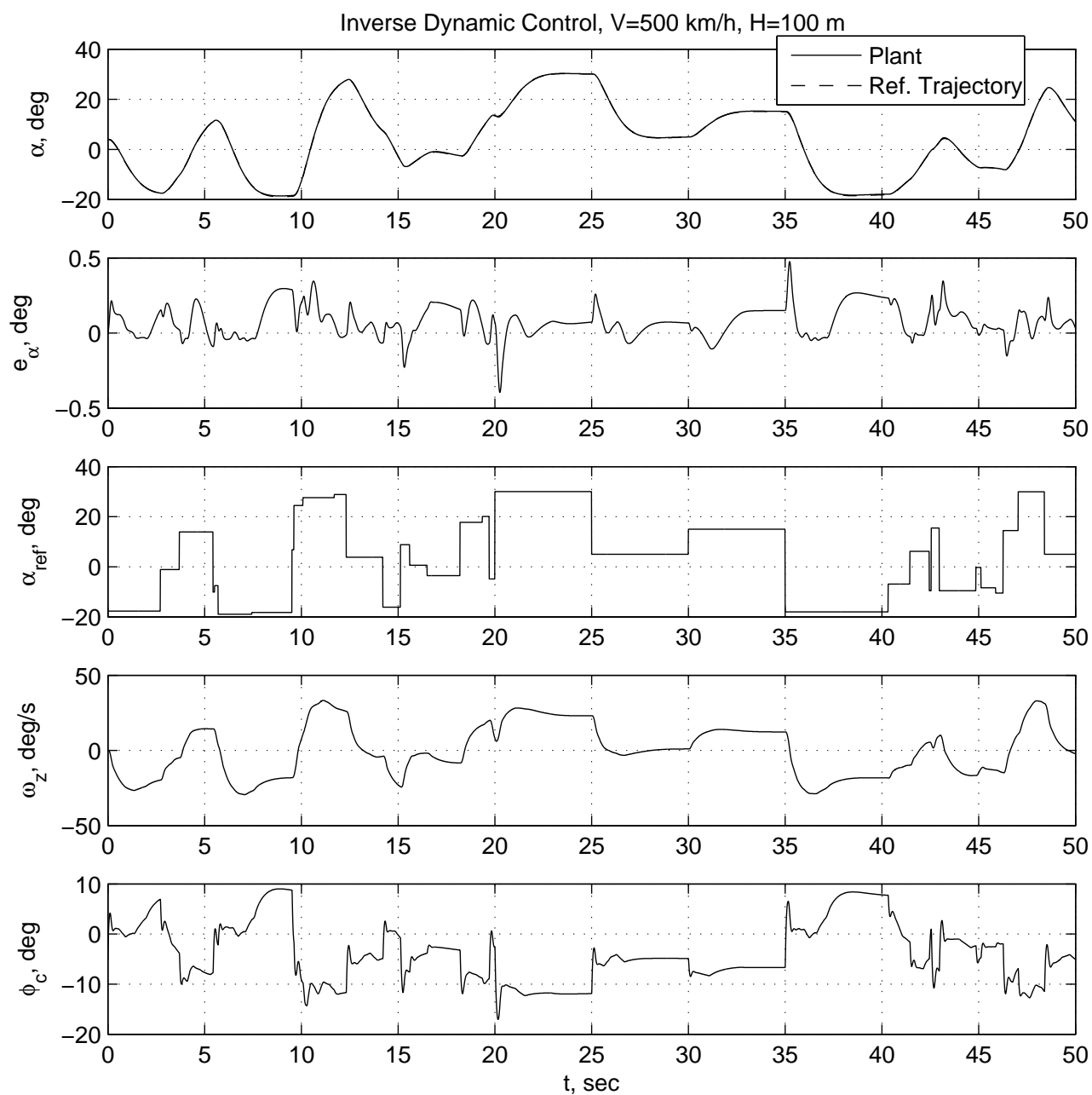


Рис. А.64. Результаты вычислительного эксперимента для системы управления с обратной динамикой (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; α_{ref} — задающий сигнал по углу атаки, град; ϕ_c — командный сигнал для привода управляемого стабилизатора, град; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

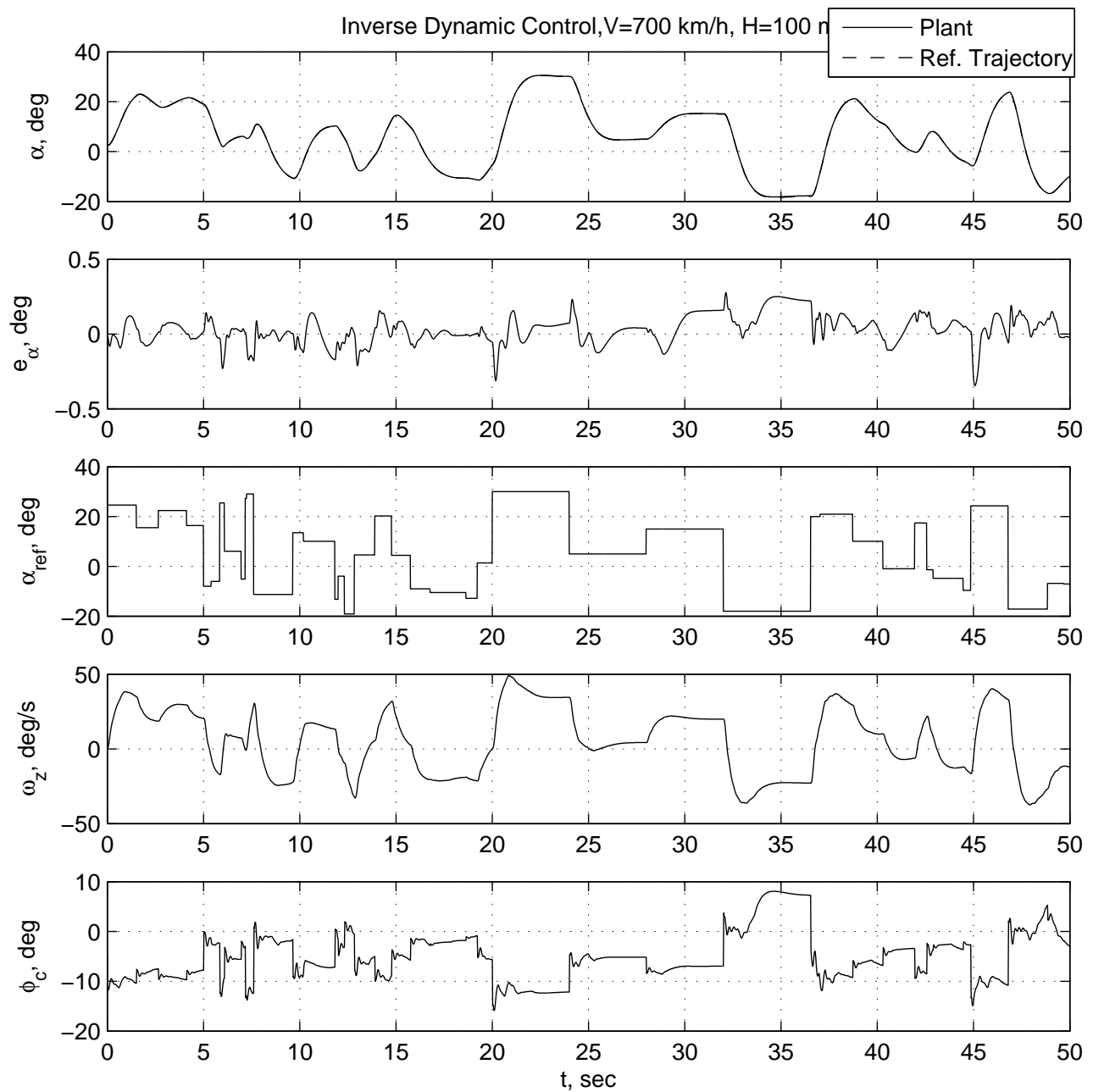


Рис. А.65. Результаты вычислительного эксперимента для системы управления с обратной динамикой (режим полета $V = 700$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; α_{ref} — задающий сигнал по углу атаки, град; ϕ_c — командный сигнал для привода управляемого стабилизатора, град; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

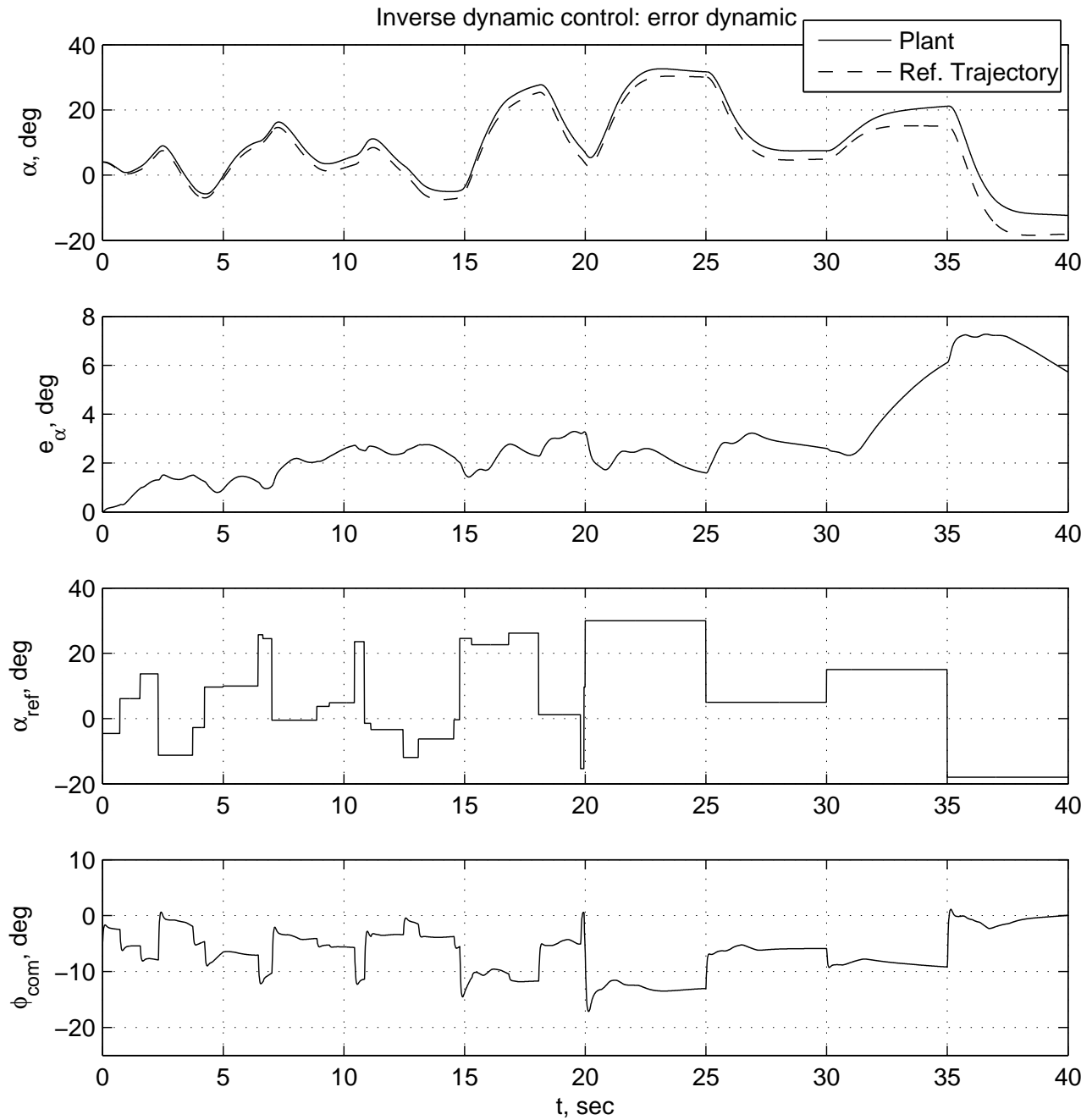


Рис. А.66. Результаты вычислительного эксперимента для системы управления с обратной динамикой — динамика ошибки (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; α_{ref} — задающий сигнал по углу атаки, град; ϕ_c — командный сигнал для привода управляемого стабилизатора, град; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

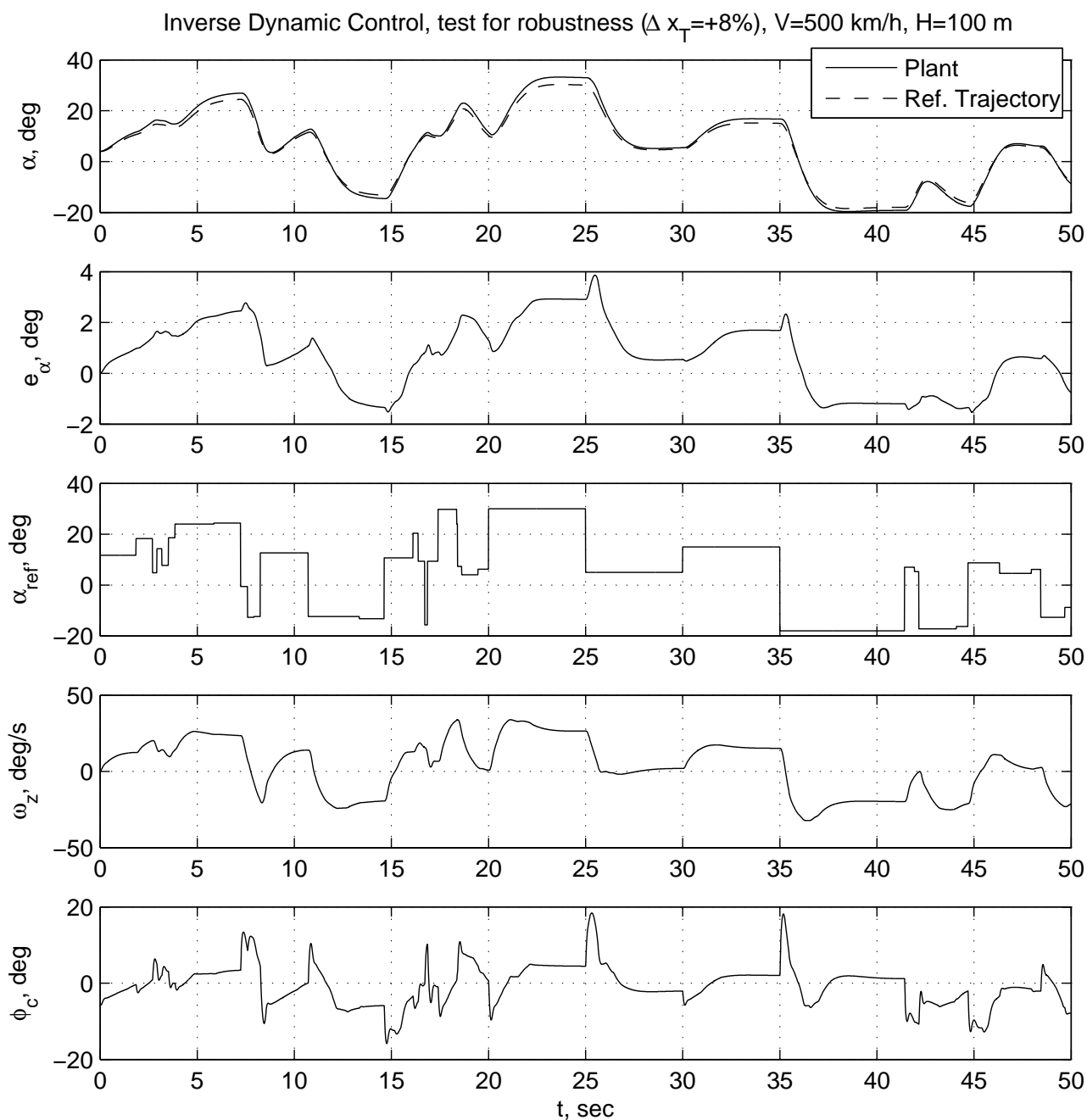


Рис. А.67. Результаты вычислительного эксперимента для системы управления с обратной динамикой — тест на робастность (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; α_{ref} — задающий сигнал по углу атаки, град; ϕ_c — командный сигнал для привода управляемого стабилизатора, град; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

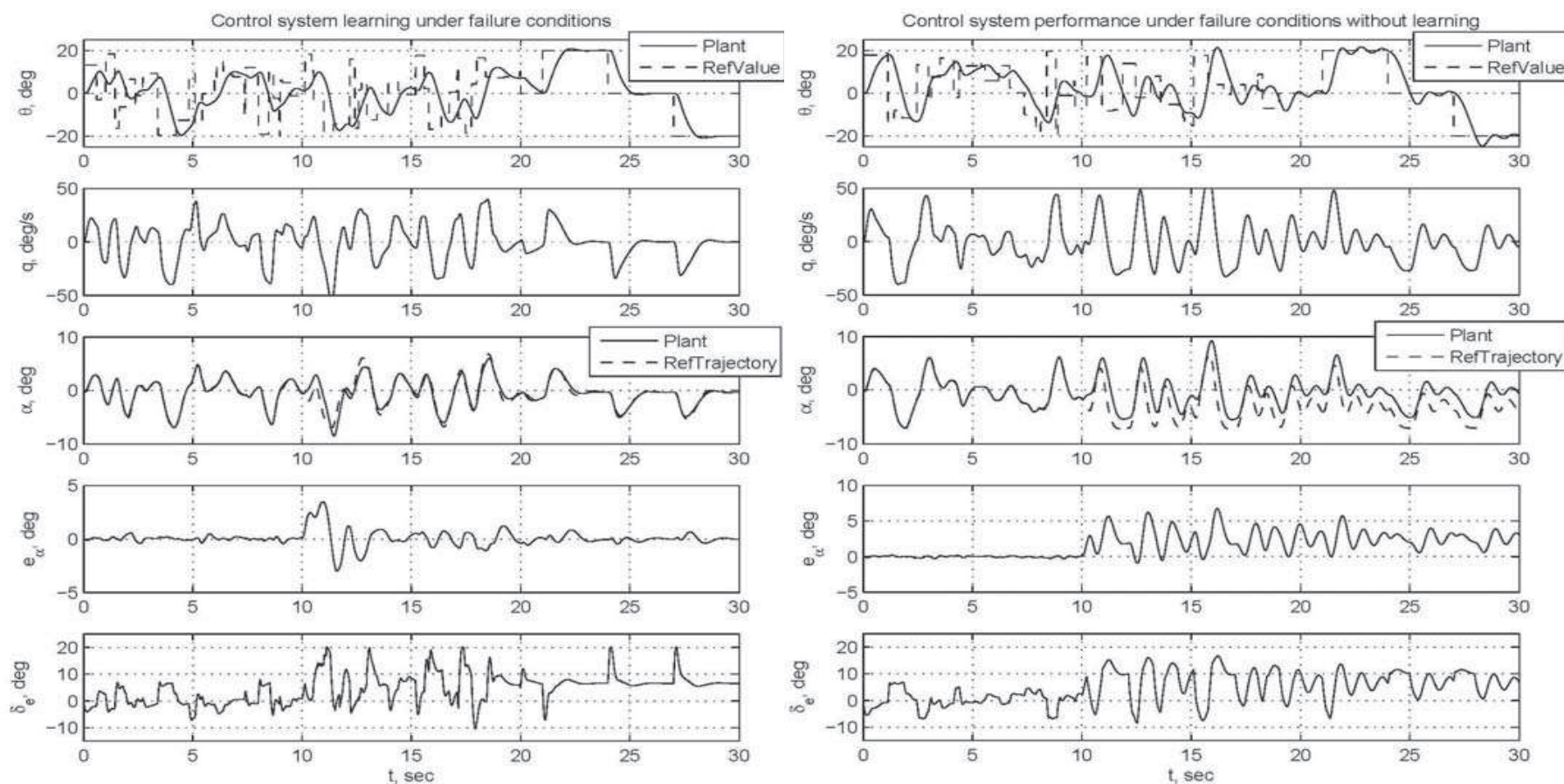


Рис. А.68. Результаты вычислительного эксперимента для системы управления с обратной динамикой (мини-БПЛА X-04) с включенным (слева) и выключенным (справа) механизмом адаптации. Полет с индикаторной скоростью $V_{ind} = 70$ км/ч; в момент времени $t = 10$ с одновременно возникают две отказные ситуации (смещение центровки на 15% назад и уменьшение на 50% эффективности органа продольного управления). Обозначения: α — угол атаки, град.; e_α — ошибка отслеживания заданного угла атаки, град.; θ — угол тангажа, град.; $q = \omega_z$ — угловая скорость тангажа, град./с; δ_e — угол отклонения руля высоты, град.; t — время, с; Plant — объект управления; RefValue — задающий сигнал; RefTrajectory — выход эталонной модели

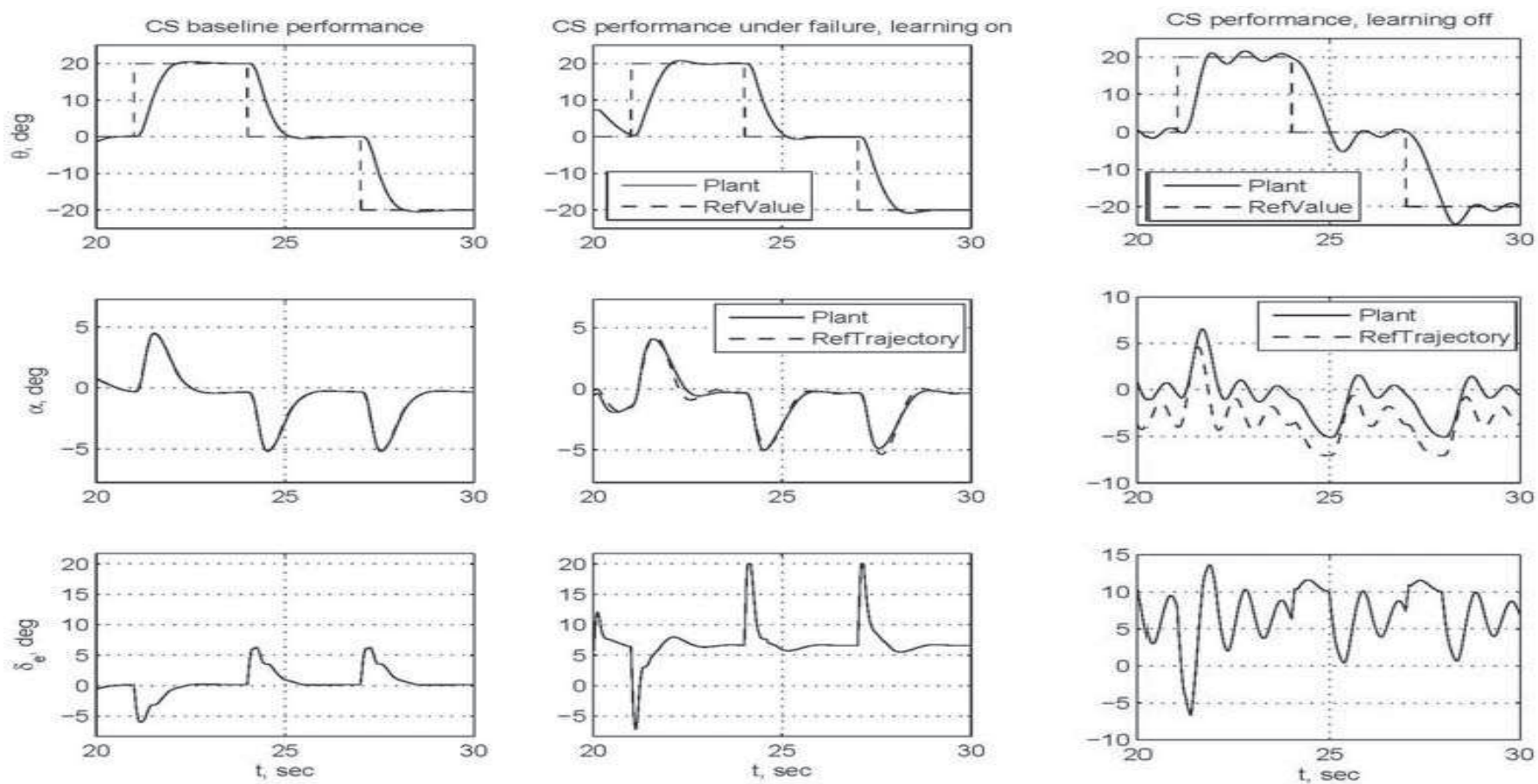


Рис. А.69. Результаты вычислительного эксперимента для системы управления с обратной динамикой (мини-БПЛА X-04): до возникновения отказной ситуации (слева) и после нее с включенным (в центре) и выключенным (справа) механизмом адаптации. Полет с индикаторной скоростью $V_{ind} = 70$ км/ч; в момент времени $t = 10$ с одновременно возникают две отказные ситуации (смещение центровки на 15% назад и уменьшение на 50% эффективности органа продольного управления). Обозначения: α — угол атаки, град.; θ — угол тангажа, град.; δ_e — угол отклонения руля высоты, град.; t — время, с; CS (Control System) — система управления; Plant — объект управления; RefValue — задающий сигнал; RefTrajectory — выход эталонной модели

А.4. Адаптивное управление с инверсной моделью

для продольного углового движения ЛА

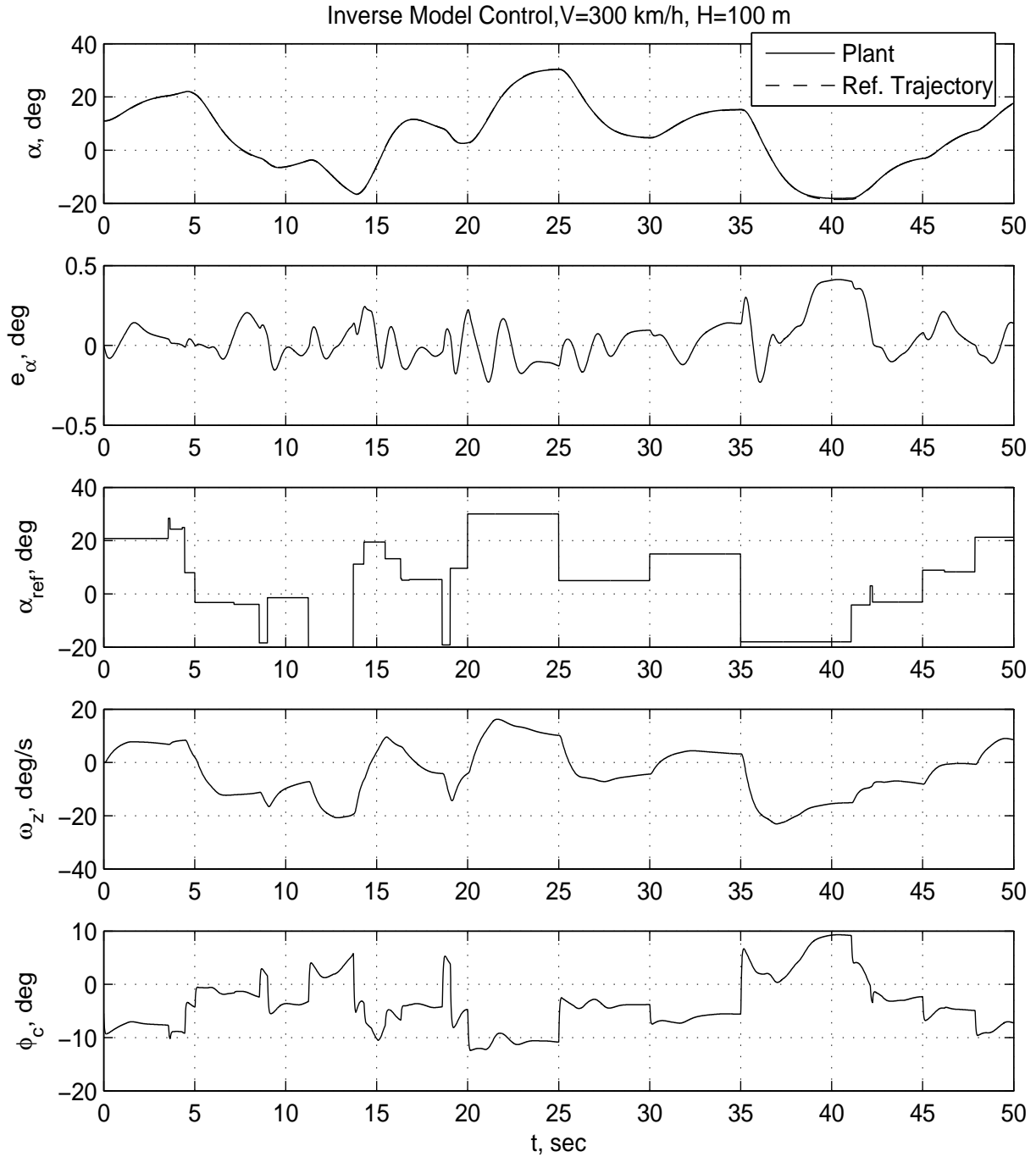


Рис. А.70. Результаты вычислительного эксперимента для системы управления с инверсной моделью (режим полета $V = 300$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; α_{ref} — задающий сигнал по углу атаки, град; ϕ_c — командный сигнал для привода управляемого стабилизатора, град; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

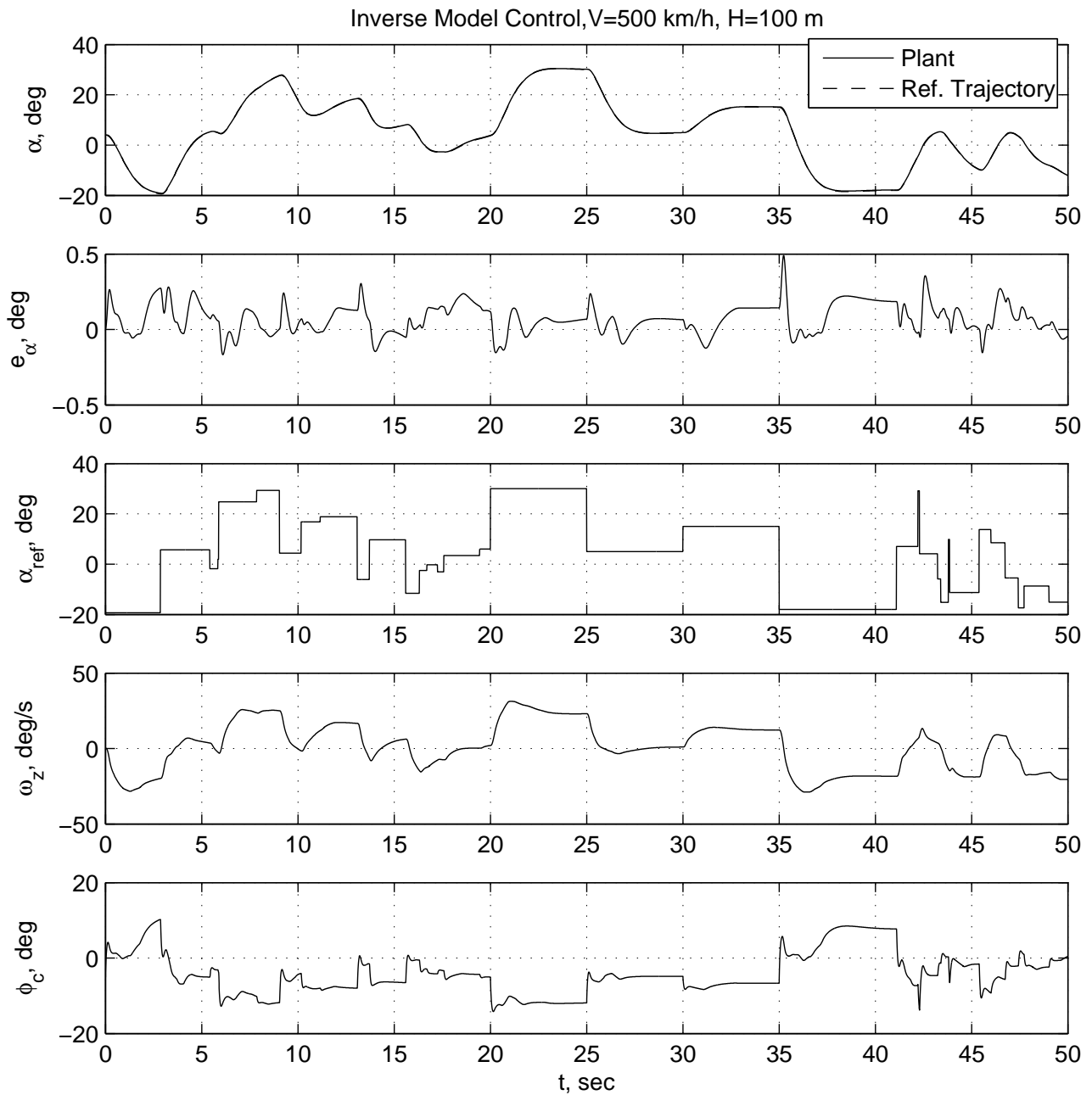


Рис. А.71. Результаты вычислительного эксперимента для системы управления с инверсной моделью (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; α_{ref} — задающий сигнал по углу атаки, град; ϕ_c — командный сигнал для привода управляемого стабилизатора, град; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

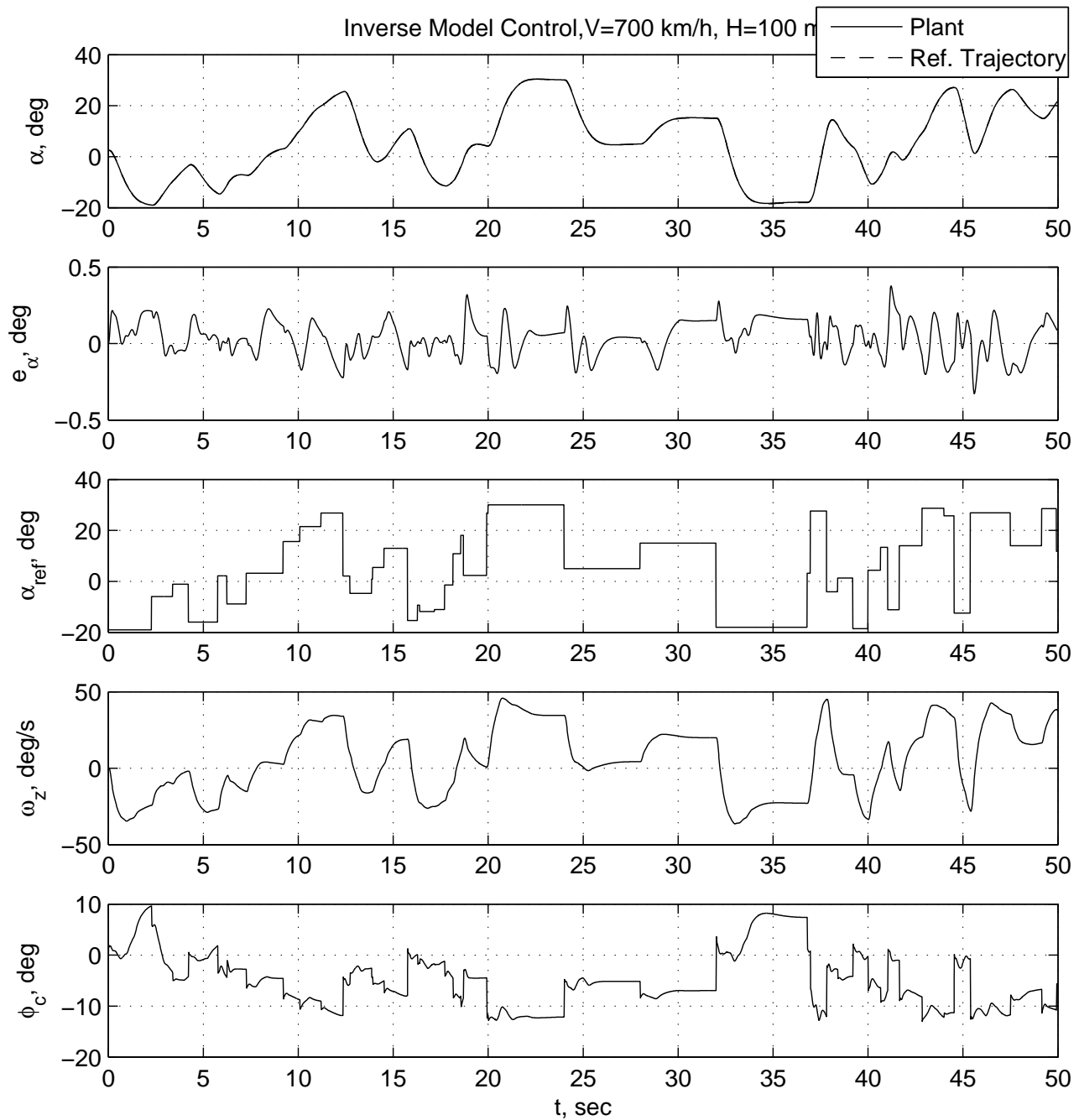


Рис. А.72. Результаты вычислительного эксперимента для системы управления с инверсной моделью (режим полета $V = 700$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; α_{ref} — задающий сигнал по углу атаки, град; ϕ_c — командный сигнал для привода управляемого стабилизатора, град; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

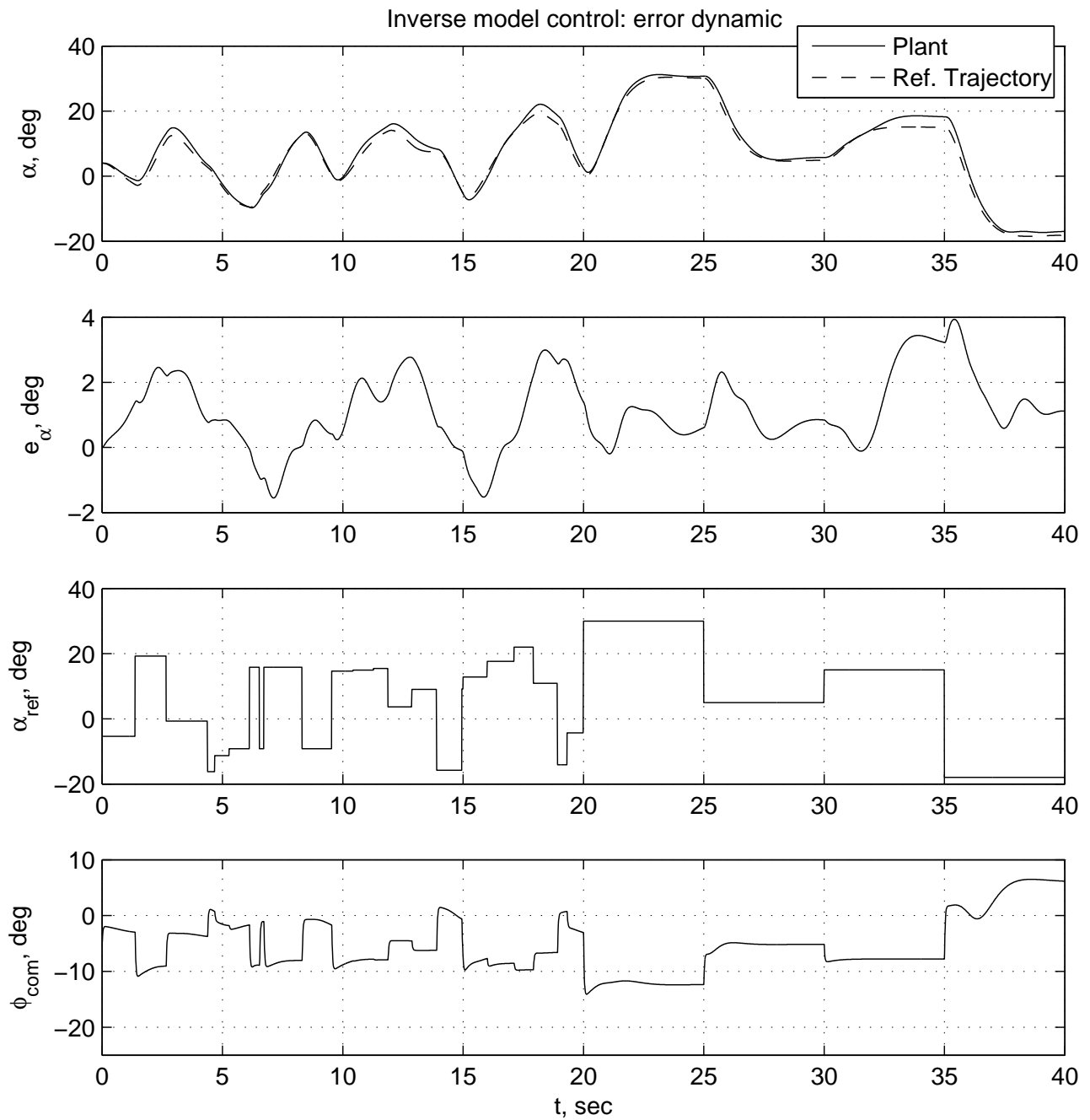


Рис. А.73. Результаты вычислительного эксперимента для системы управления с инверсной моделью — динамика ошибки (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; α_{ref} — задающий сигнал по углу атаки, град; ϕ_c — командный сигнал для привода управляемого стабилизатора, град; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

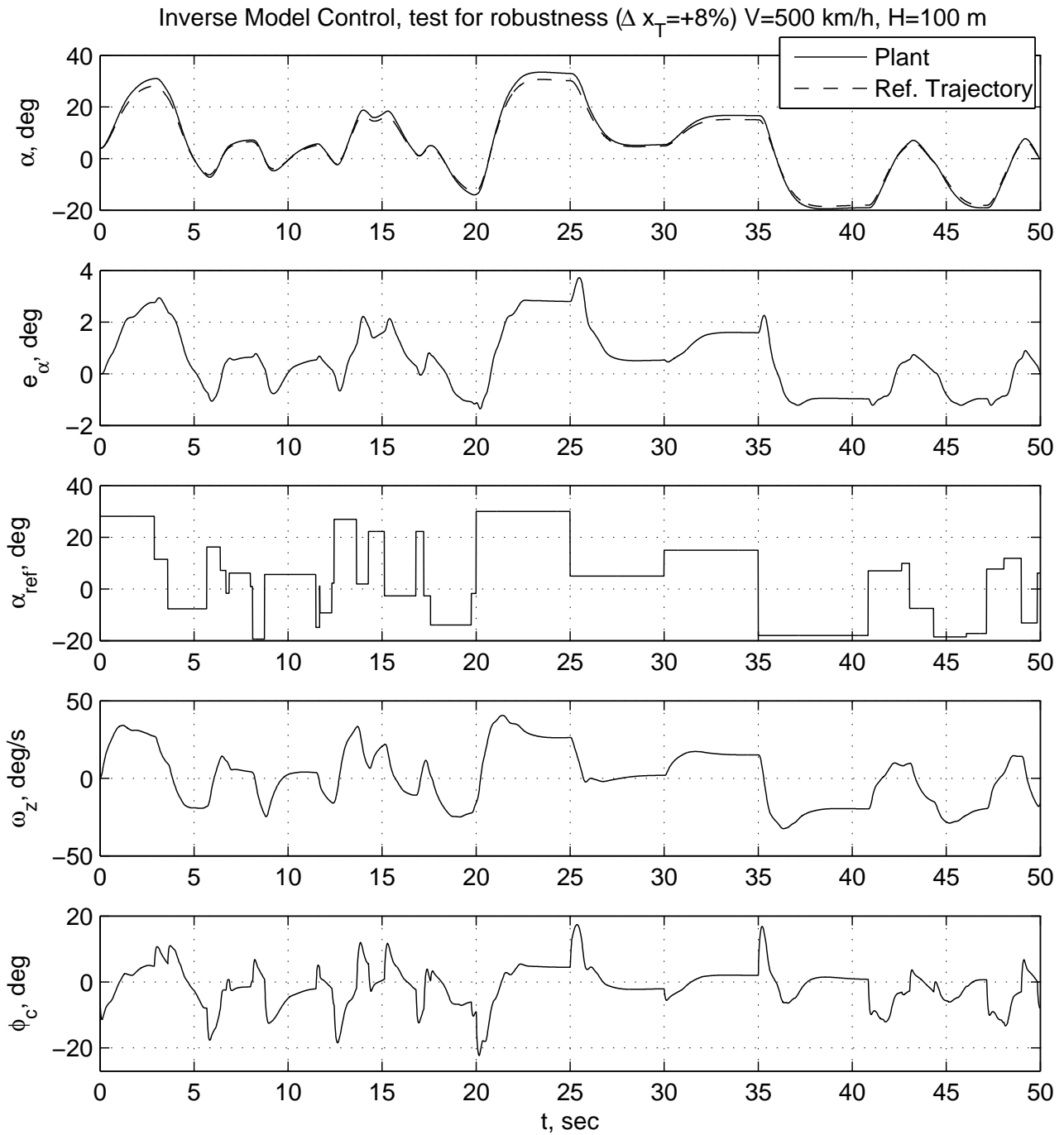


Рис. А.74. Результаты вычислительного эксперимента для системы управления с инверсной моделью — тест на робастность (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; α_{ref} — задающий сигнал по углу атаки, град; ϕ_c — командный сигнал для привода управляемого стабилизатора, град; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

**А.5. Адаптивное управление с обратной динамикой и с инверсной моделью
для пространственного движения ЛА**

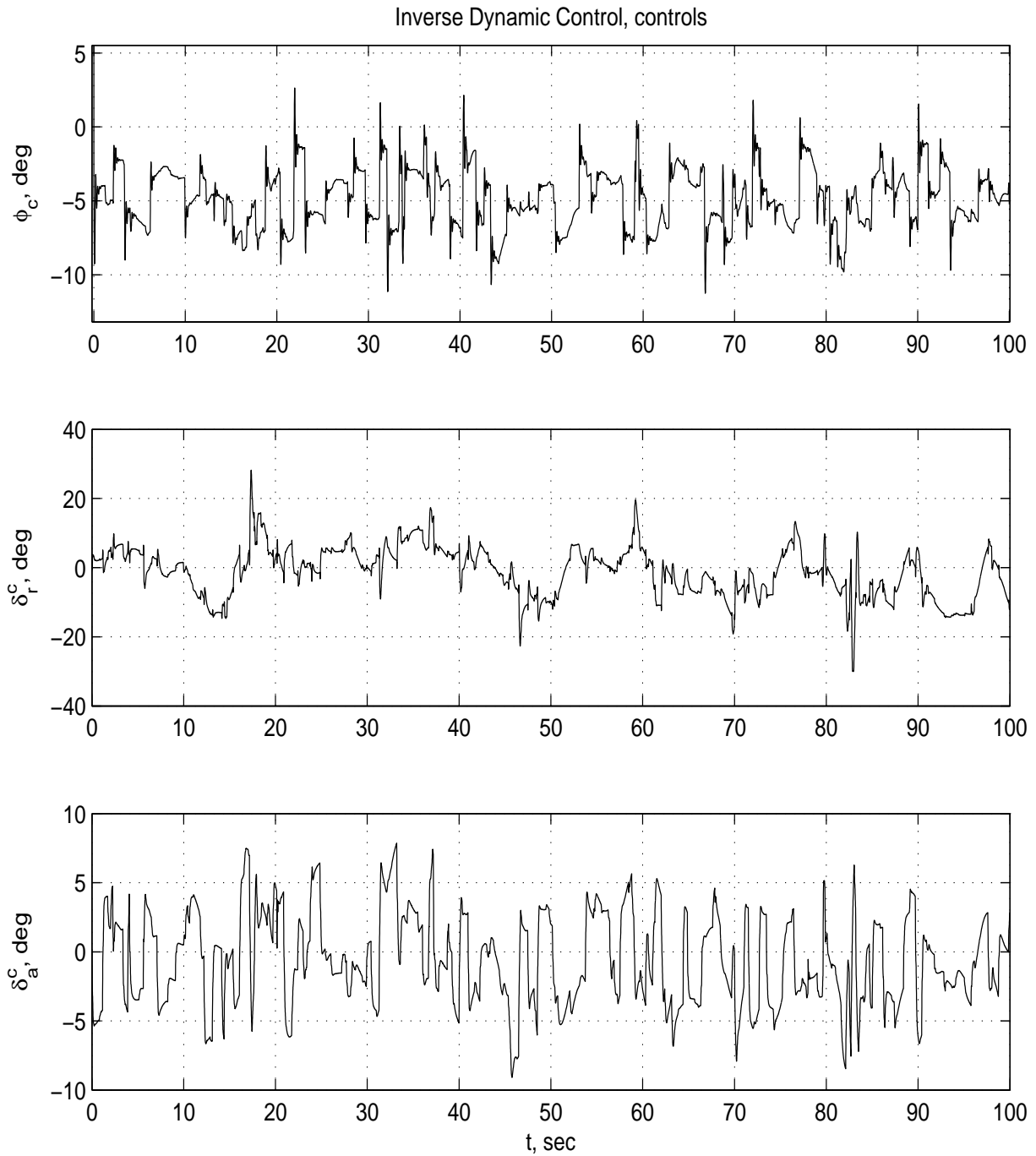


Рис. А.75. Результаты вычислительного эксперимента для системы управления с обратной динамикой без использования декомпозиции — управляющие сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: ϕ_c — командный сигнал для привода управляемого стабилизатора, град; δ_r^c — командный сигнал для привода руля направления, град; δ_a^c — командный сигнал для привода элеронов, град; t — время, с

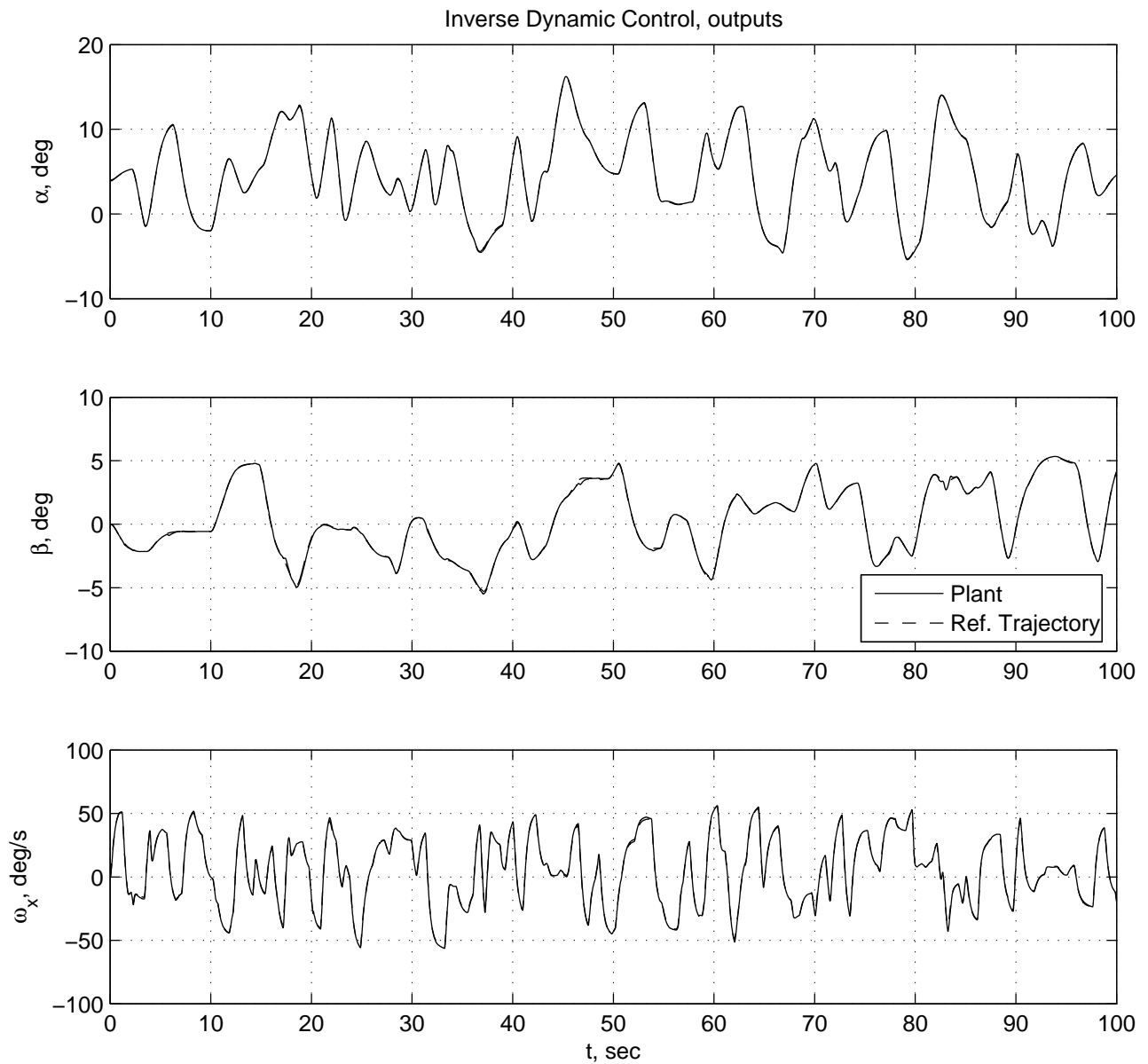


Рис. А.76. Результаты вычислительного эксперимента для системы управления с обратной динамикой без использования декомпозиции — выходные сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; β — угол скольжения, град; ω_x — угловая скорость крена, град/с; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

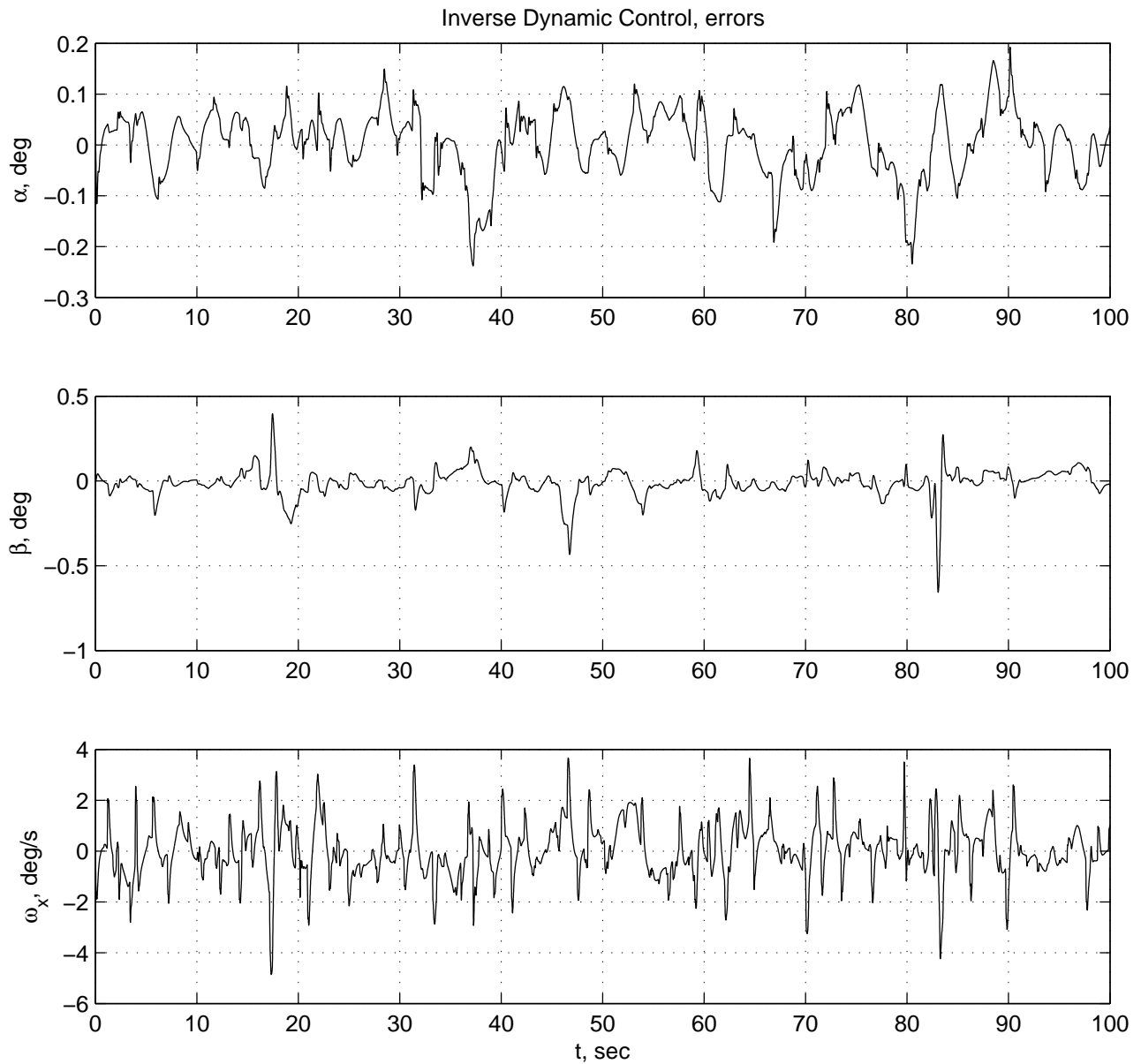


Рис. А.77. Результаты вычислительного эксперимента для системы управления с обратной динамикой без использования декомпозиции — сигналы ошибки (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — ошибка по углу атаки, град; β — ошибка по углу скольжения, град; ω_x — ошибка по угловой скорости крена, град/с; t — время, с

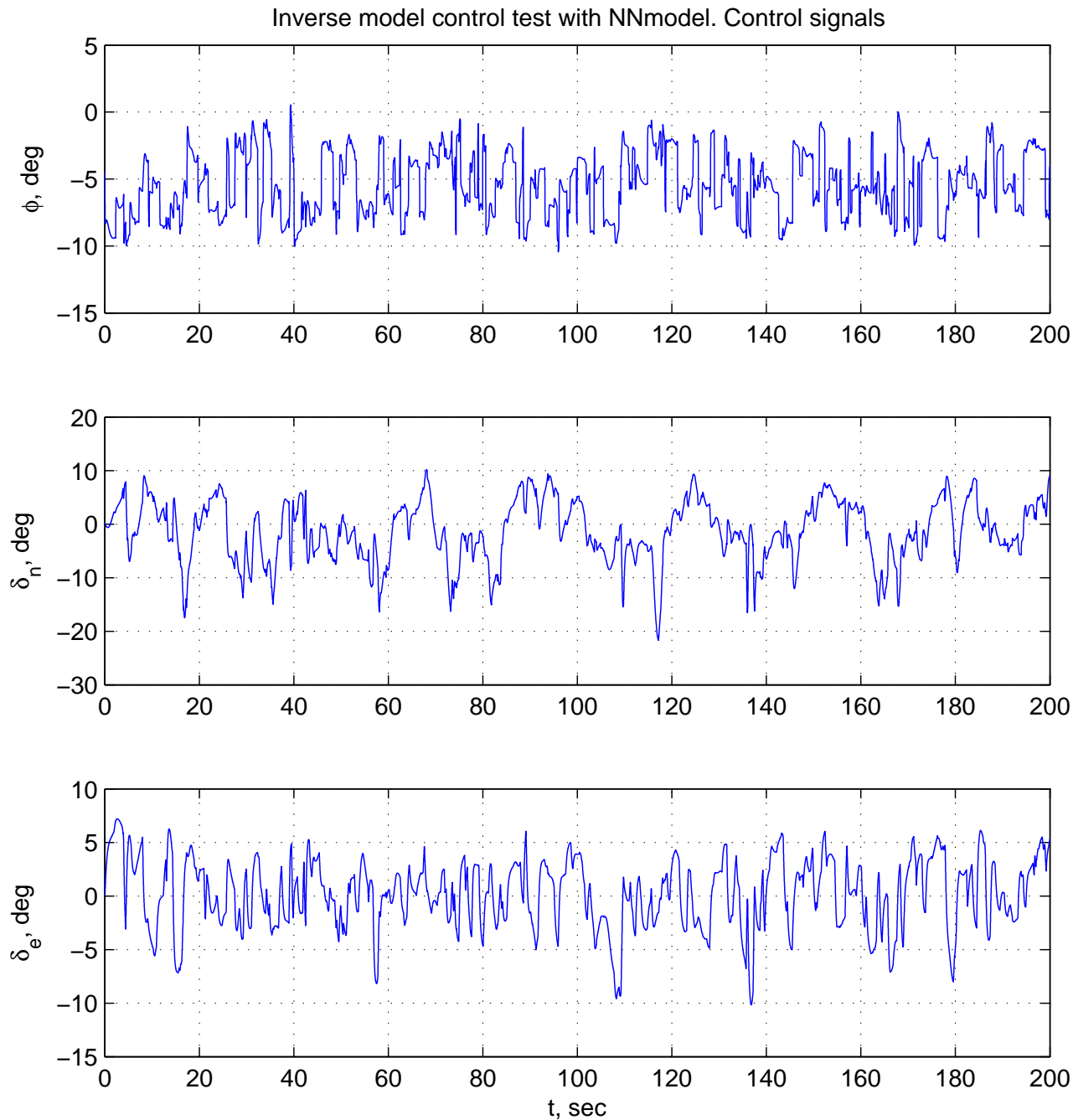


Рис. А.78. Результаты вычислительного эксперимента для системы управления с инверсной моделью без использования декомпозиции — управляющие сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: ϕ_c — командный сигнал для привода управляемого стабилизатора, град; δ_r^c — командный сигнал для привода руля направления, град; δ_a^c — командный сигнал для привода элеронов, град; t — время, с

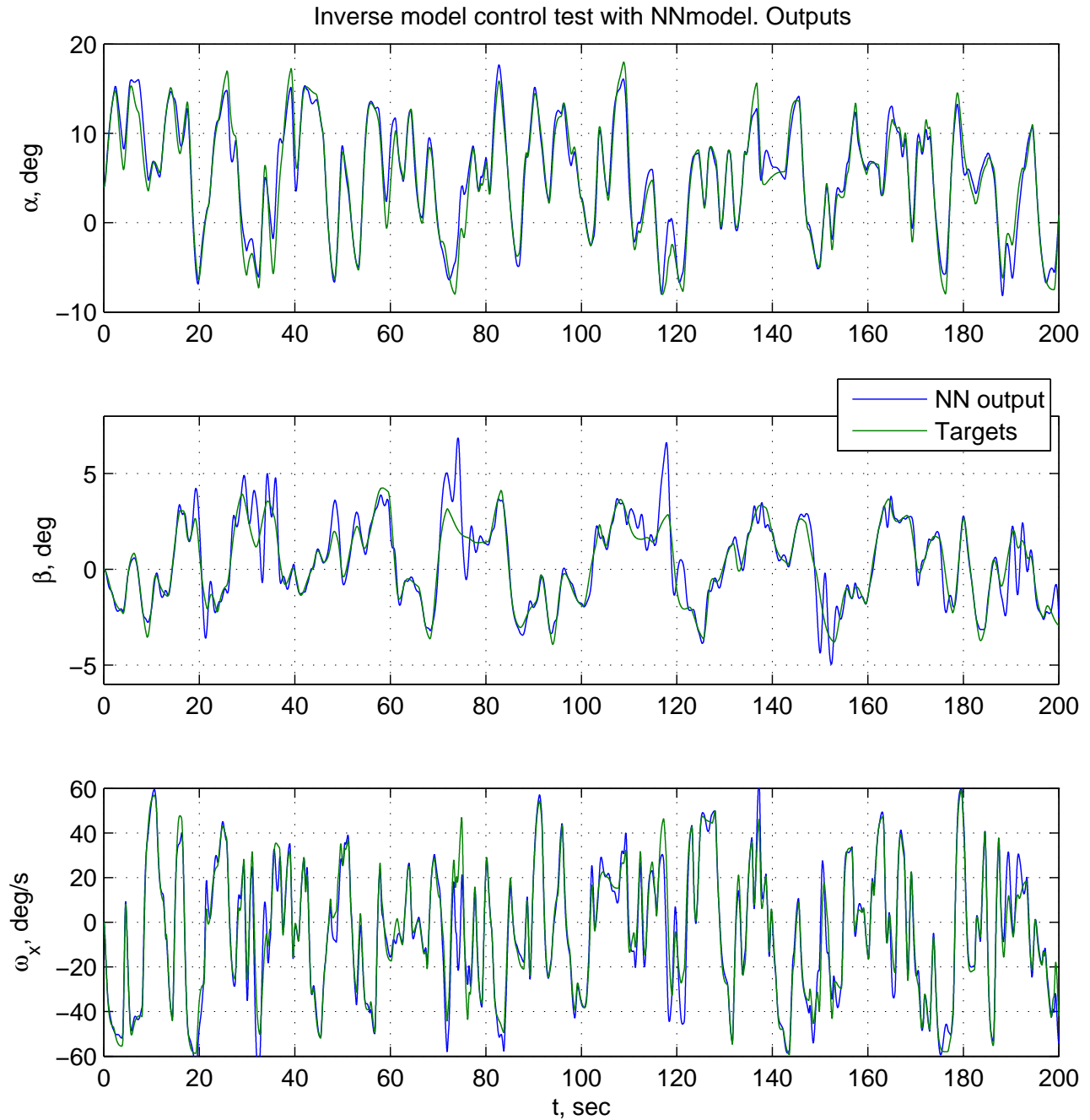


Рис. А.79. Результаты вычислительного эксперимента для системы управления с инверсной моделью без использования декомпозиции — выходные сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; β — угол скольжения, град; ω_x — угловая скорость крена, град/с; t — время, с; NN output — выход НС-модели; Targets — эталонная траектория

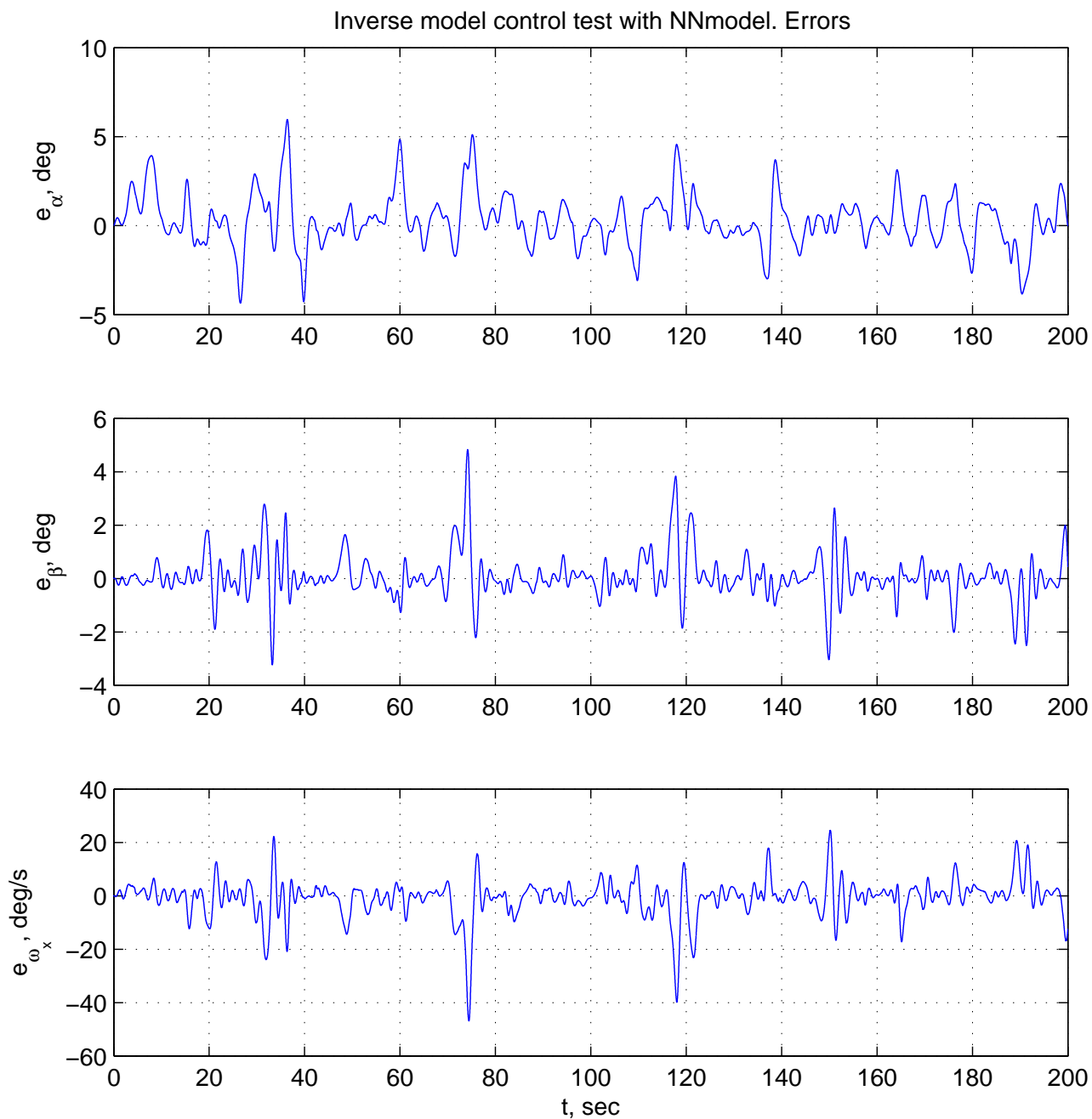


Рис. А.80. Результаты вычислительного эксперимента для системы управления с инверсной моделью без использования декомпозиции — сигналы ошибки (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — ошибка по углу атаки, град; β — ошибка по углу скольжения, град; ω_x — ошибка по угловой скорости крена, град/с; t — время, с

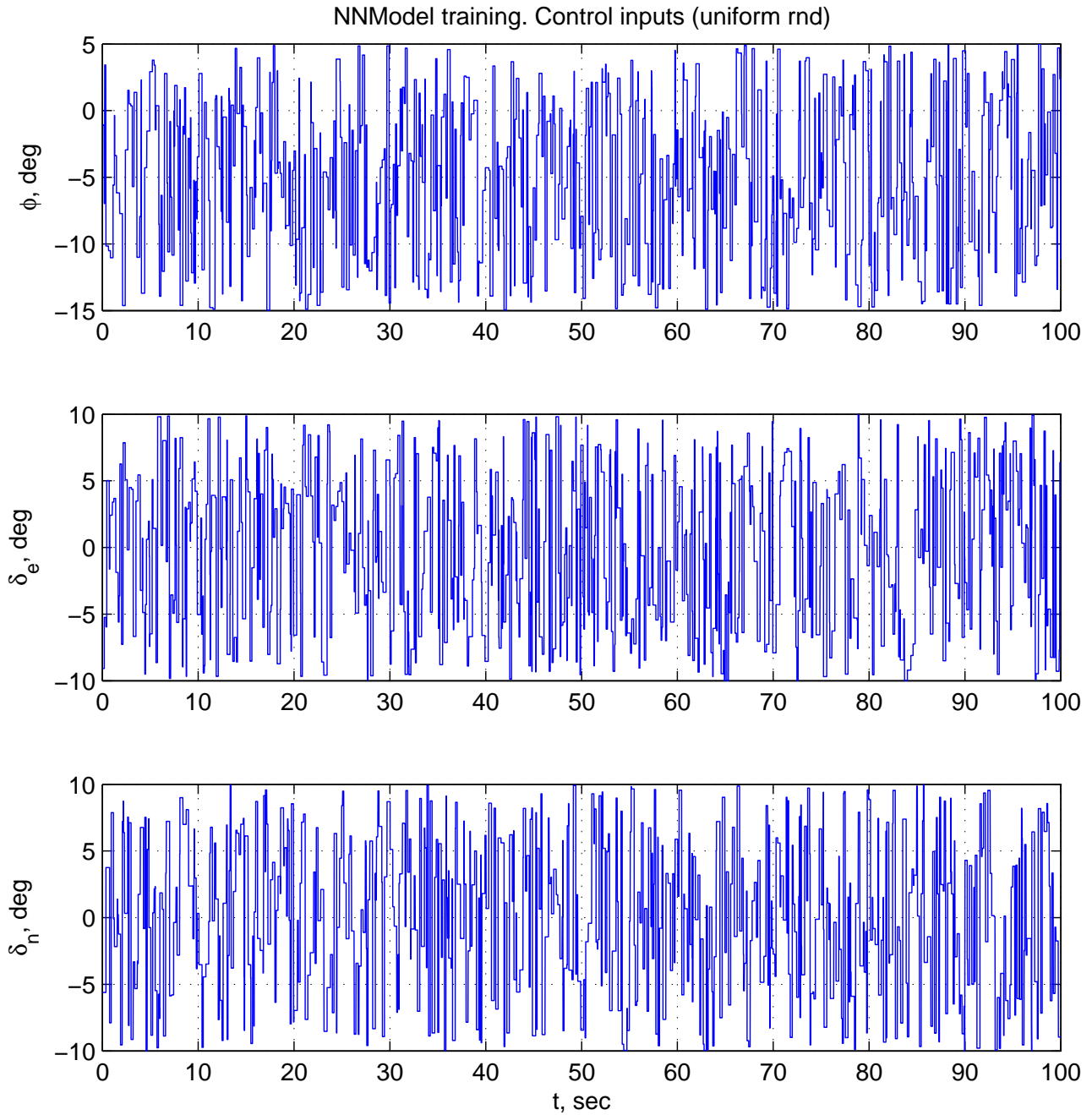


Рис. А.81. Результаты вычислительного эксперимента по обучению нейросетевой модели с использованием декомпозиции — случайные равномерно распределенные управляющие сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: ϕ — командный сигнал для привода управляемого стабилизатора, град; δ_n^c — командный сигнал для привода руля направления, град; δ_a^c — командный сигнал для привода элеронов, град; t — время, с

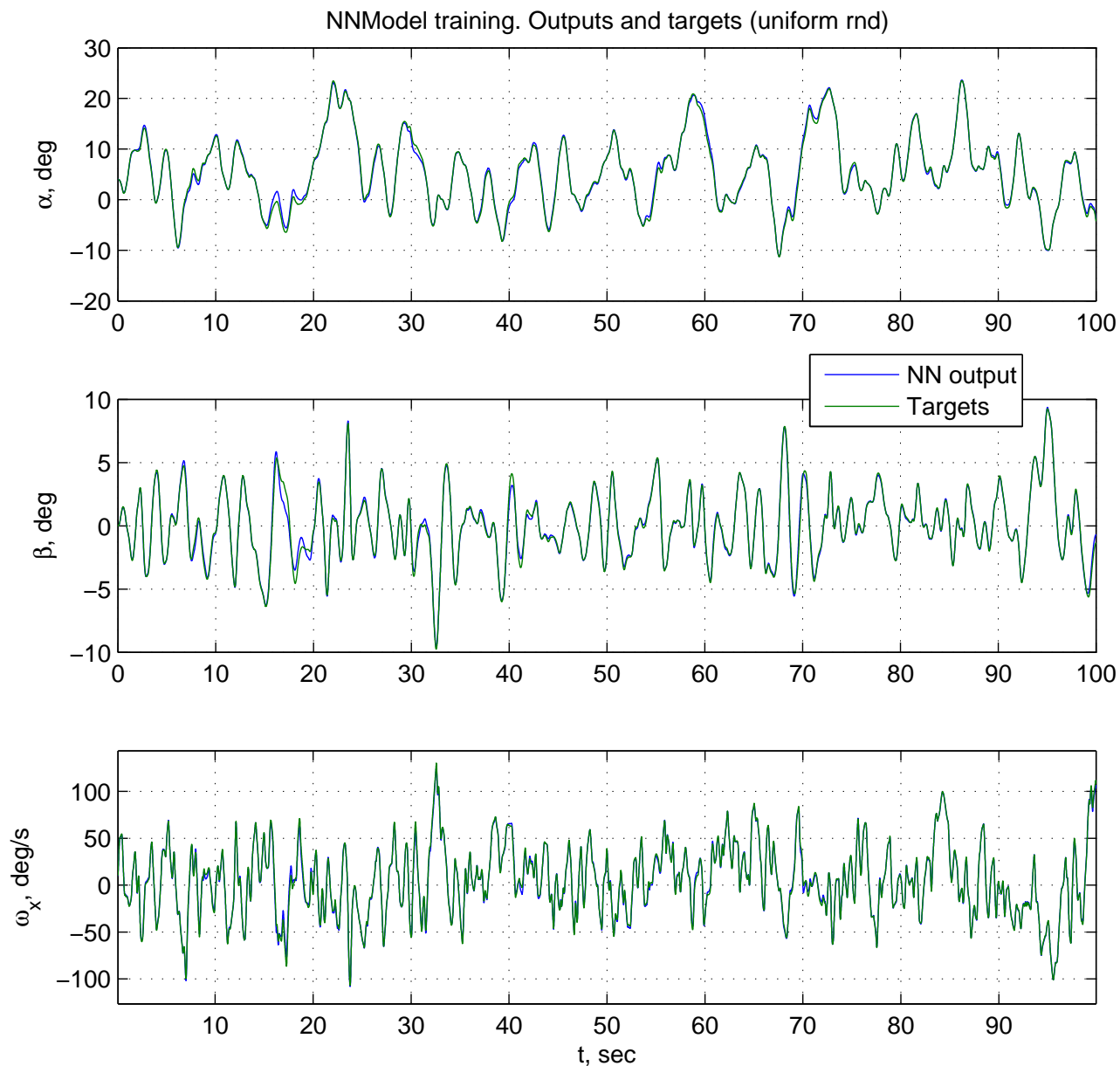


Рис. А.82. Результаты вычислительного эксперимента по обучению нейросетевой модели с использованием декомпозиции — выходные сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; β — угол скольжения, град; ω_x — угловая скорость крена, град/с; t — время, с; NN output — выход НС-модели; Targets — эталонная траектория

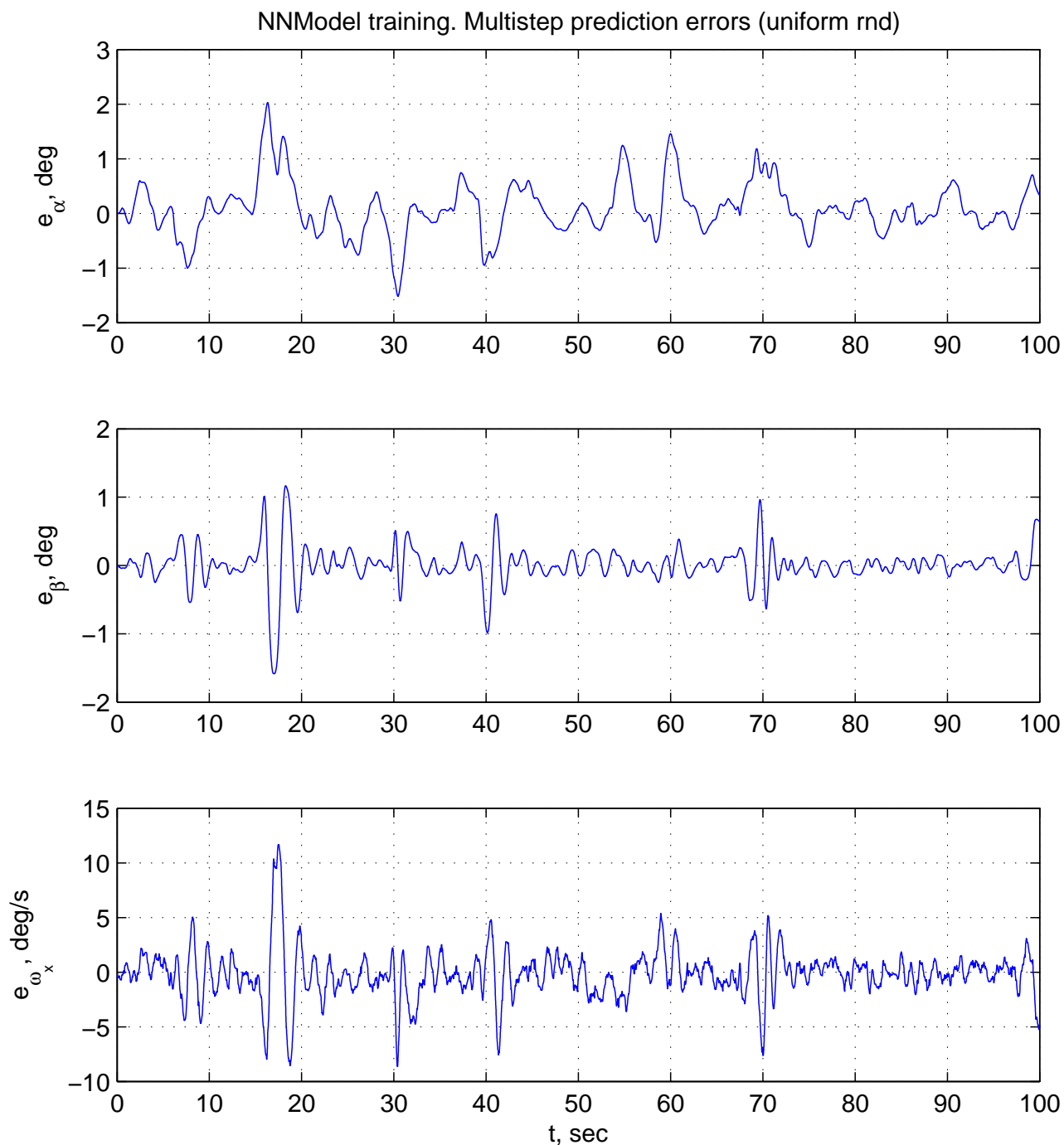


Рис. А.83. Результаты вычислительного эксперимента по обучению нейросетевой модели с использованием декомпозиции — сигналы ошибки (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: e_{α} — ошибка по углу атаки, град; e_{β} — ошибка по углу скольжения, град; e_{ω_x} — ошибка по угловой скорости крена, град/с; t — время, с

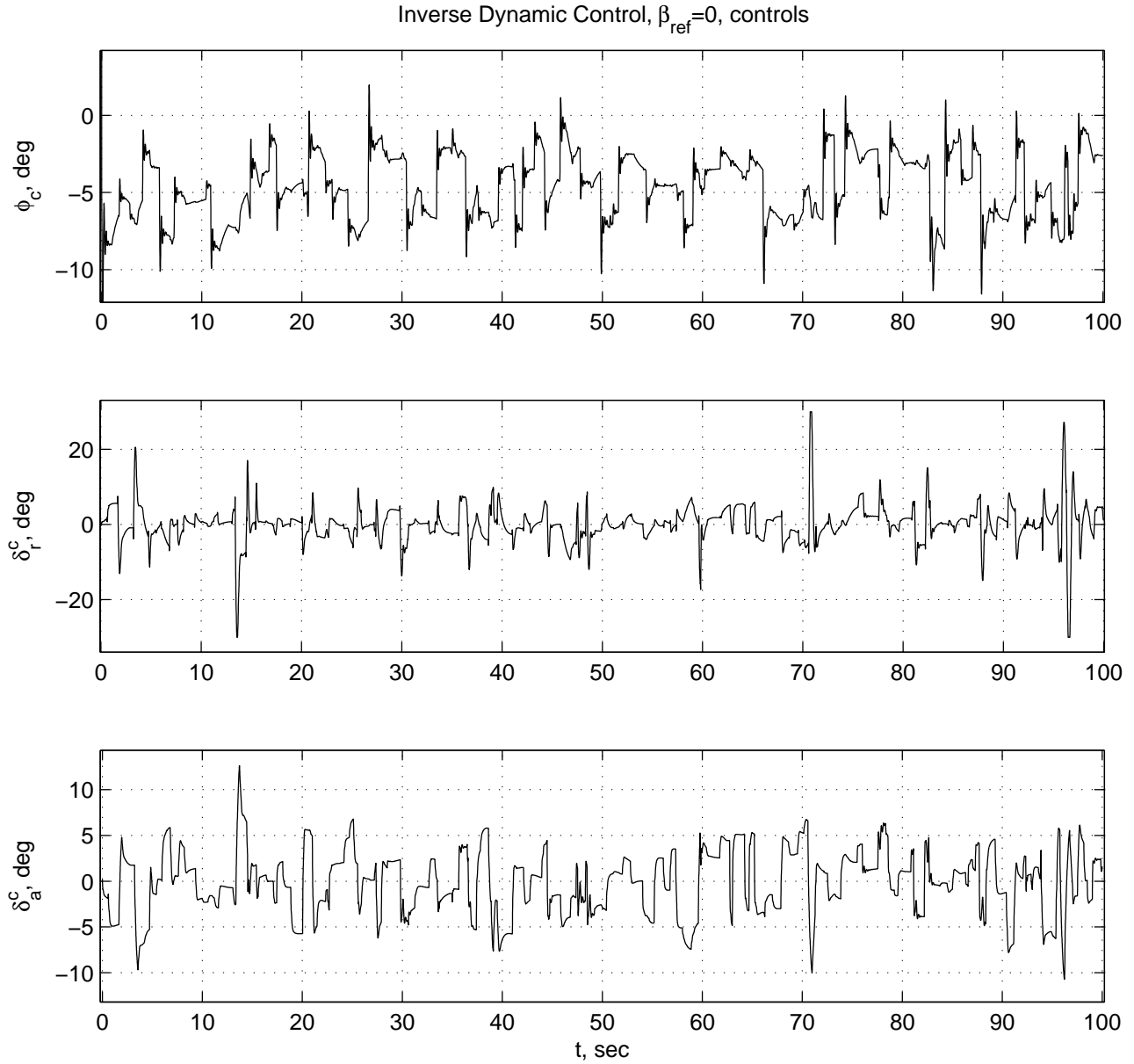


Рис. А.84. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\alpha_{ref} = var$, $\beta_{ref} = 0$) — управляющие сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: ϕ_c — командный сигнал для привода управляемого стабилизатора, град; δ_r^c — командный сигнал для привода руля направления, град; δ_a^c — командный сигнал для привода элеронов, град; t — время, с

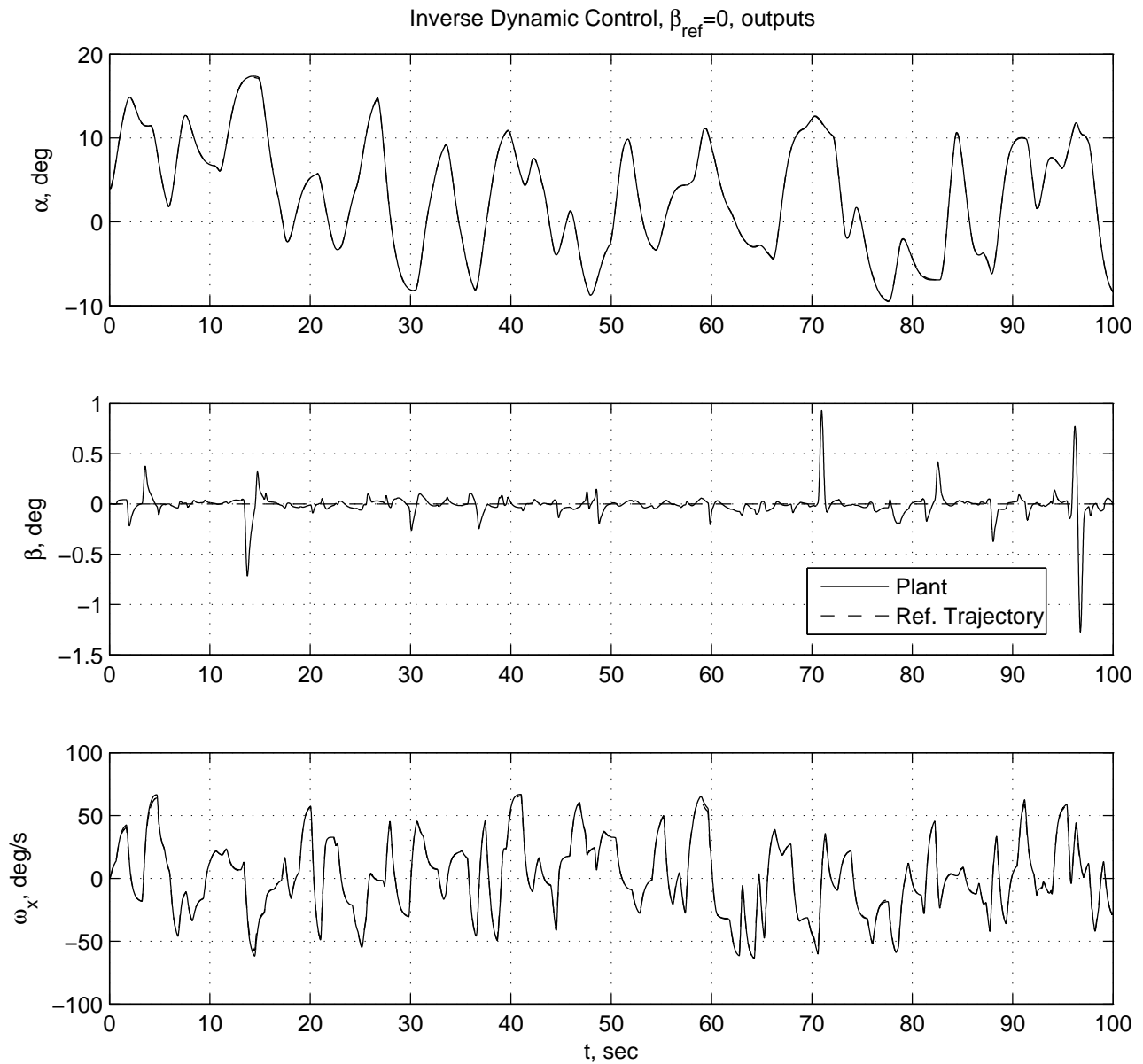


Рис. А.85. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\alpha_{ref} = var, \beta_{ref} = 0$) — выходные сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; β — угол скольжения, град; ω_x — угловая скорость крена, град/с; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

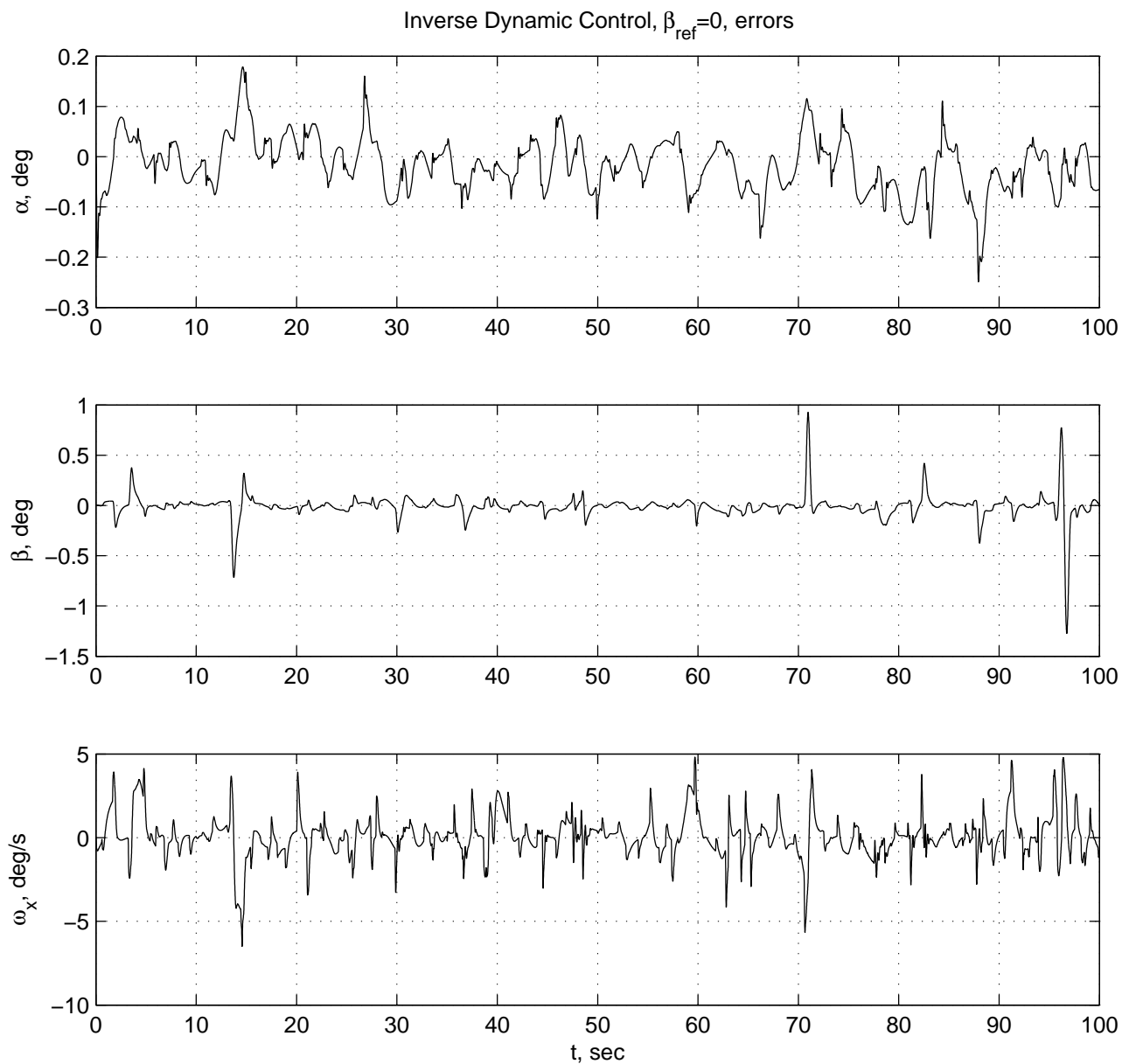


Рис. А.86. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\alpha_{ref} = var$, $\beta_{ref} = 0$) — сигналы ошибки (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — ошибка по углу атаки, град; β — ошибка по углу скольжения, град; ω_x — ошибка по угловой скорости крена, град/с; t — время, с

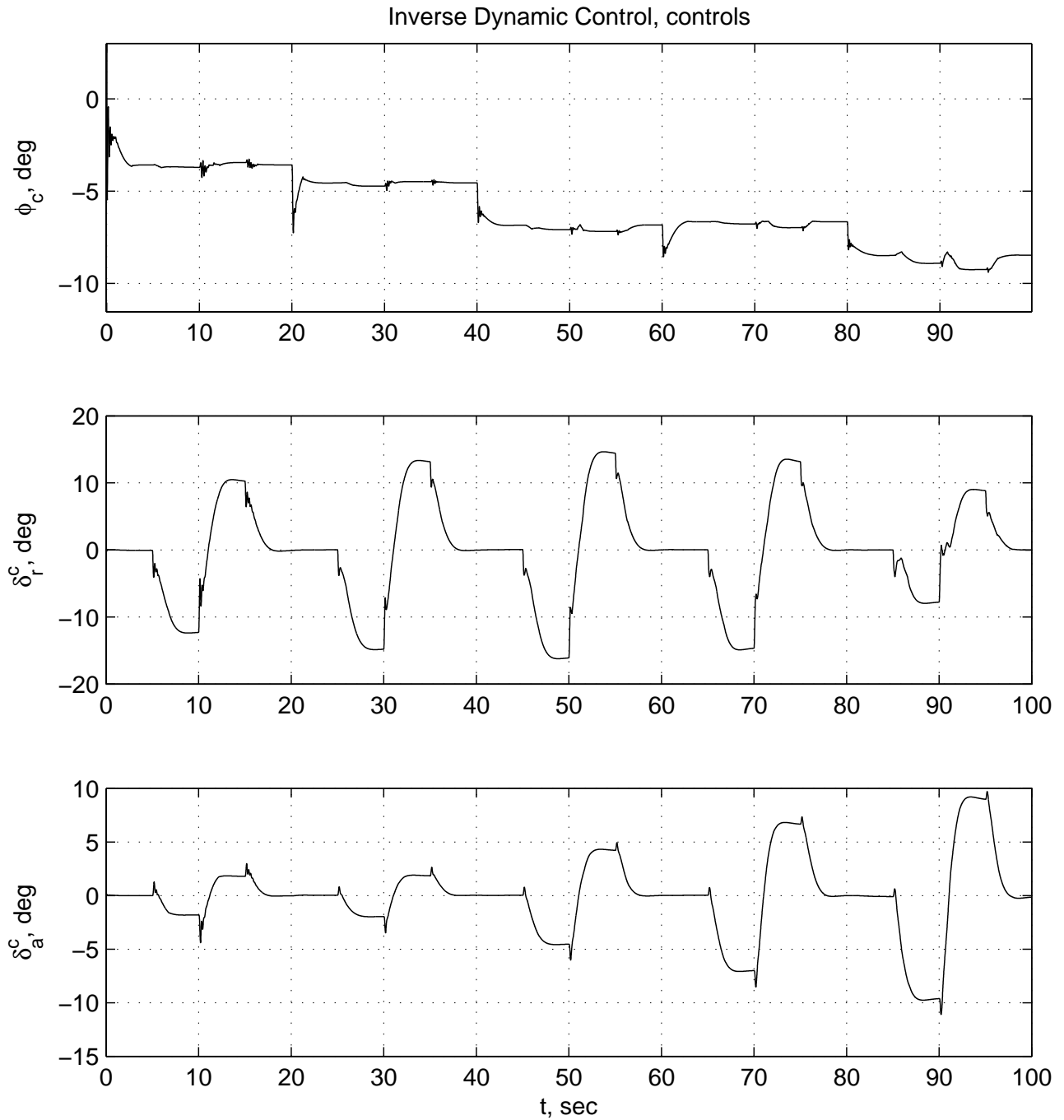


Рис. А.87. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\alpha_{ref} = var$, $\beta_{ref} = 5^0$) — управляющие сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: ϕ_c — командный сигнал для привода управляемого стабилизатора, град; δ_r^c — командный сигнал для привода руля направления, град; δ_a^c — командный сигнал для привода элеронов, град; t — время, с

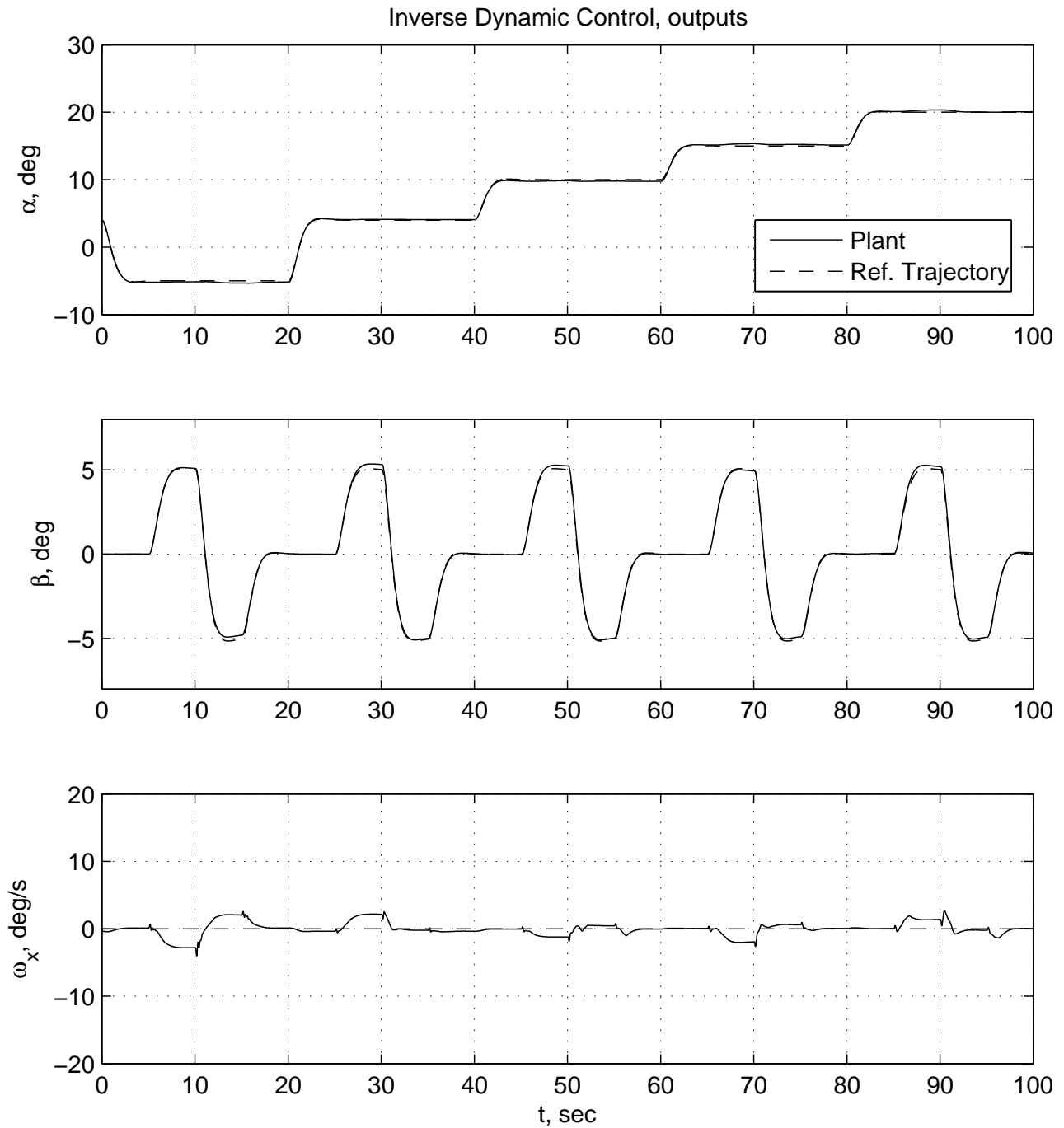


Рис. А.88. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\alpha_{ref} = var, \beta_{ref} = 5^0$) — выходные сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; β — угол скольжения, град; ω_x — угловая скорость крена, град/с; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

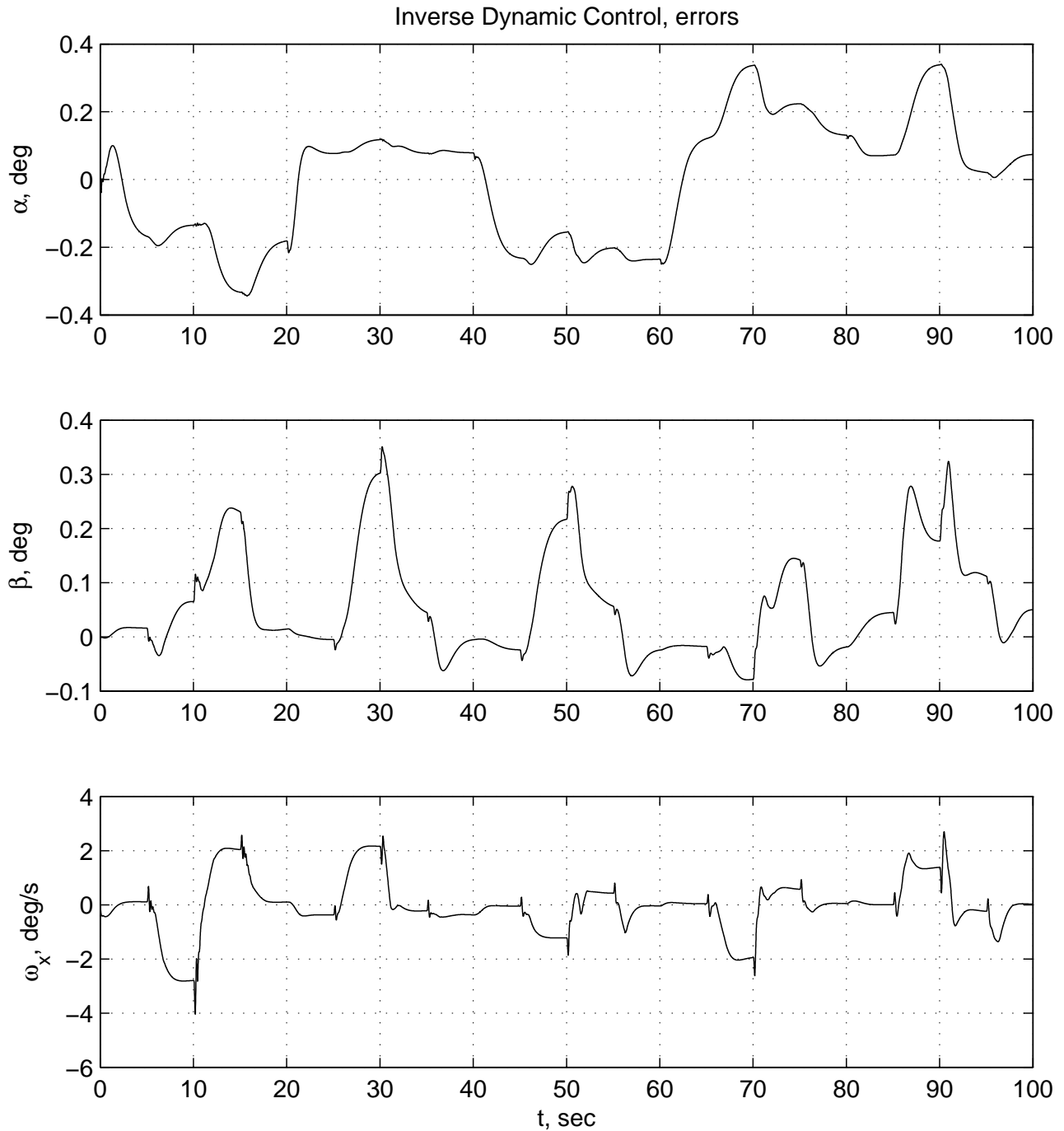


Рис. А.89. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\alpha_{ref} = var$, $\beta_{ref} = 5^0$) — сигналы ошибки (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — ошибка по углу атаки, град; β — ошибка по углу скольжения, град; ω_x — ошибка по угловой скорости крена, град/с; t — время, с

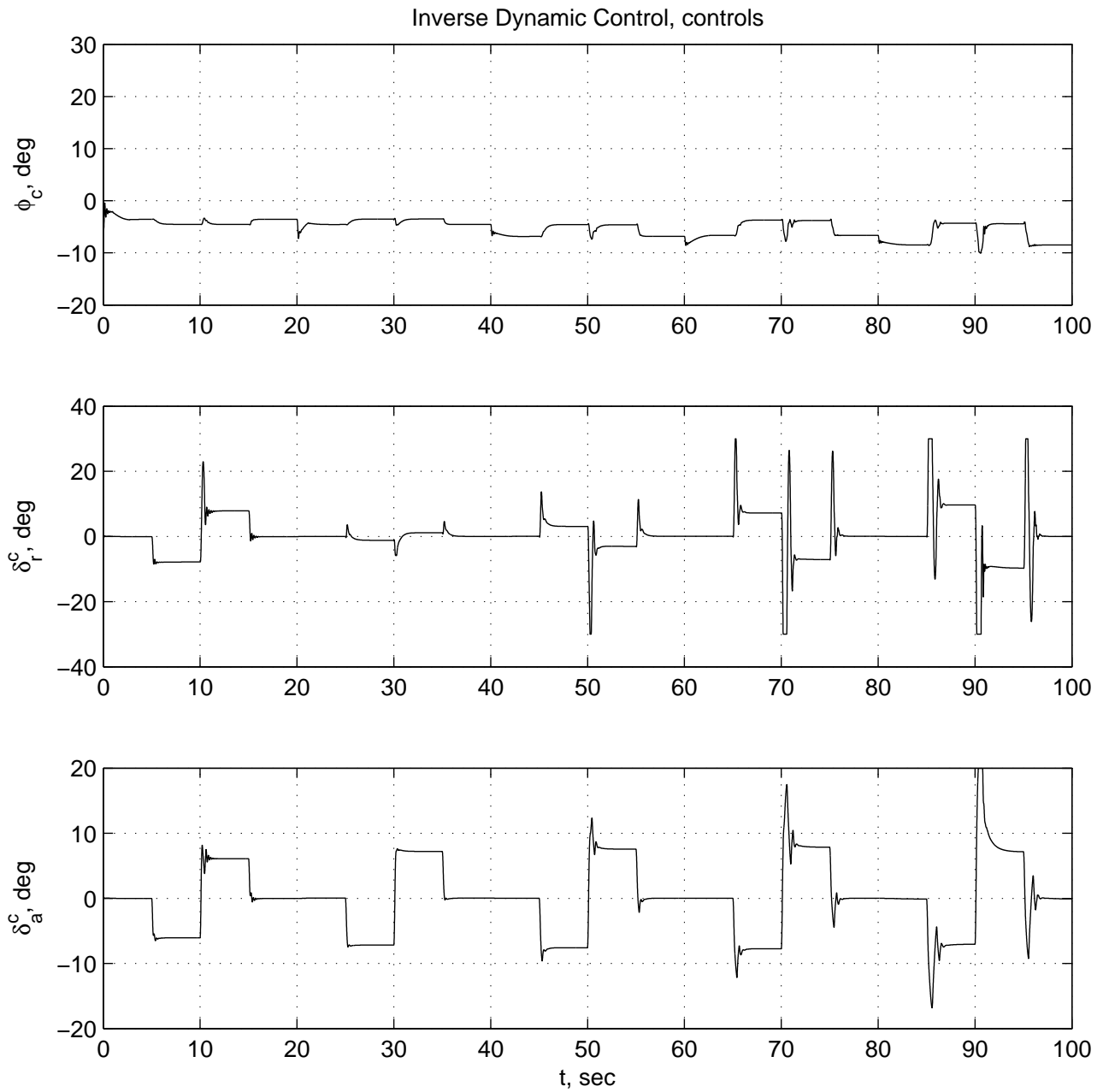


Рис. А.90. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\omega_x = 80$ град/с) — управляющие сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: ϕ_c — командный сигнал для привода управляемого стабилизатора, град; δ_r^c — командный сигнал для привода руля направления, град; δ_a^c — командный сигнал для привода элеронов, град; t — время, с

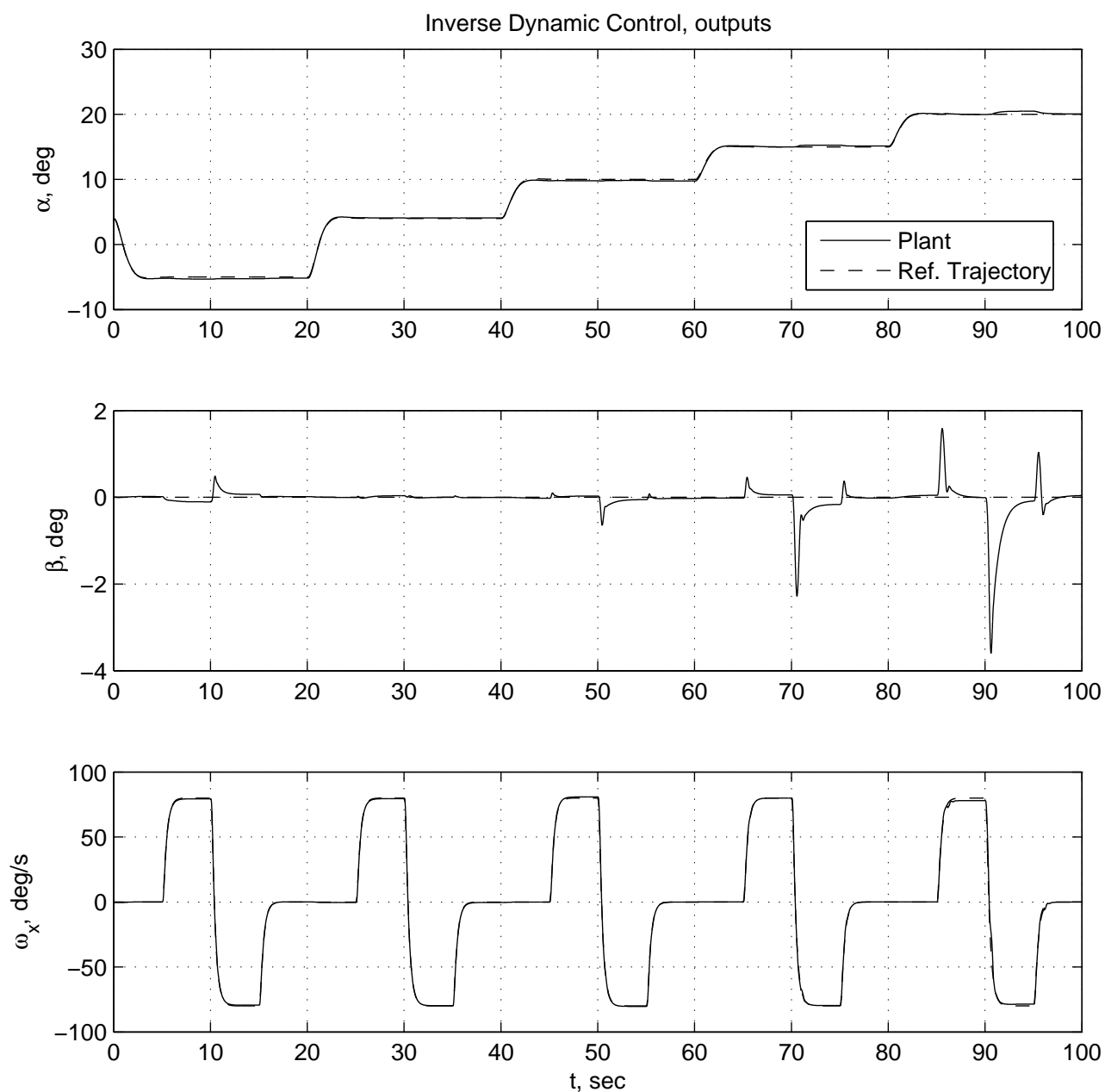


Рис. А.91. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\omega_x = 80$ град/с) — выходные сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; β — угол скольжения, град; ω_x — угловая скорость крена, град/с; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

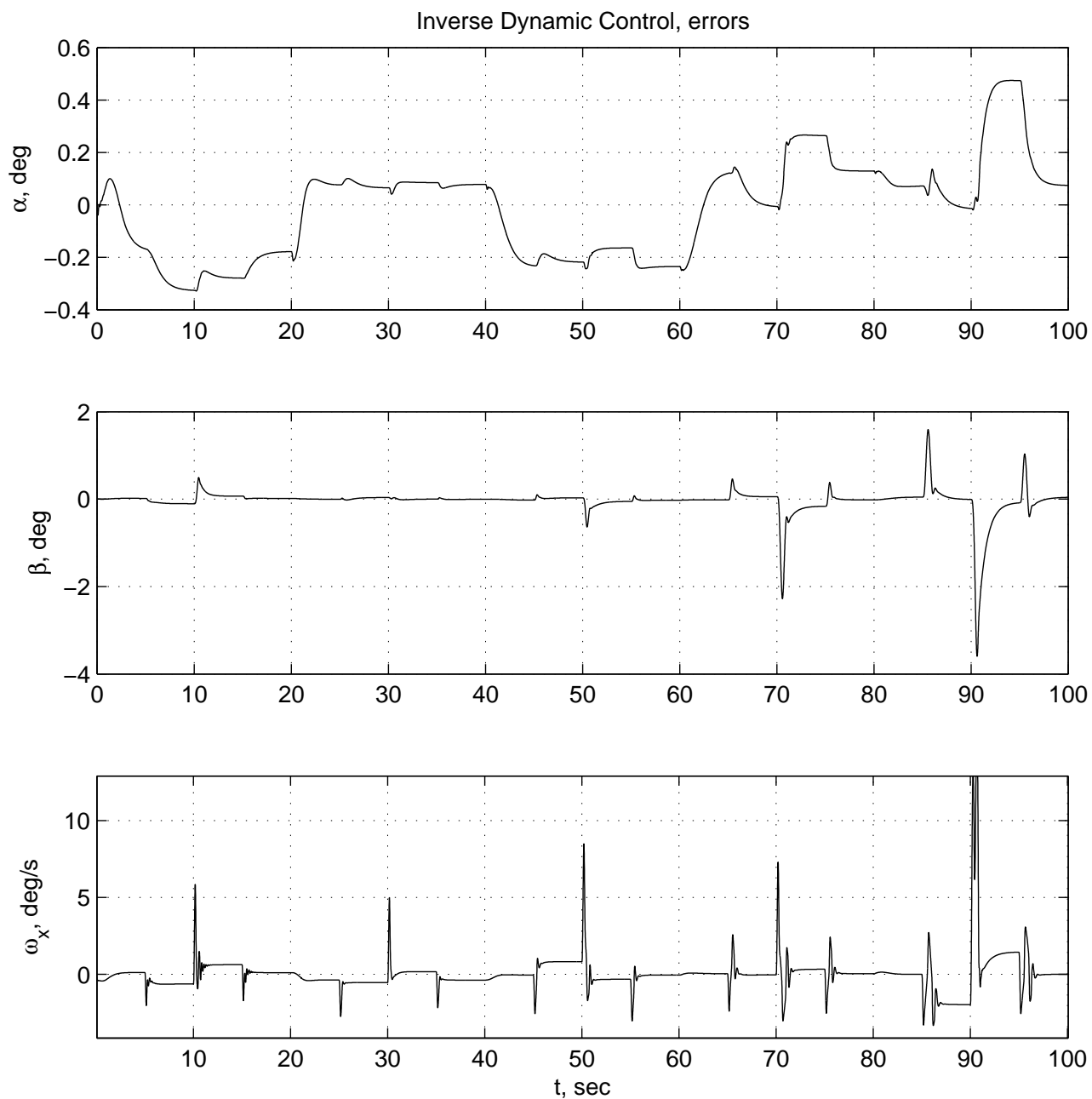


Рис. А.92. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\omega_x = 80$ град/с) — сигналы ошибки (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — ошибка по углу атаки, град; β — ошибка по углу скольжения, град; ω_x — ошибка по угловой скорости крена, град/с; t — время, с

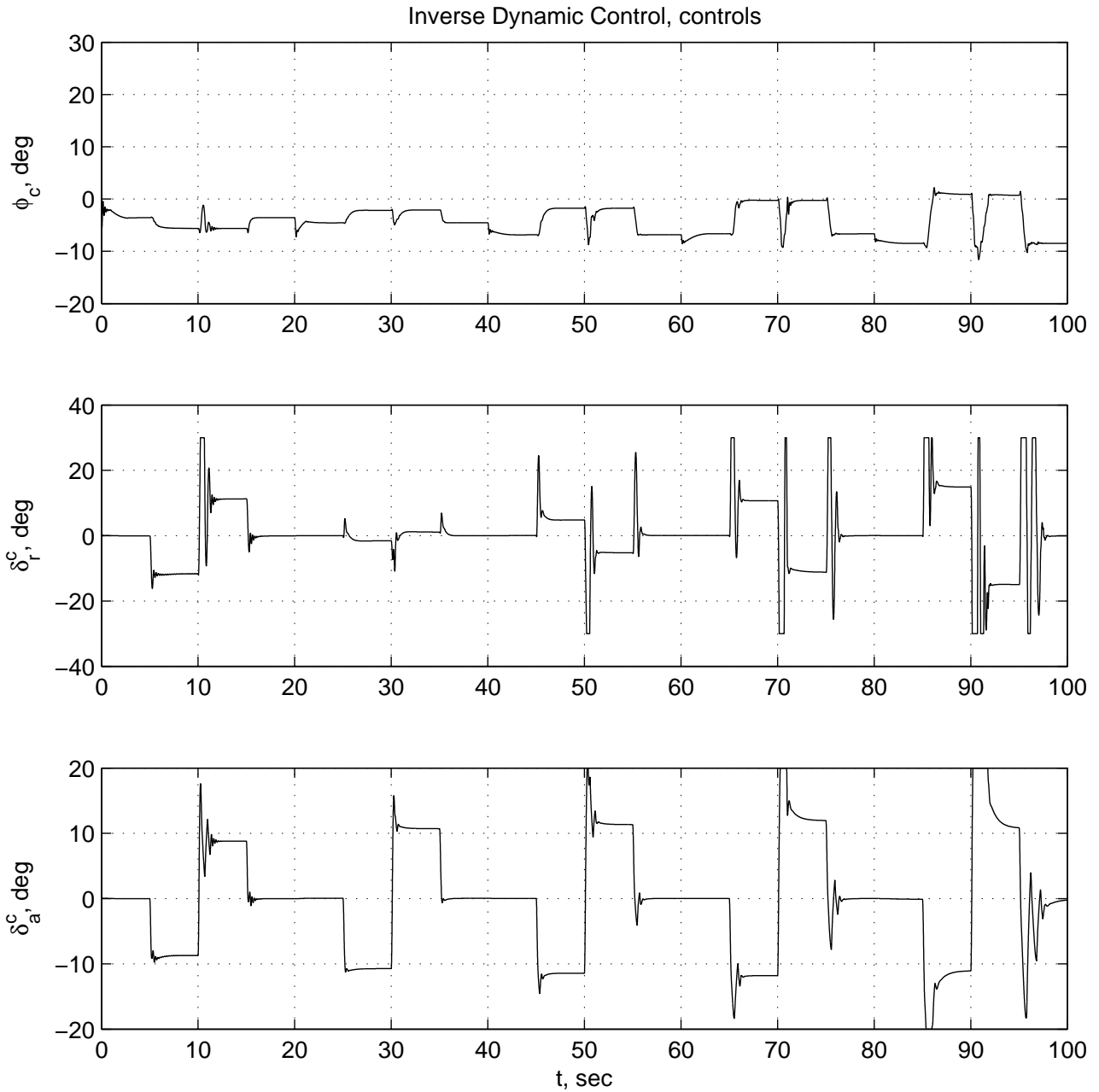


Рис. А.93. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\omega_x = 120$ град/с) — управляющие сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: ϕ_c — командный сигнал для привода управляемого стабилизатора, град; δ_r^c — командный сигнал для привода руля направления, град; δ_a^c — командный сигнал для привода элеронов, град; t — время, с

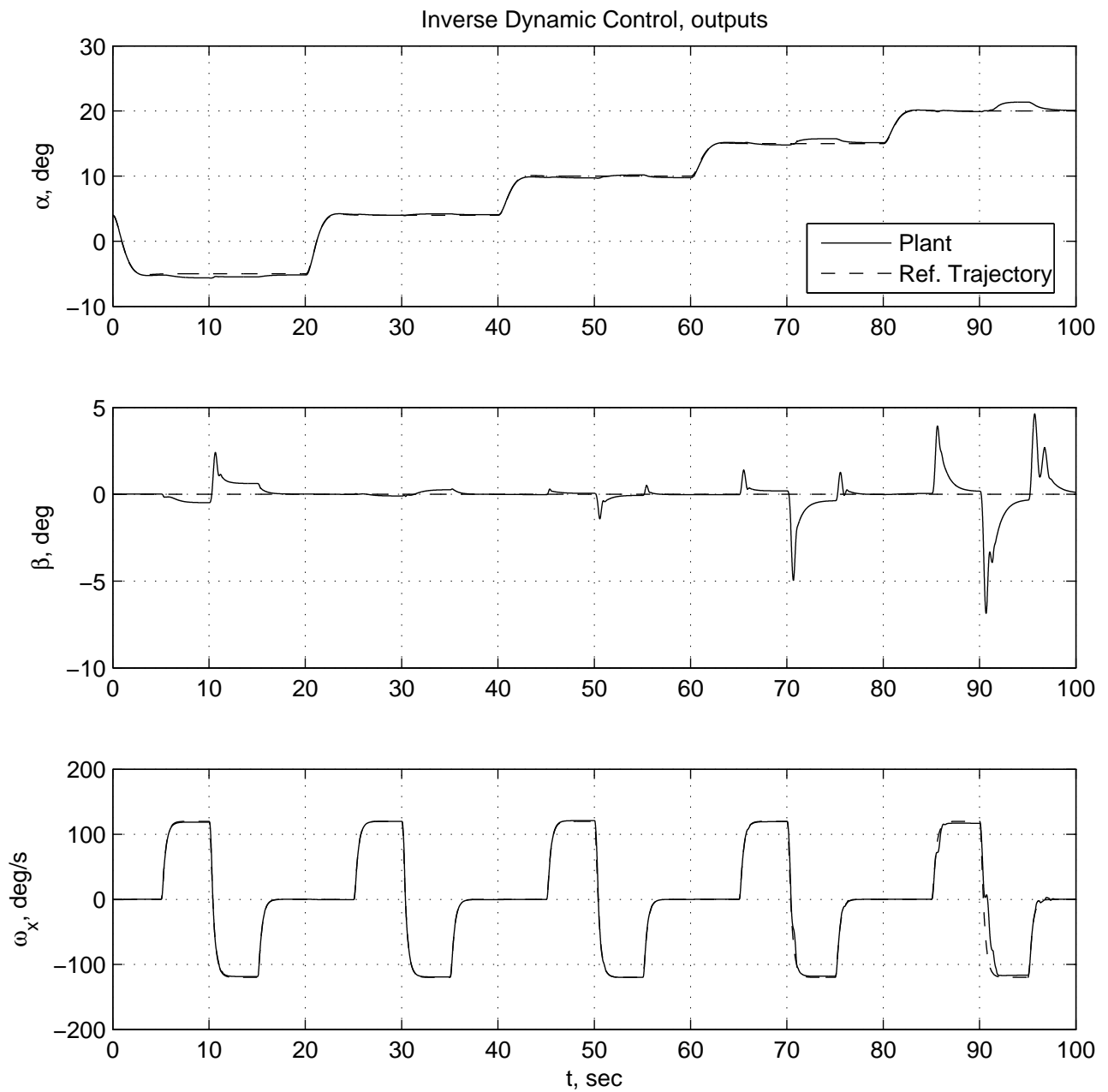


Рис. А.94. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\omega_x = 120$ град/с) — выходные сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; β — угол скольжения, град; ω_x — угловая скорость крена, град/с; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

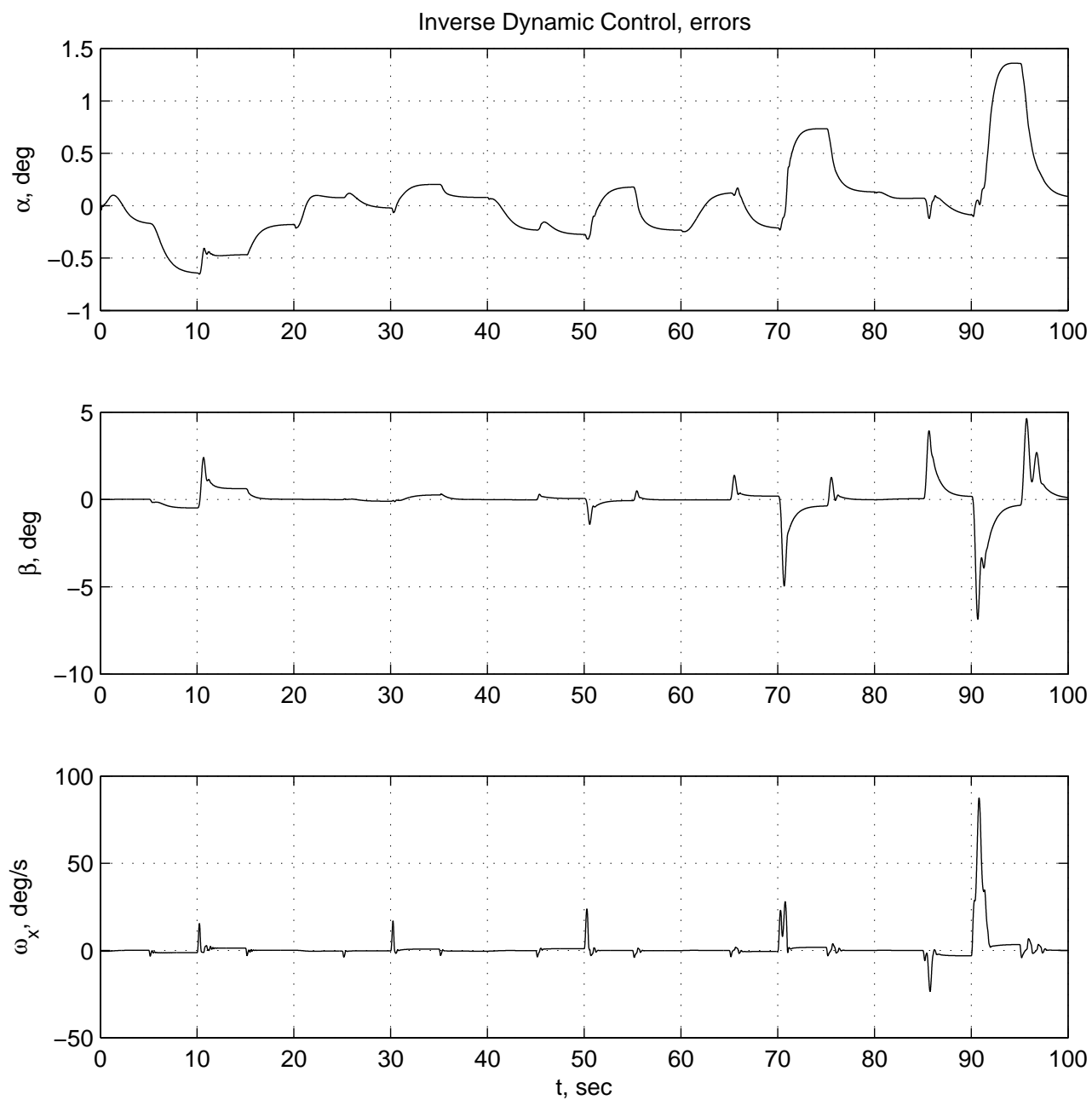


Рис. А.95. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\omega_x = 120$ град/с) — сигналы ошибки (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — ошибка по углу атаки, град; β — ошибка по углу скольжения, град; ω_x — ошибка по угловой скорости крена, град/с; t — время, с

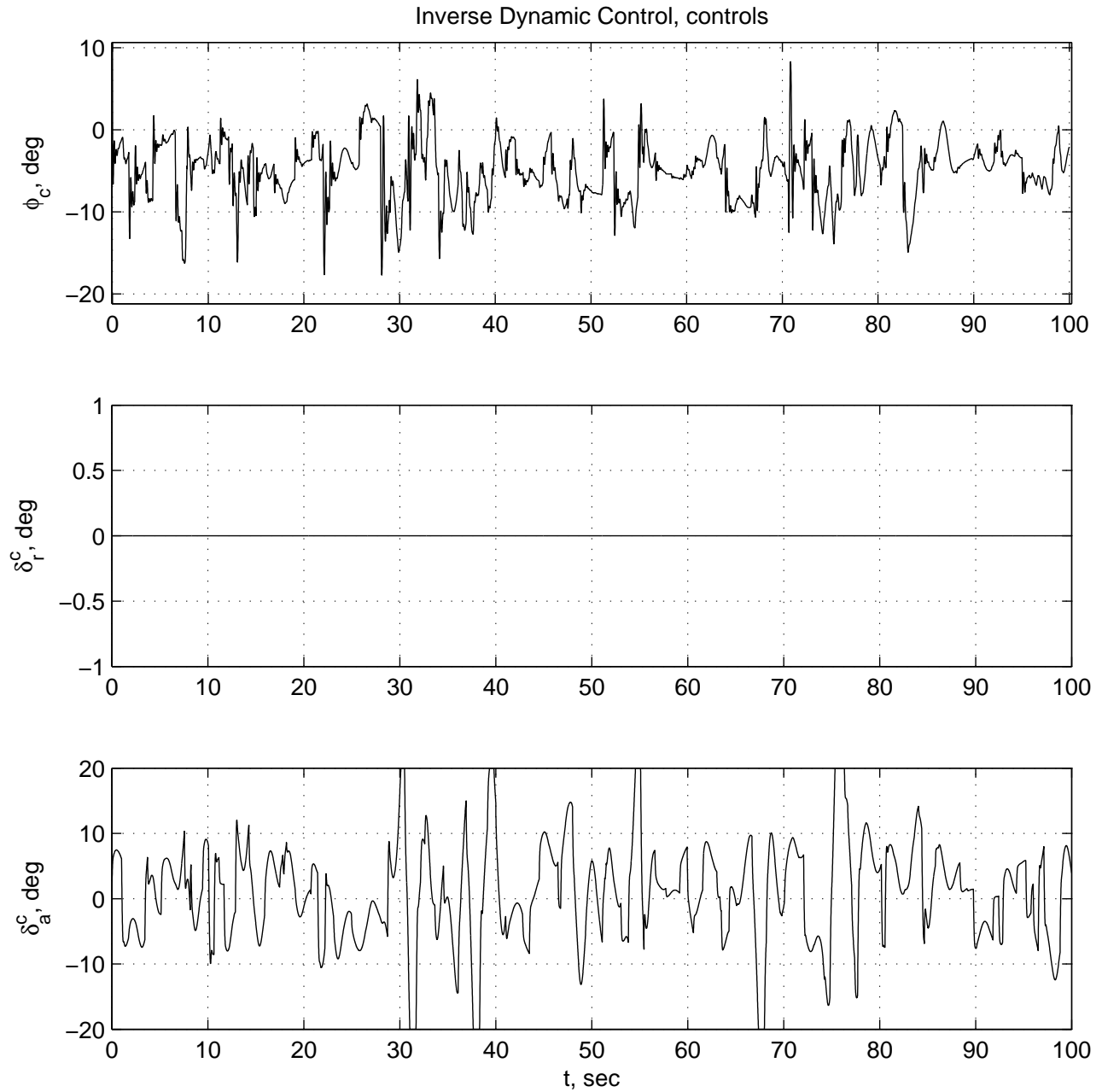


Рис. А.96. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\omega_x = 120$ град/с), компенсация перекрестных связей — управляющие сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: ϕ_c — командный сигнал для привода управляемого стабилизатора, град; δ_r^c — командный сигнал для привода руля направления, град; δ_a^c — командный сигнал для привода элеронов, град; t — время, с

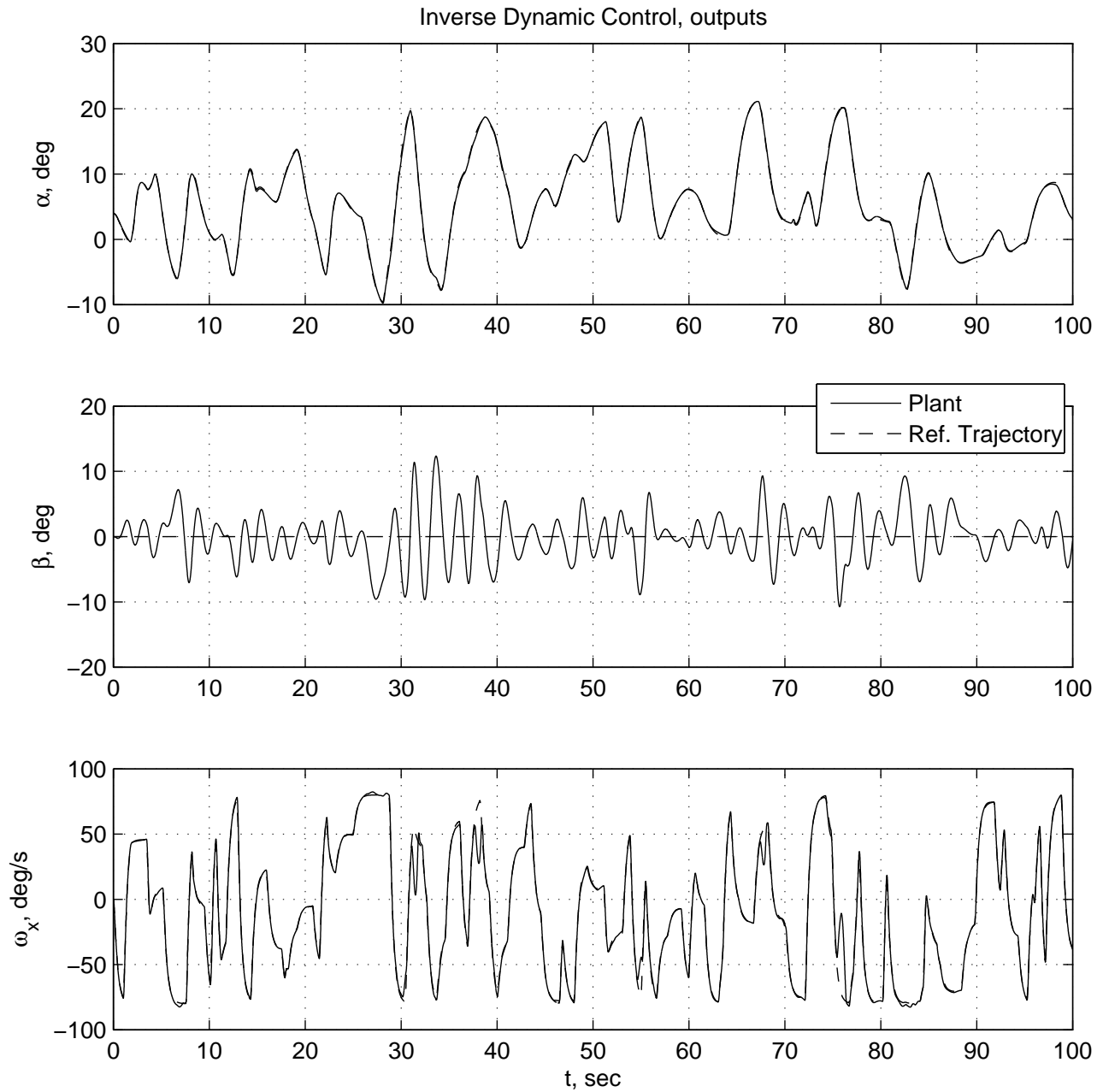


Рис. А.97. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\omega_x = 120$ град/с), компенсация перекрестных связей — выходные сигналы (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — угол атаки, град; β — угол скольжения, град; ω_x — угловая скорость крена, град/с; t — время, с; Plant — объект управления; Ref. Trajectory — эталонная траектория

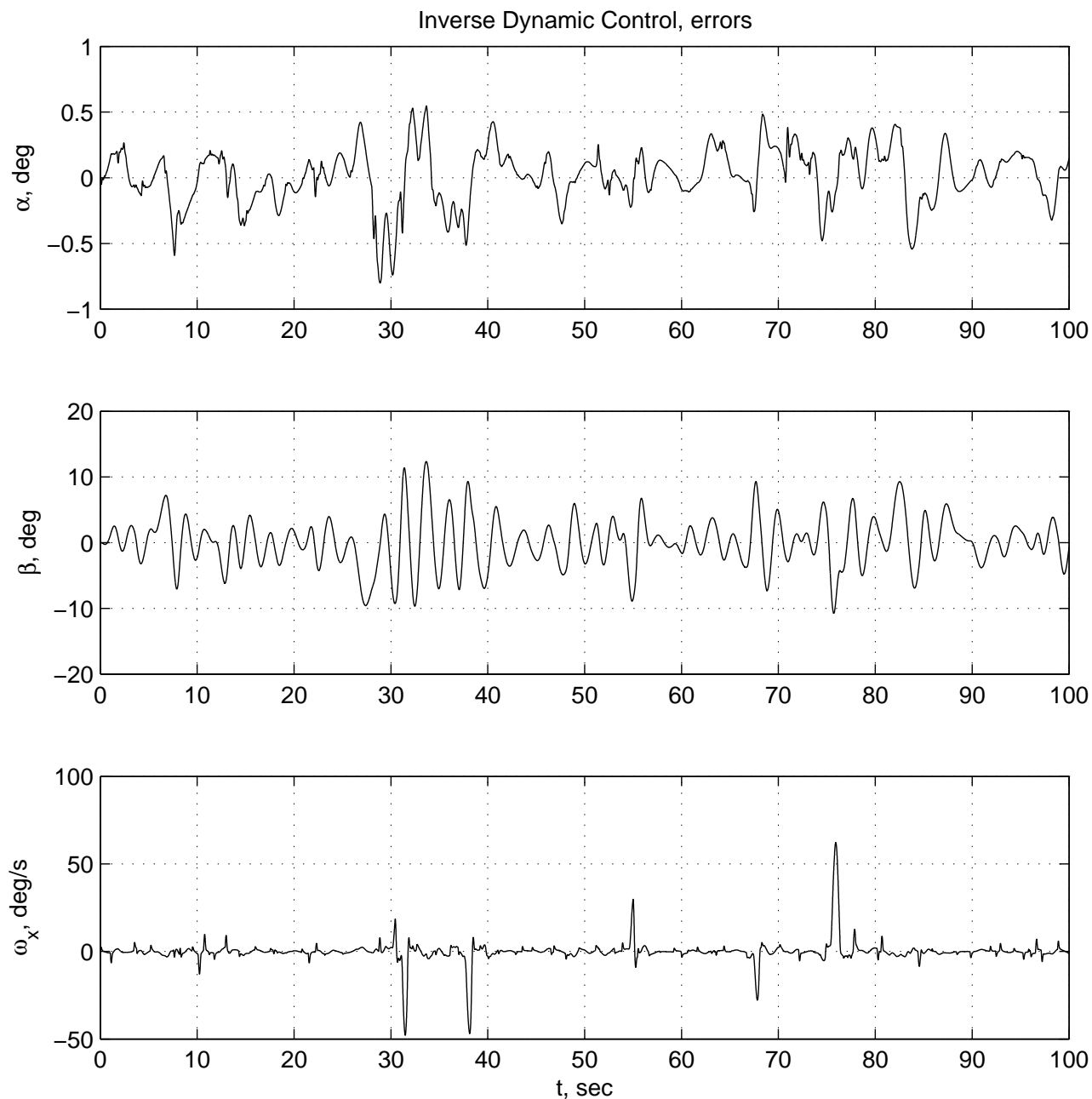


Рис. А.98. Результаты вычислительного эксперимента для системы управления с обратной динамикой с использованием декомпозиции ($\omega_x = 120$ град/с), компенсация перекрестных связей — сигналы ошибки (режим полета $V = 500$ км/ч, $H = 100$ м). Обозначения: α — ошибка по углу атаки, град; β — ошибка по углу скольжения, град; ω_x — ошибка по угловой скорости крена, град/с; t — время, с

А.6. Влияние атмосферной турбулентности на работоспособность адаптивной системы управления продольным движением самолета

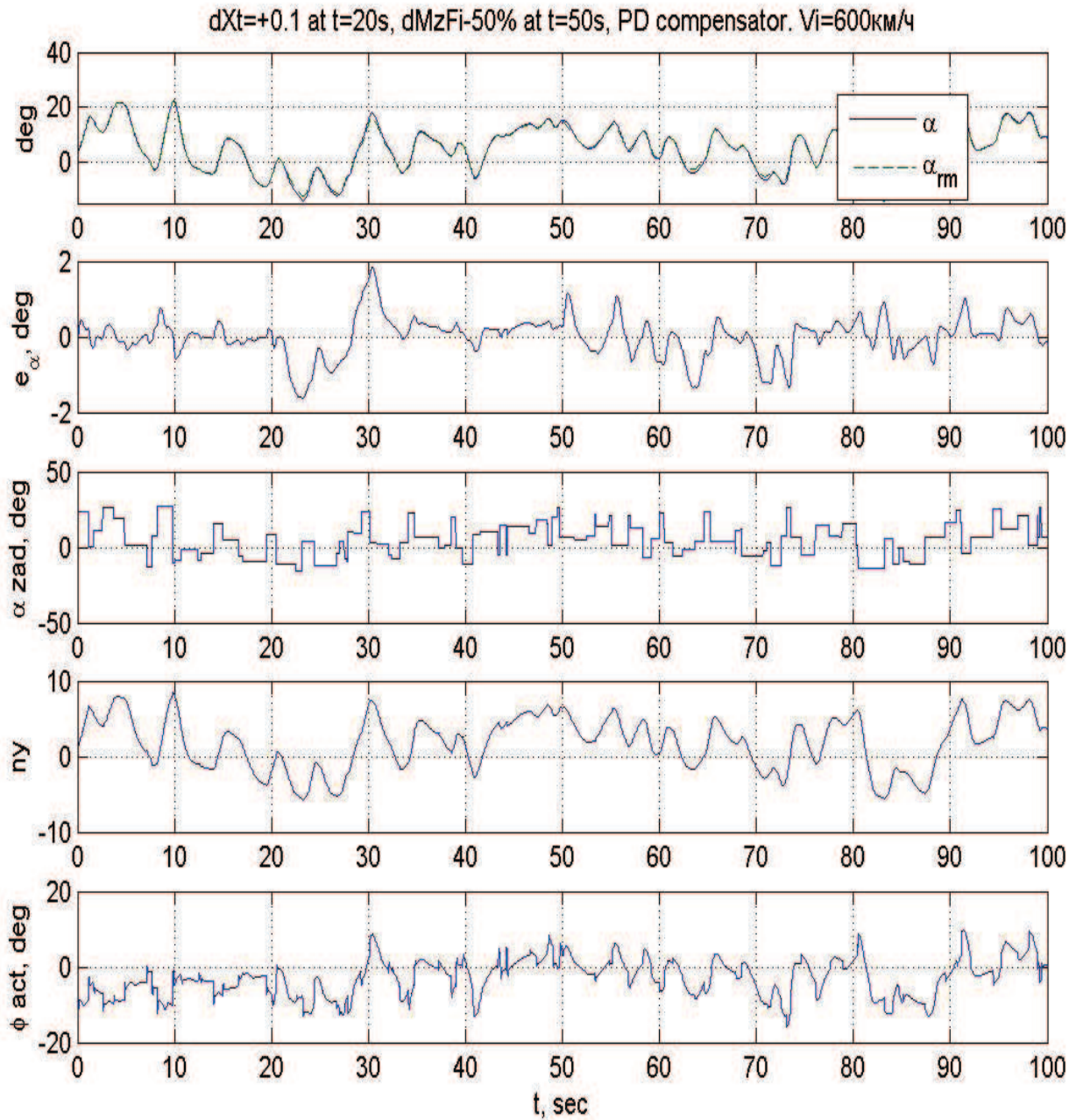


Рис. А.99. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие отсутствует. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_{α} — ошибка отслеживания заданного угла атаки, град; α_{zad} — задающий сигнал по углу атаки, град; n_y — нормальная перегрузка; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

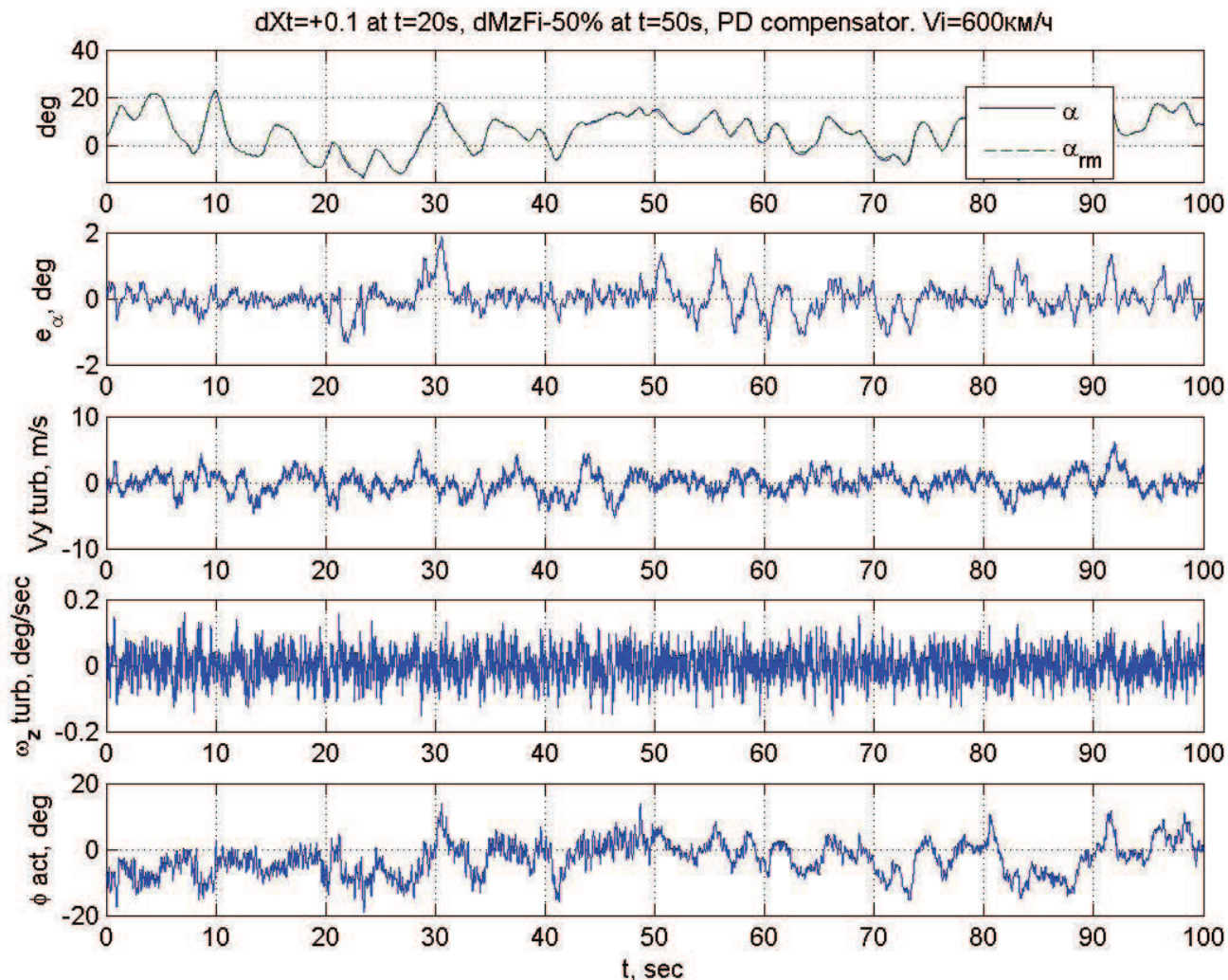


Рис. А.100. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие: $V_{y,turb}$ в диапазоне ± 10 м/с, $\omega_{z,turb}$ в диапазоне ± 0.2 град/с. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $V_{y,turb}$ — поступательная (вертикальная) составляющая атмосферного возмущения; $\omega_{z,turb}$ — вращательная составляющая атмосферного возмущения; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

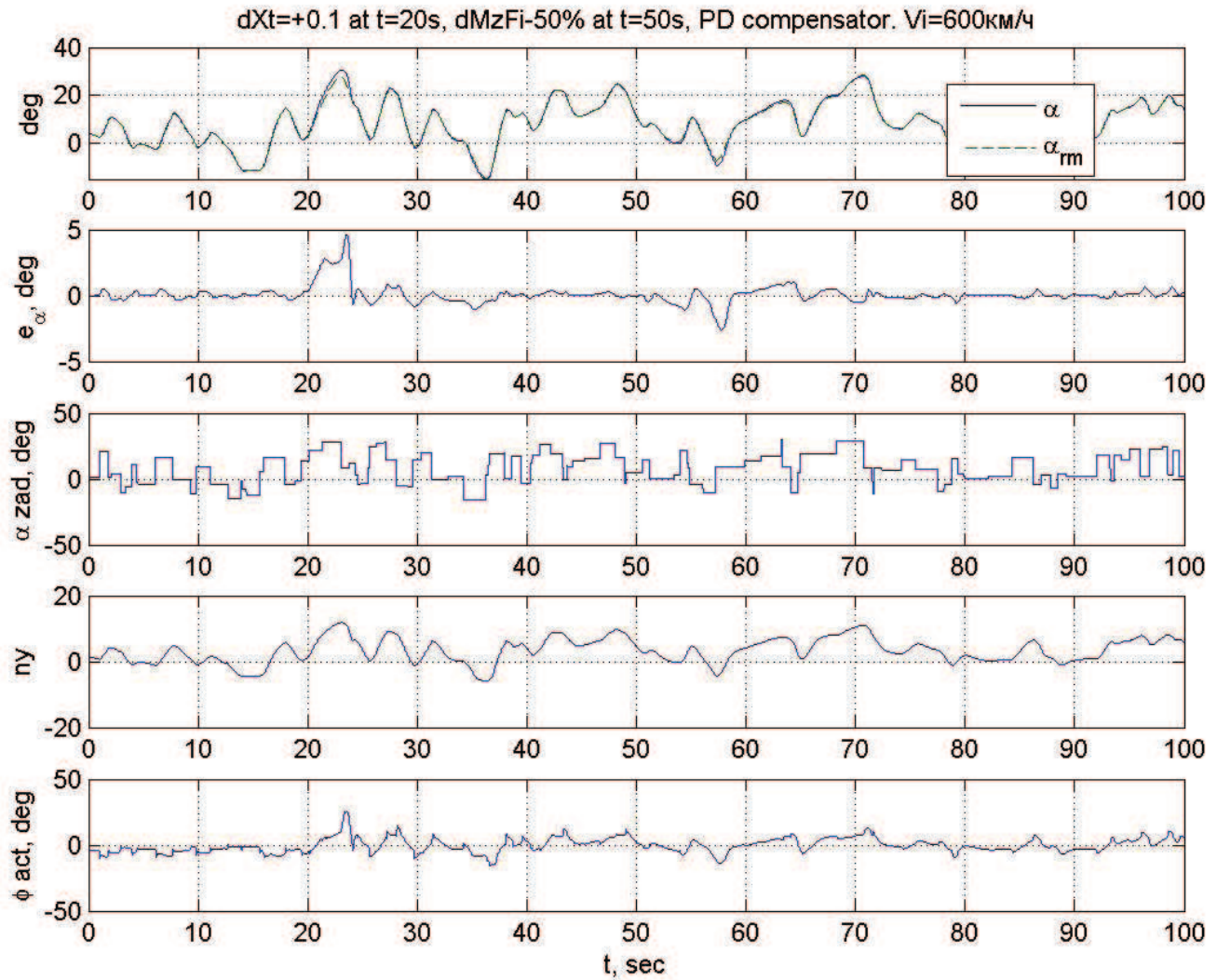


Рис. А.101. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие отсутствует. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_α — ошибка отслеживания заданного угла атаки, град; α_{zad} — задающий сигнал по углу атаки, град; n_y — нормальная перегрузка; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

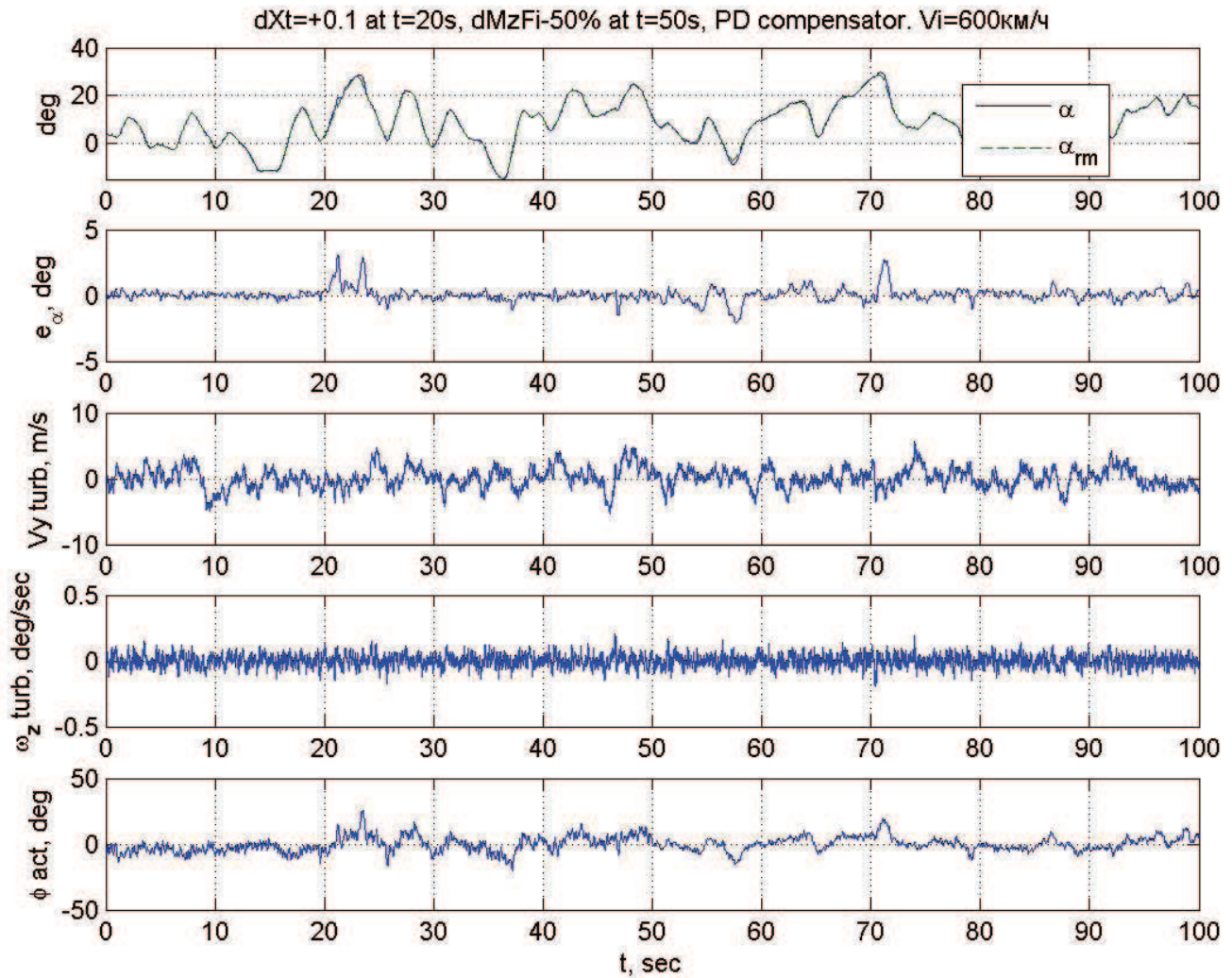


Рис. А.102. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие: $V_{y,turb}$ в диапазоне ± 10 м/с, $\omega_{z,turb}$ в диапазоне ± 0.2 град/с. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $V_{y,turb}$ — поступательная (вертикальная) составляющая атмосферного возмущения; $\omega_{z,turb}$ — вращательная составляющая атмосферного возмущения; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

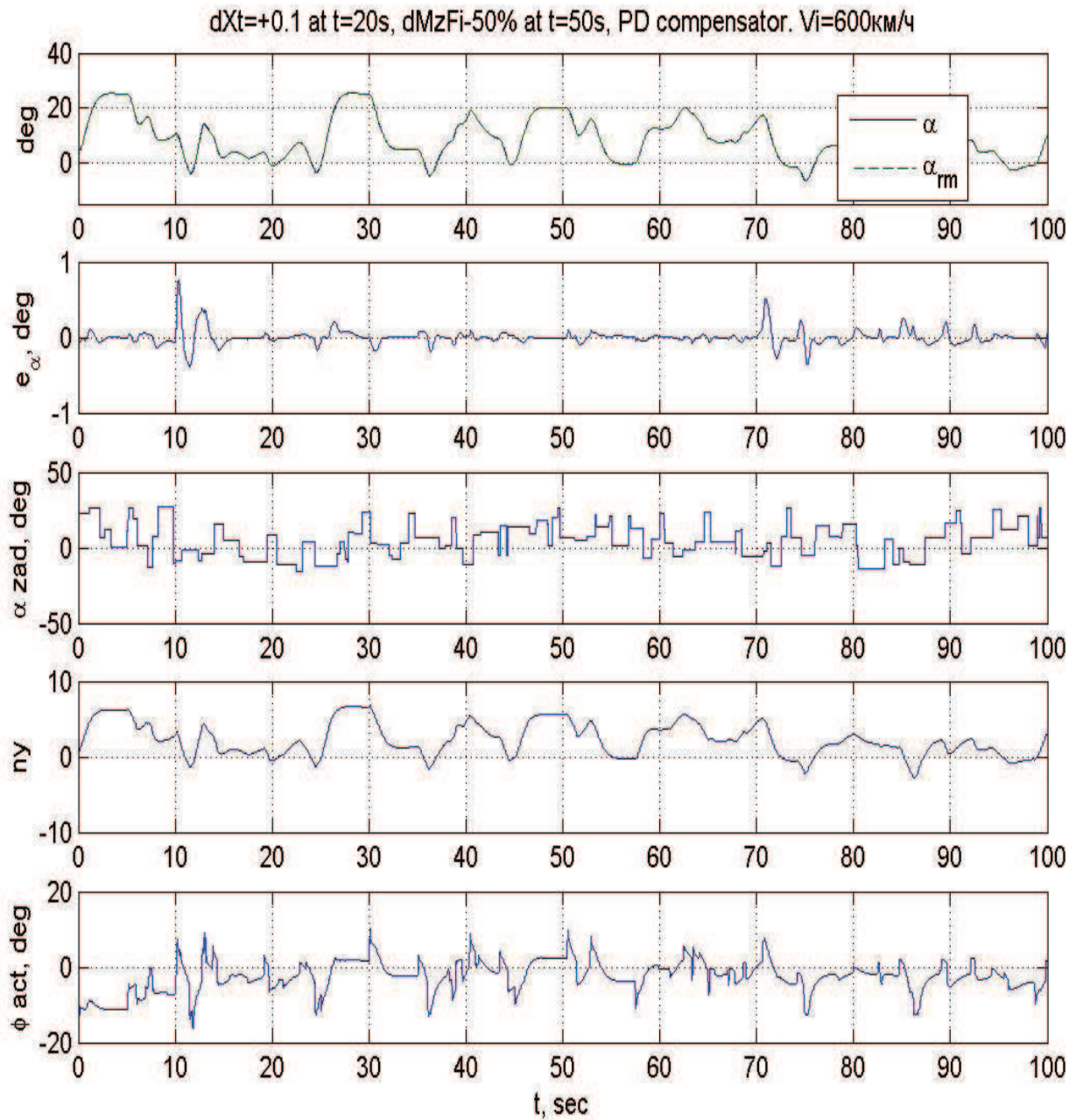


Рис. А.103. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие отсутствует. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_{α} — ошибка отслеживания заданного угла атаки, град; α_{zad} — задающий сигнал по углу атаки, град; n_y — нормальная перегрузка; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

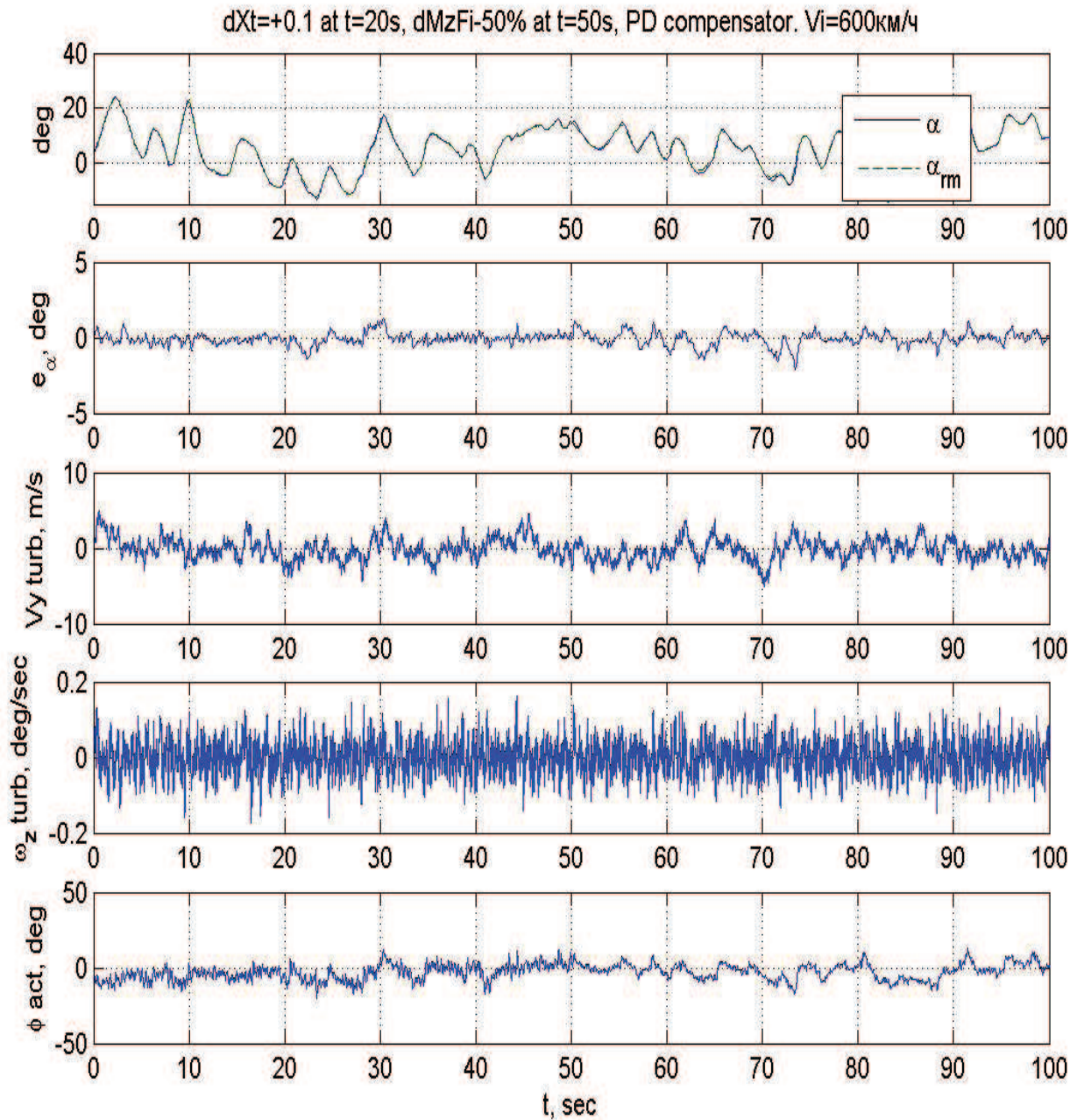


Рис. А.104. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие: $V_{y,turb}$ в диапазоне ± 10 м/с, $\omega_{z,turb}$ в диапазоне ± 0.2 град/с. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_α — ошибка отслеживания заданного угла атаки, град; $V_{y,turb}$ — поступательная (вертикальная) составляющая атмосферного возмущения; $\omega_{z,turb}$ — вращательная составляющая атмосферного возмущения; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

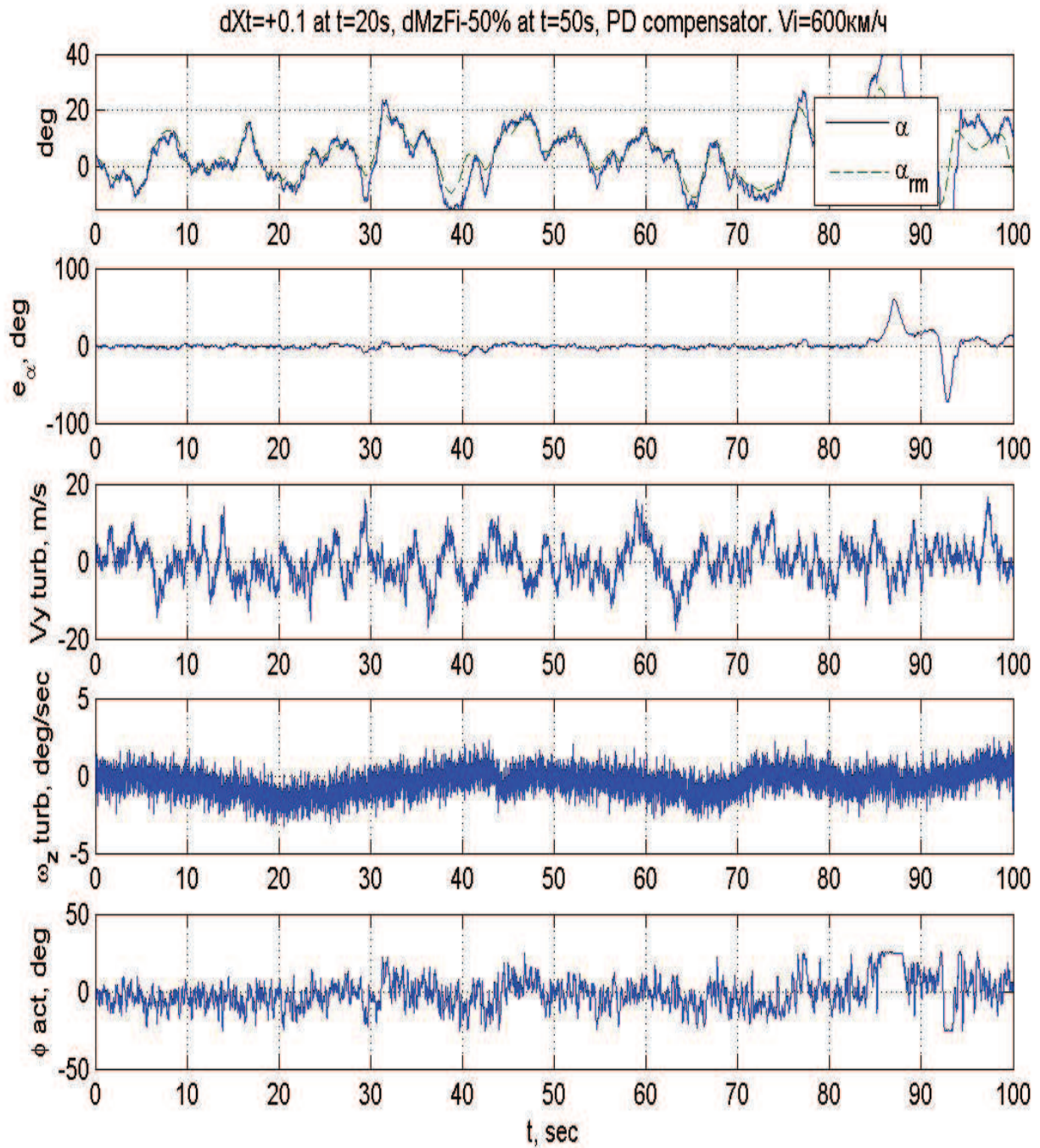


Рис. А.105. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие отсутствует. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_{α} — ошибка отслеживания заданного угла атаки, град; α_{zad} — задающий сигнал по углу атаки, град; n_y — нормальная перегрузка; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

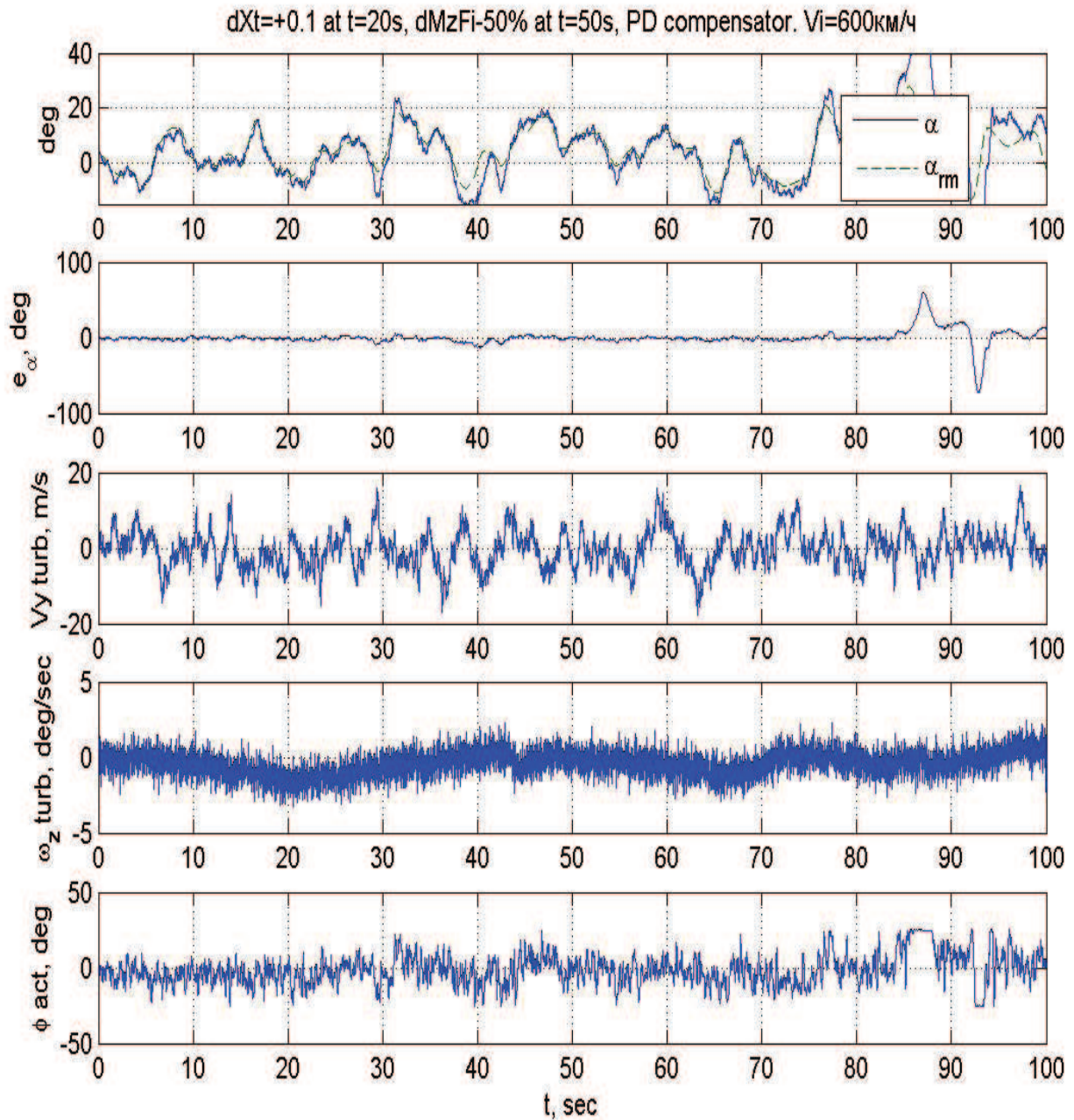


Рис. А.106. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие: $V_{y,turb}$ в диапазоне ± 20 м/с, $\omega_{z,turb}$ в диапазоне ± 2 град/с. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_α — ошибка отслеживания заданного угла атаки, град; $V_{y,turb}$ — поступательная (вертикальная) составляющая атмосферного возмущения; $\omega_{z,turb}$ — вращательная составляющая атмосферного возмущения; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

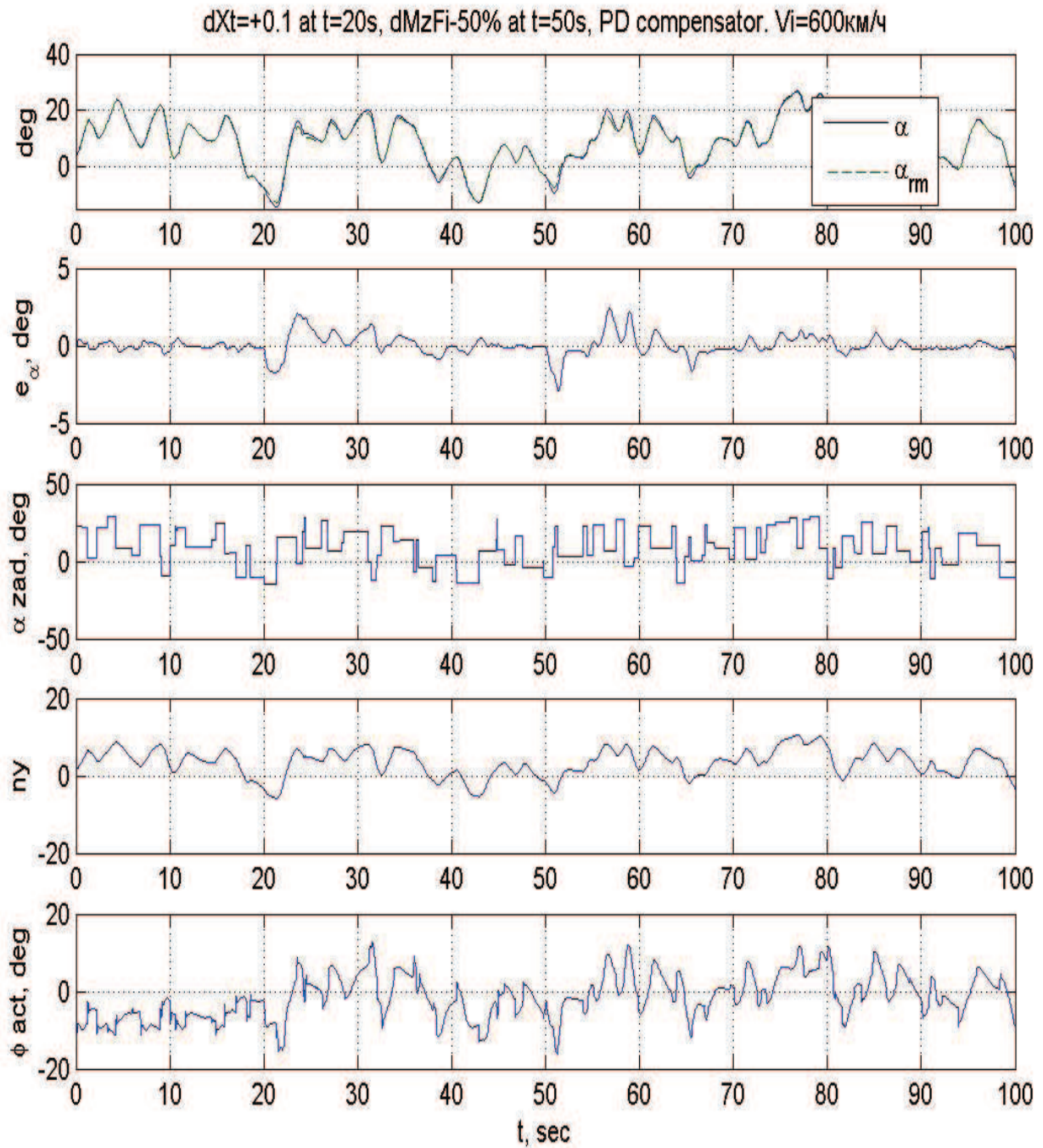


Рис. А.107. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие отсутствует. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_α — ошибка отслеживания заданного угла атаки, град; α_{zad} — задающий сигнал по углу атаки, град; n_y — нормальная перегрузка; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

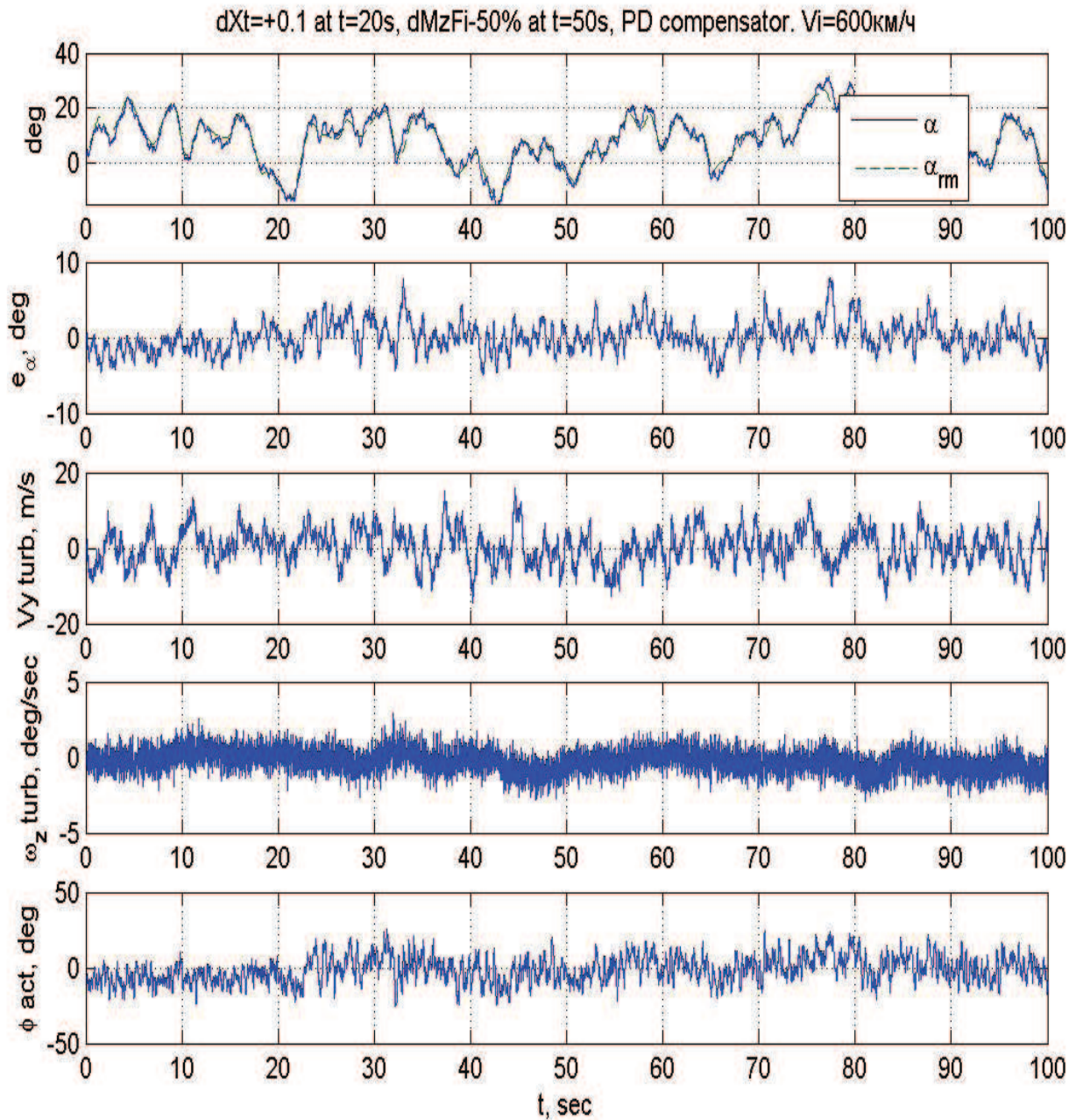


Рис. А.108. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие: $V_{y,turb}$ в диапазоне ± 20 м/с, $\omega_{z,turb}$ в диапазоне ± 2 град/с. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_α — ошибка отслеживания заданного угла атаки, град; $V_{y,turb}$ — поступательная (вертикальная) составляющая атмосферного возмущения; $\omega_{z,turb}$ — вращательная составляющая атмосферного возмущения; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

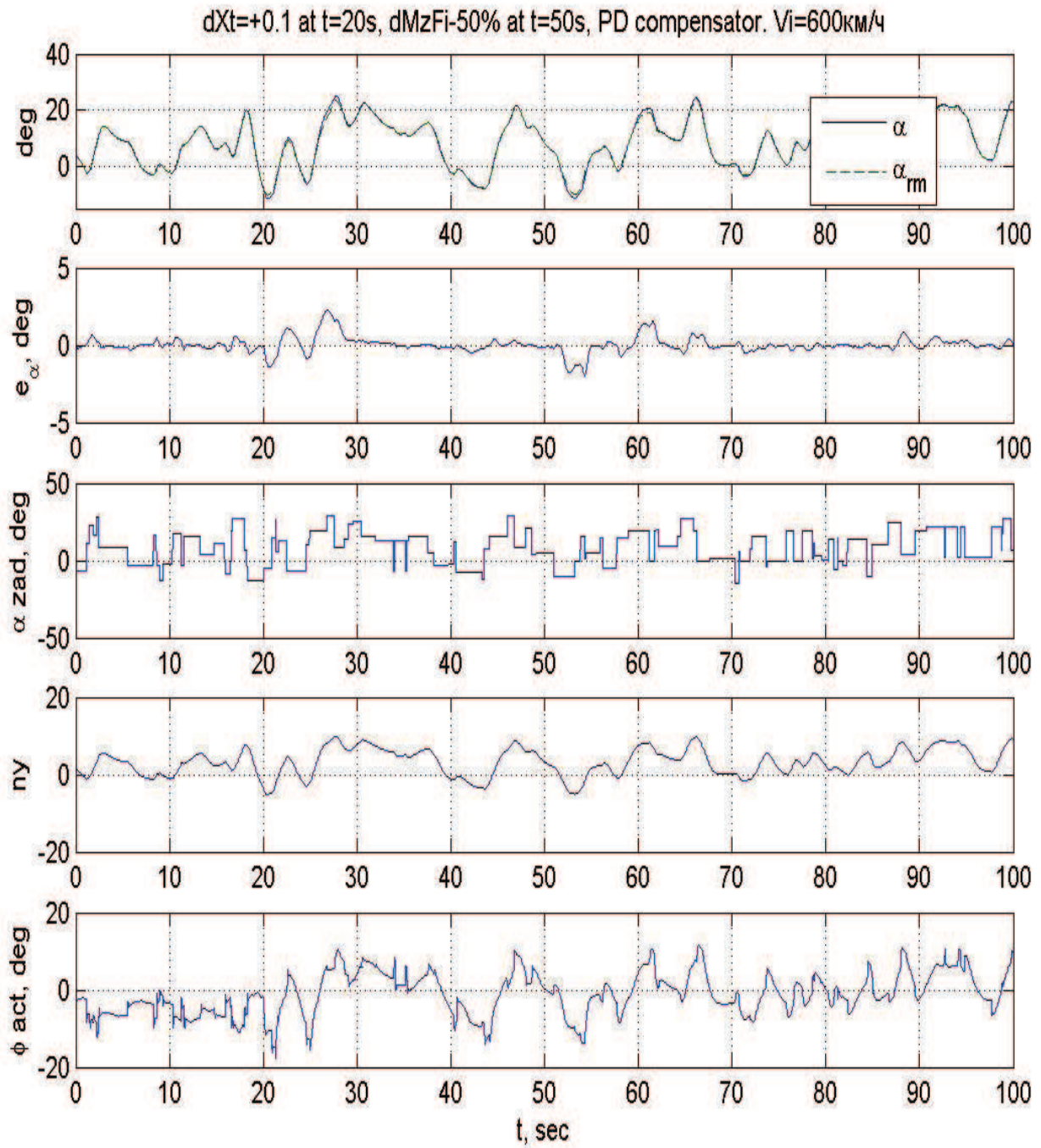


Рис. А.109. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие отсутствует. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_α — ошибка отслеживания заданного угла атаки, град; α_{zad} — задающий сигнал по углу атаки, град; n_y — нормальная перегрузка; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

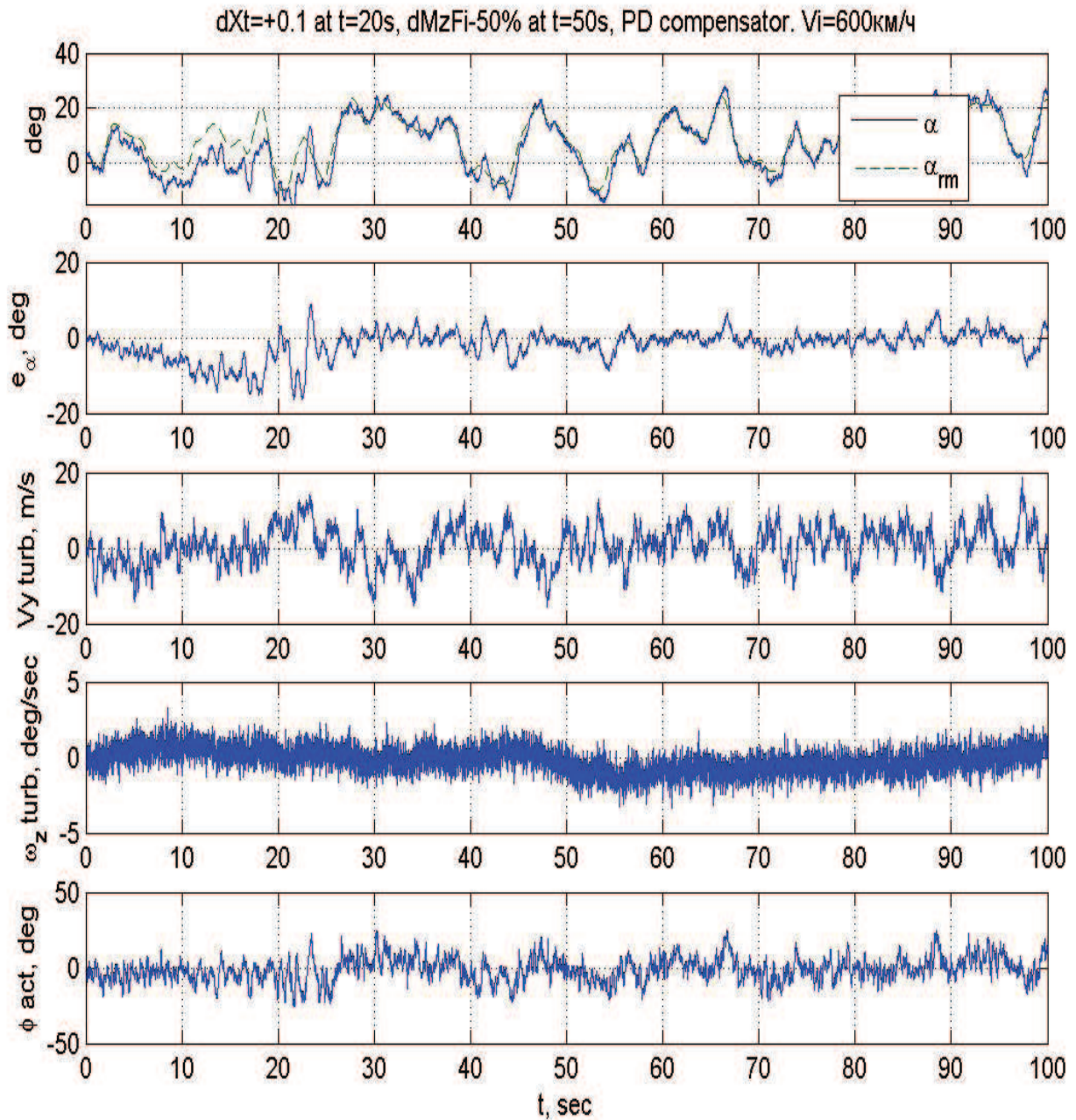


Рис. А.110. Результаты вычислительного эксперимента по оценке характера воздействия на управляемую систему атмосферной турбулентности: адаптивная система управления с эталонной моделью. Возмущающее воздействие: $V_{y,turb}$ в диапазоне ± 20 м/с, $\omega_{z,turb}$ в диапазоне ± 2 град/с. Режим полета $H = 100$ м, $V = 600$ км/ч. Обозначения: α — угол атаки, град; α_{rm} — угол атаки на выходе эталонной модели, град; e_α — ошибка отслеживания заданного угла атаки, град; $V_{y,turb}$ — поступательная (вертикальная) составляющая атмосферного возмущения; $\omega_{z,turb}$ — вращательная составляющая атмосферного возмущения; ϕ_{act} — командный сигнал для привода управляемого стабилизатора, град; t — время, с

А.7. Адаптация к неопределенности в исходных данных

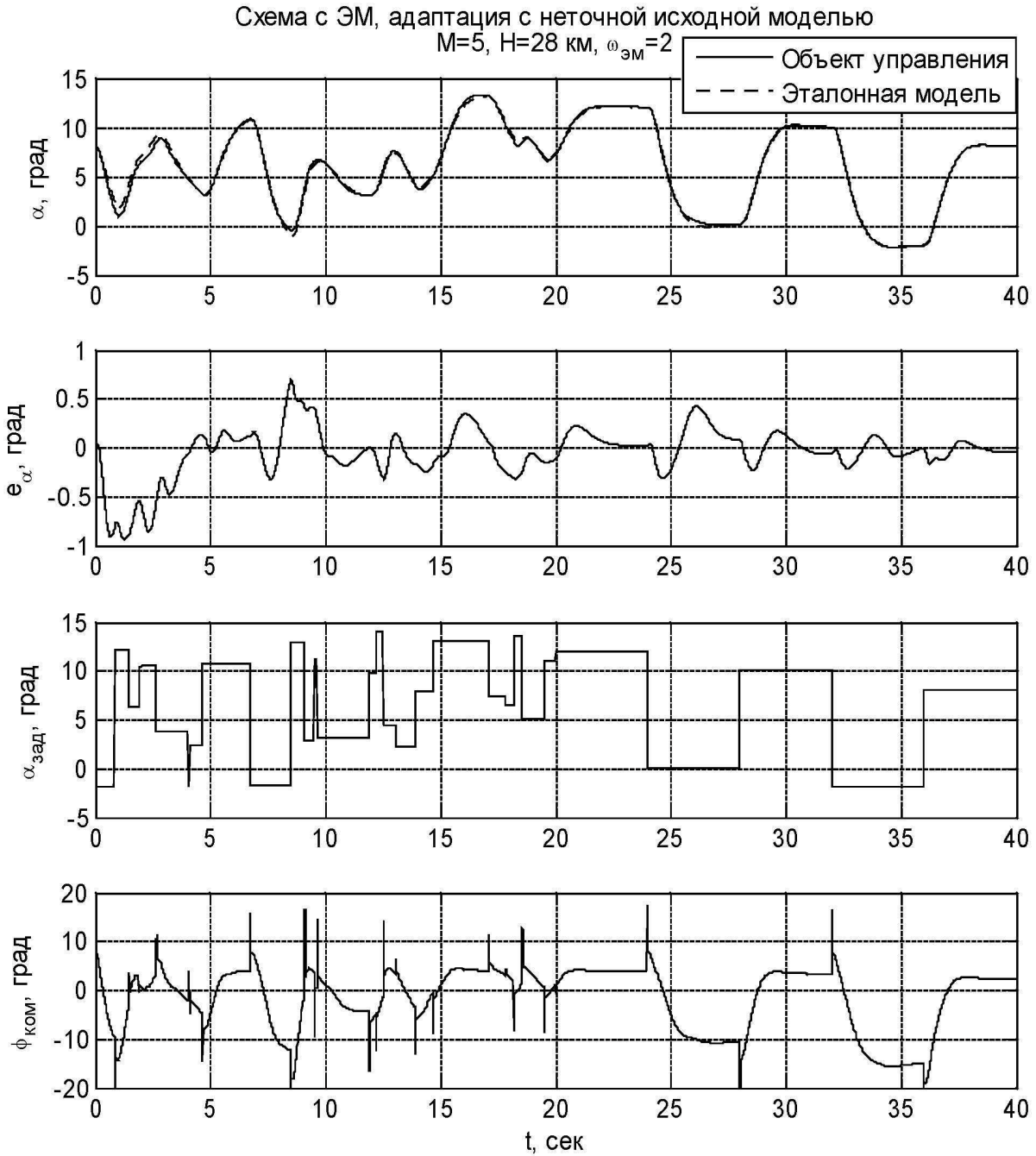


Рис. А.111. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 5, H = 28 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

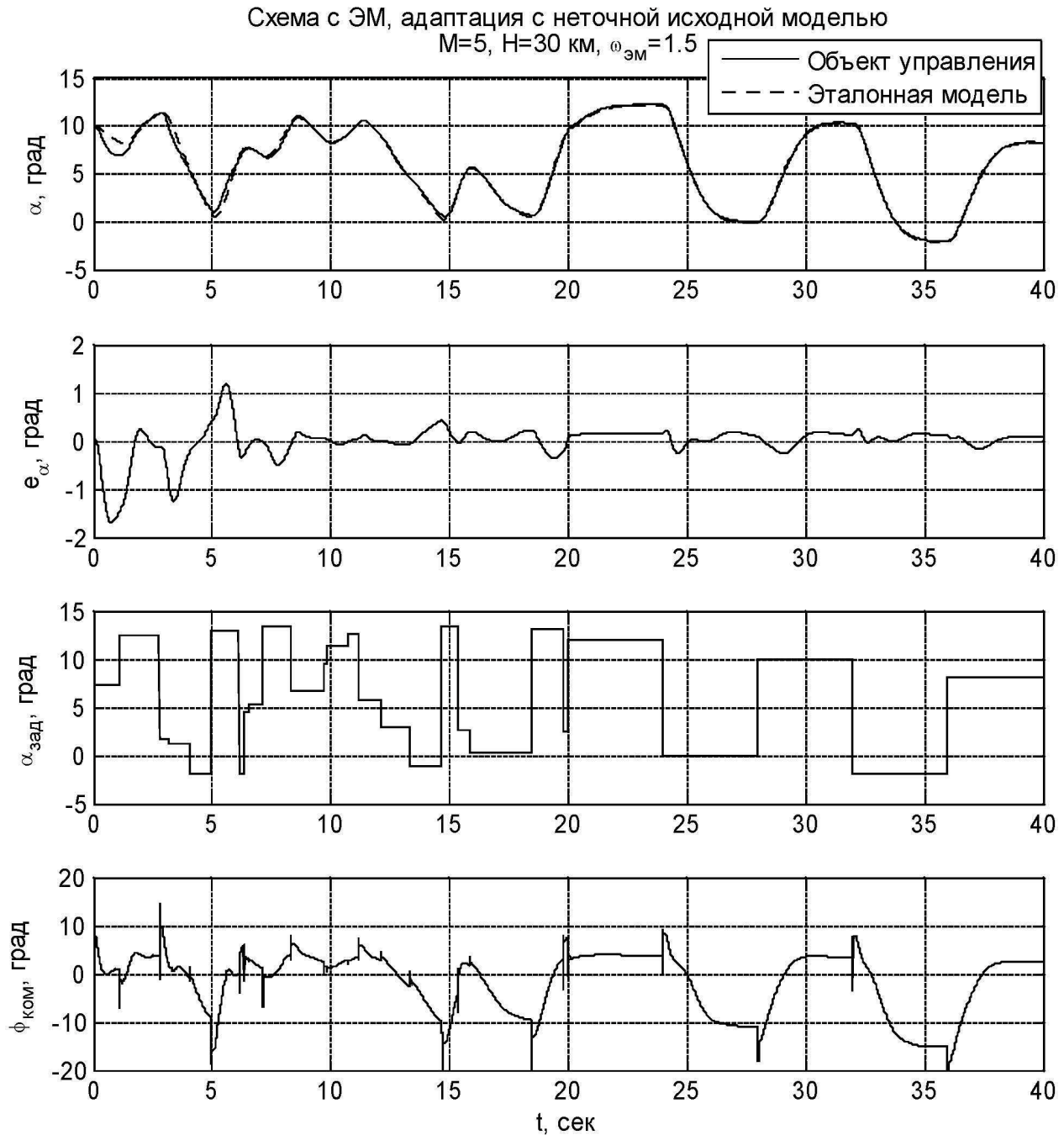


Рис. А.112. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 5, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

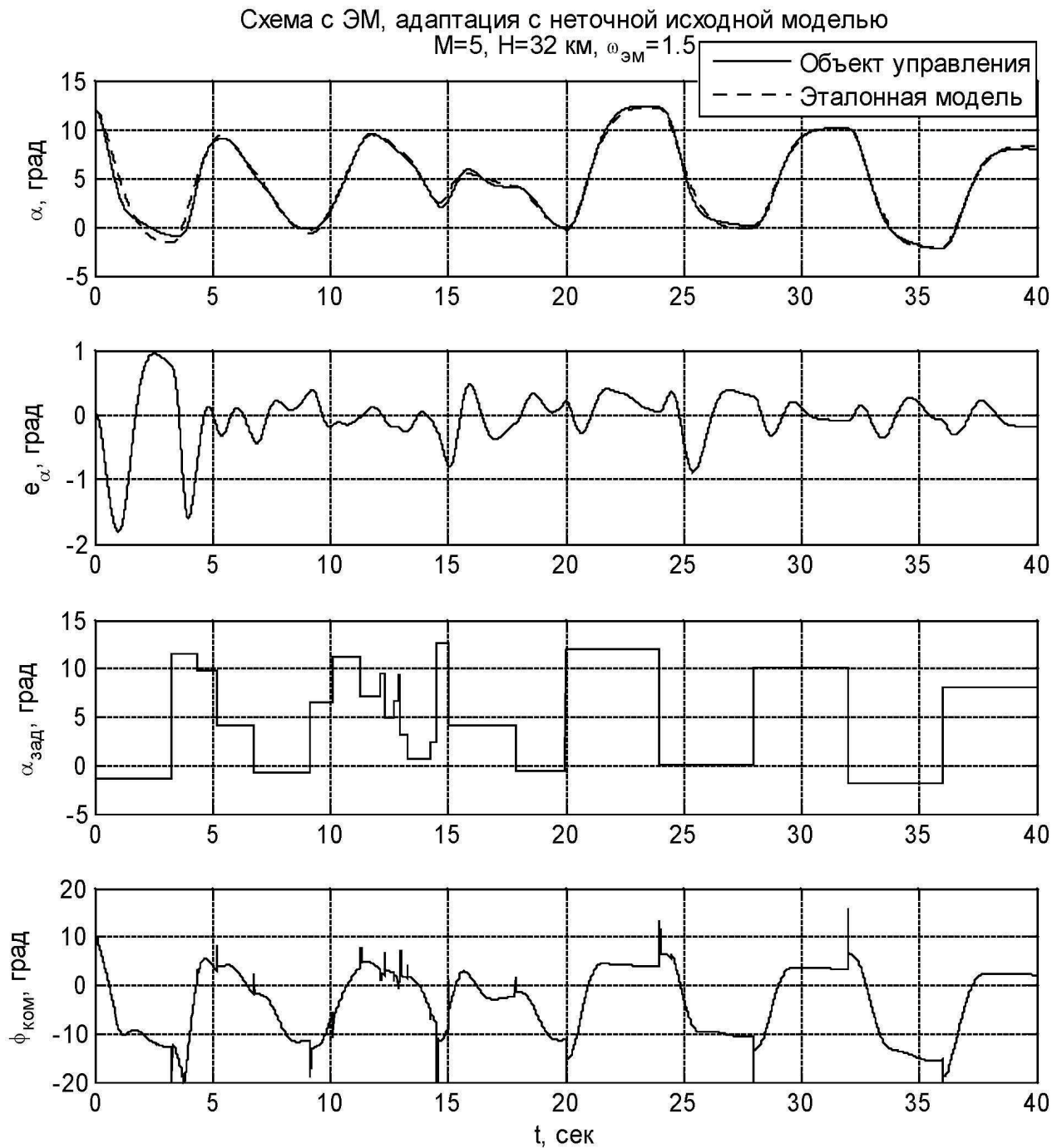


Рис. А.113. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 5, H = 32 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{\text{зад}}$ — задающий сигнал по углу атаки, град.; $\phi_{\text{ком}}$ — командный сигнал для привода элевонов, град; t — время, с

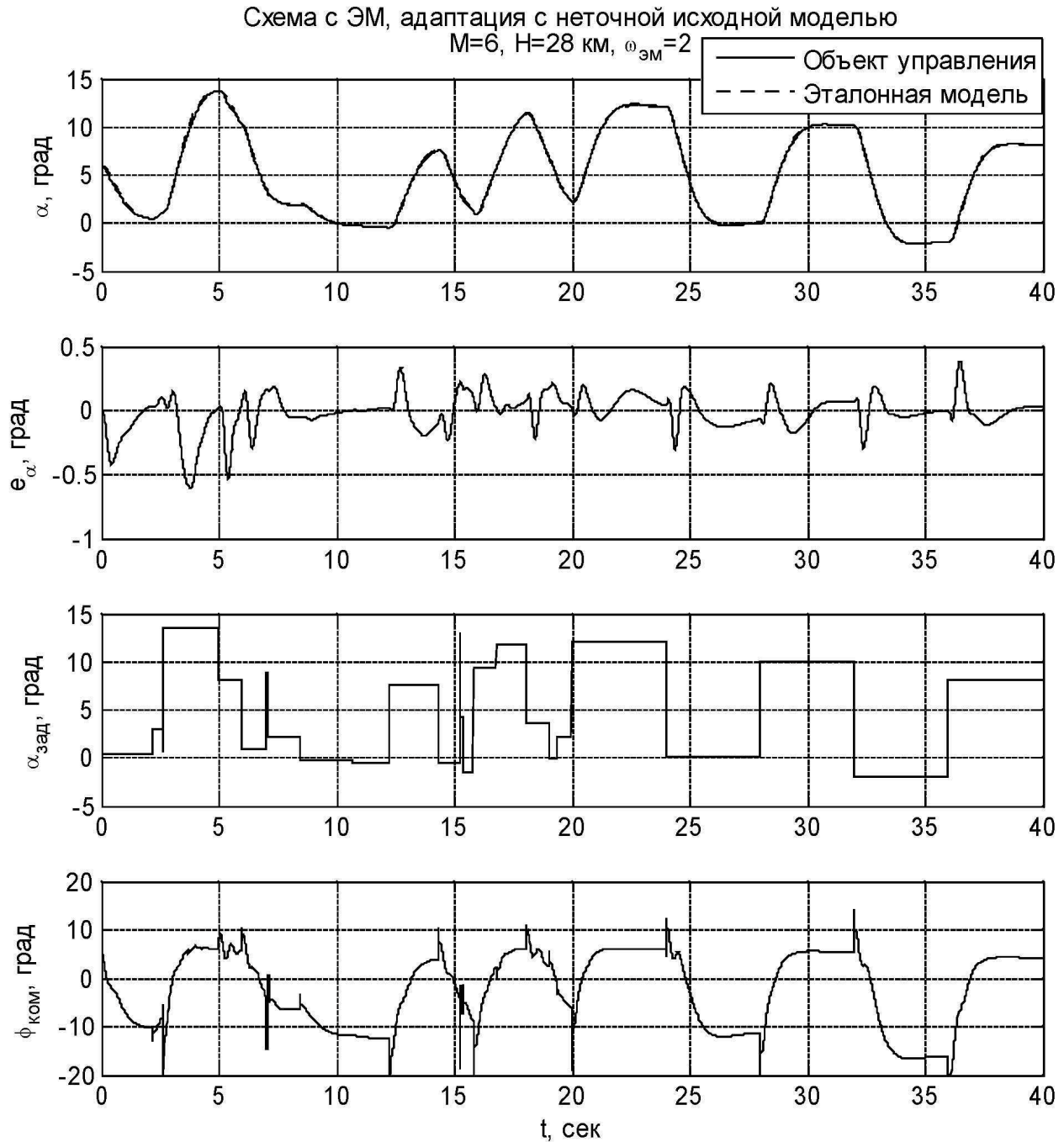


Рис. А.114. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 6, H = 28 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{\text{зад}}$ — задающий сигнал по углу атаки, град.; $\phi_{\text{ком}}$ — командный сигнал для привода элевонов, град; t — время, с

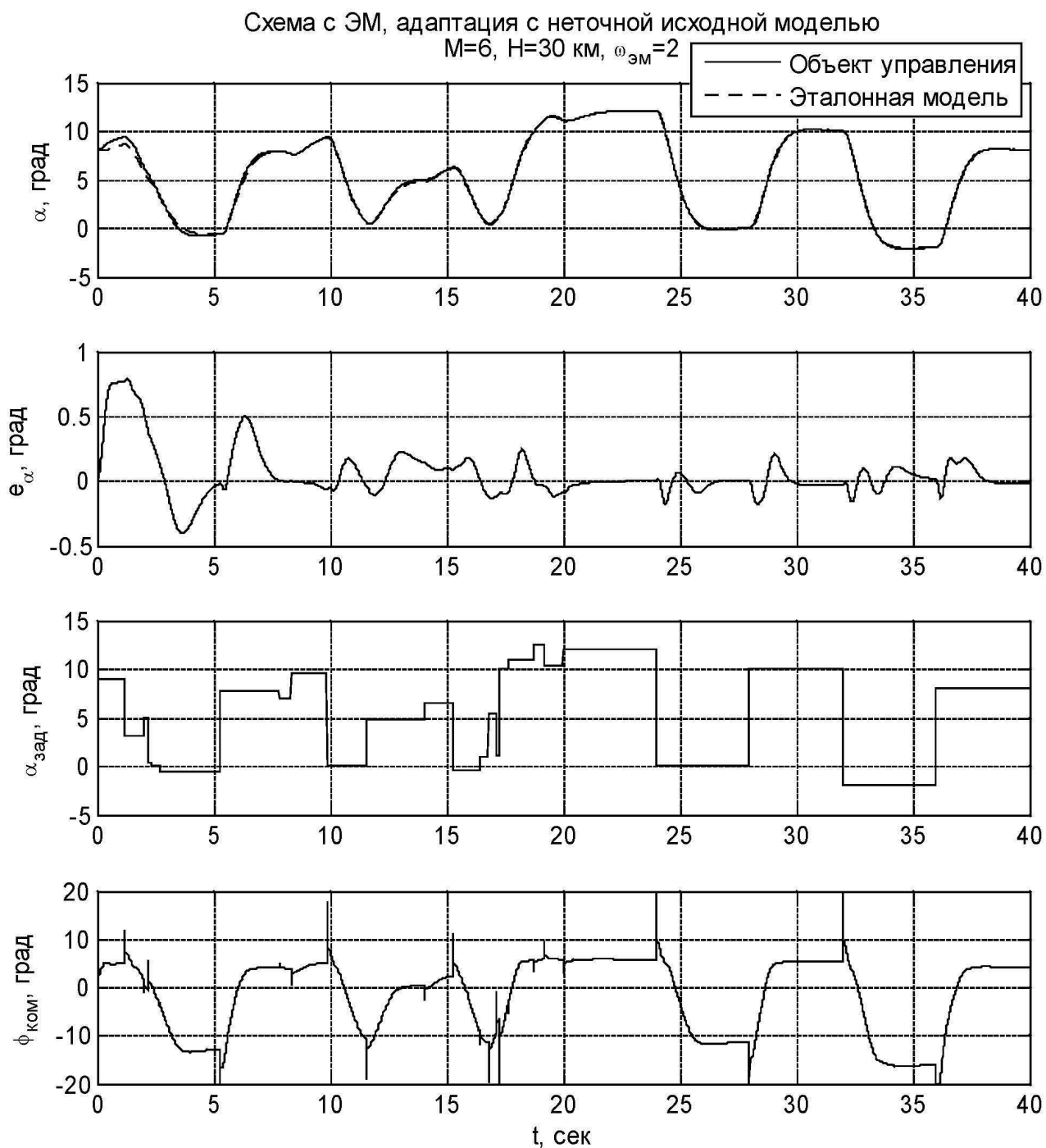


Рис. А.115. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 6, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

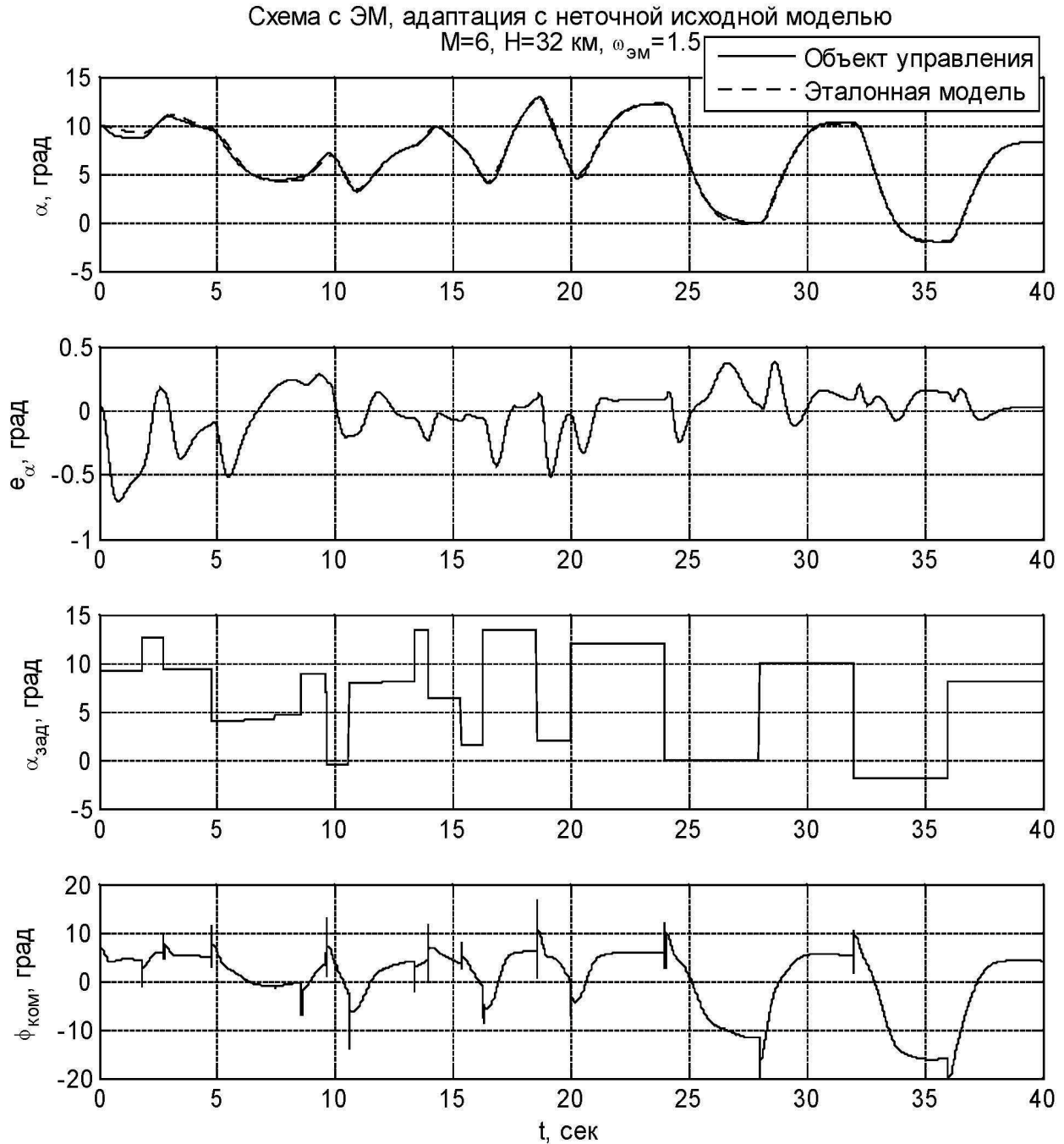


Рис. А.116. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 6, H = 32 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{\text{зад}}$ — задающий сигнал по углу атаки, град.; $\phi_{\text{ком}}$ — командный сигнал для привода элевонов, град; t — время, с

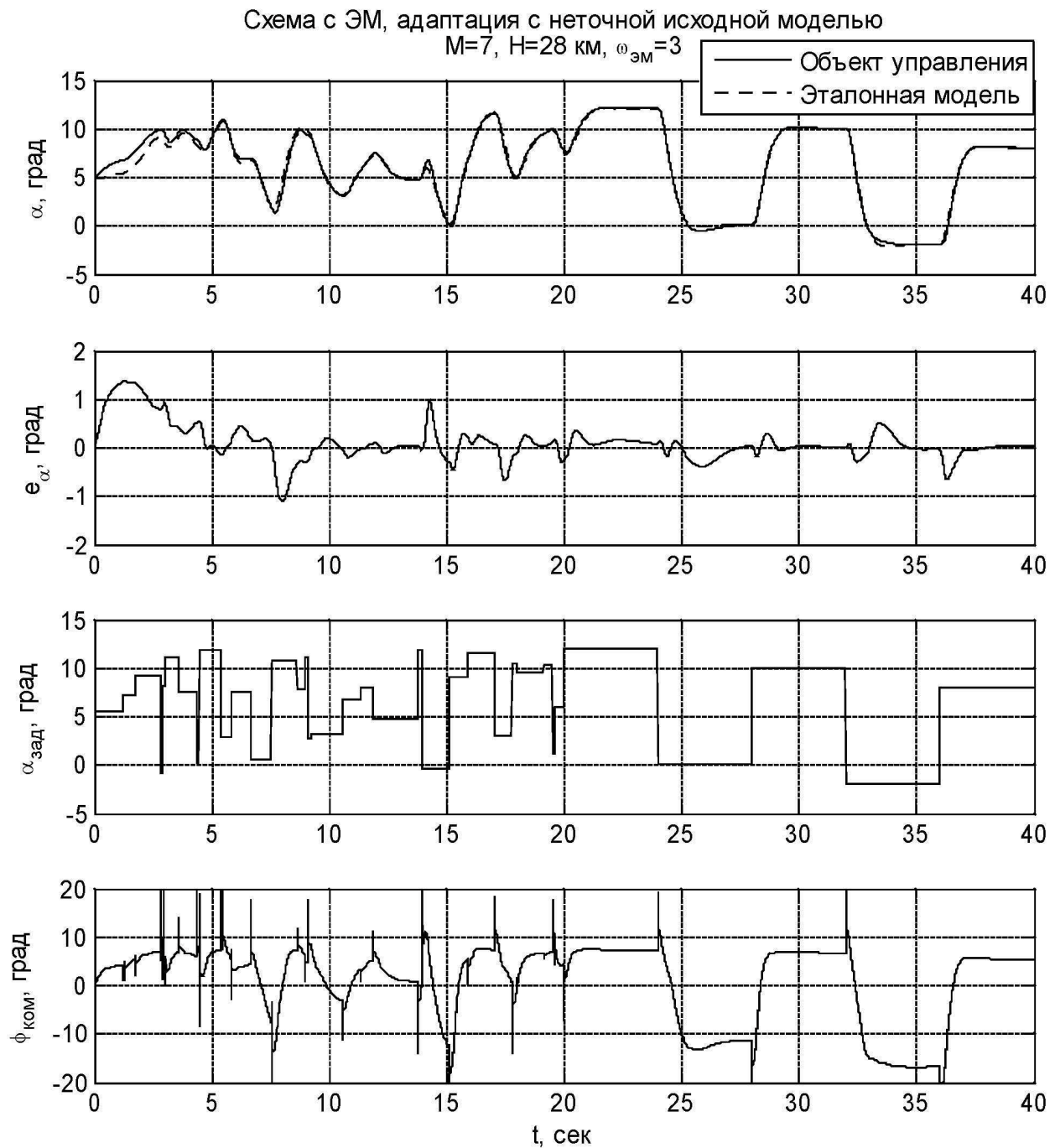


Рис. А.117. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 7, H = 28 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{\text{зад}}$ — задающий сигнал по углу атаки, град.; $\phi_{\text{ком}}$ — командный сигнал для привода элевонов, град; t — время, с

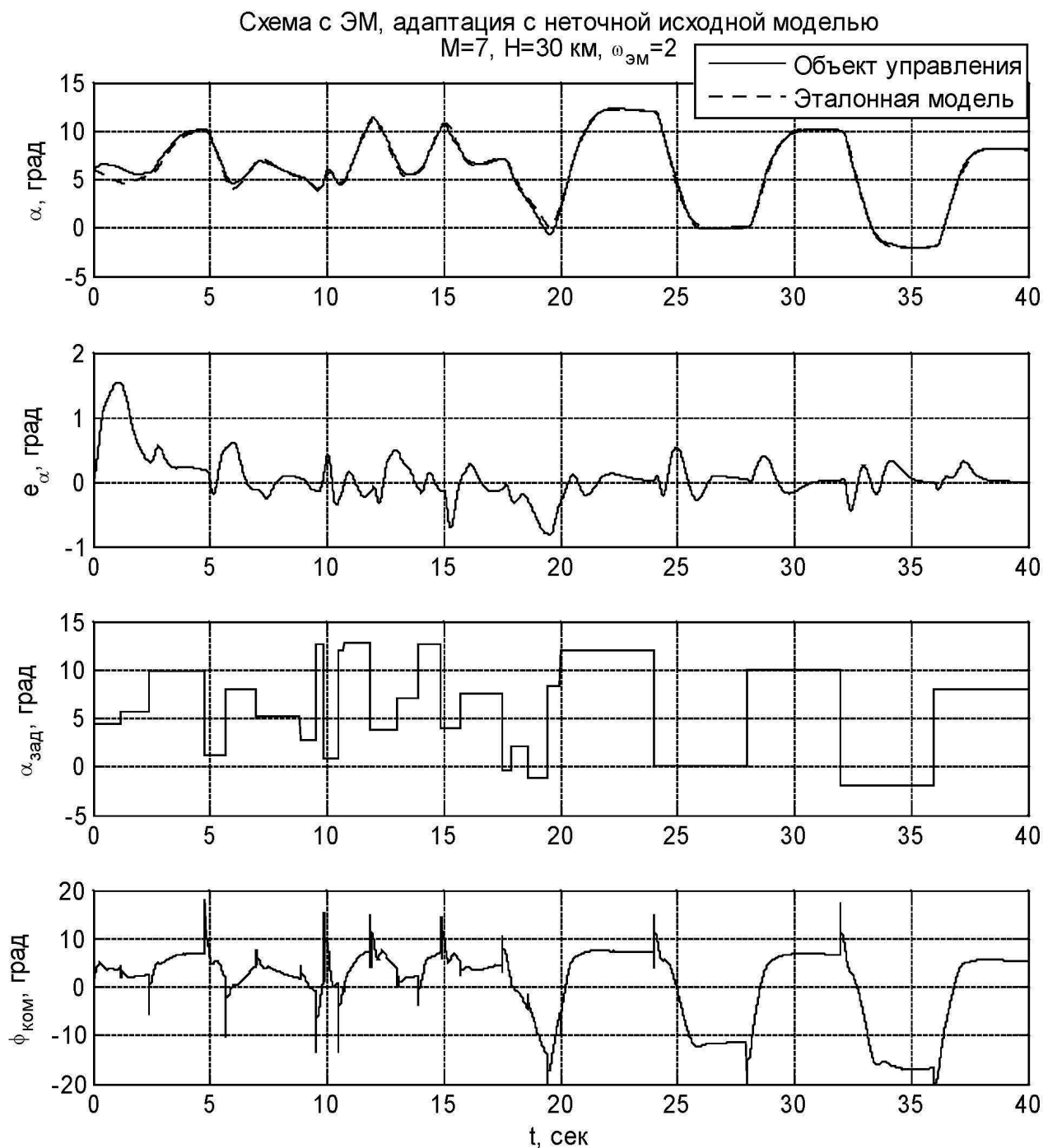


Рис. А.118. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 7, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{\text{зад}}$ — задающий сигнал по углу атаки, град.; $\phi_{\text{ком}}$ — командный сигнал для привода элевонов, град; t — время, с

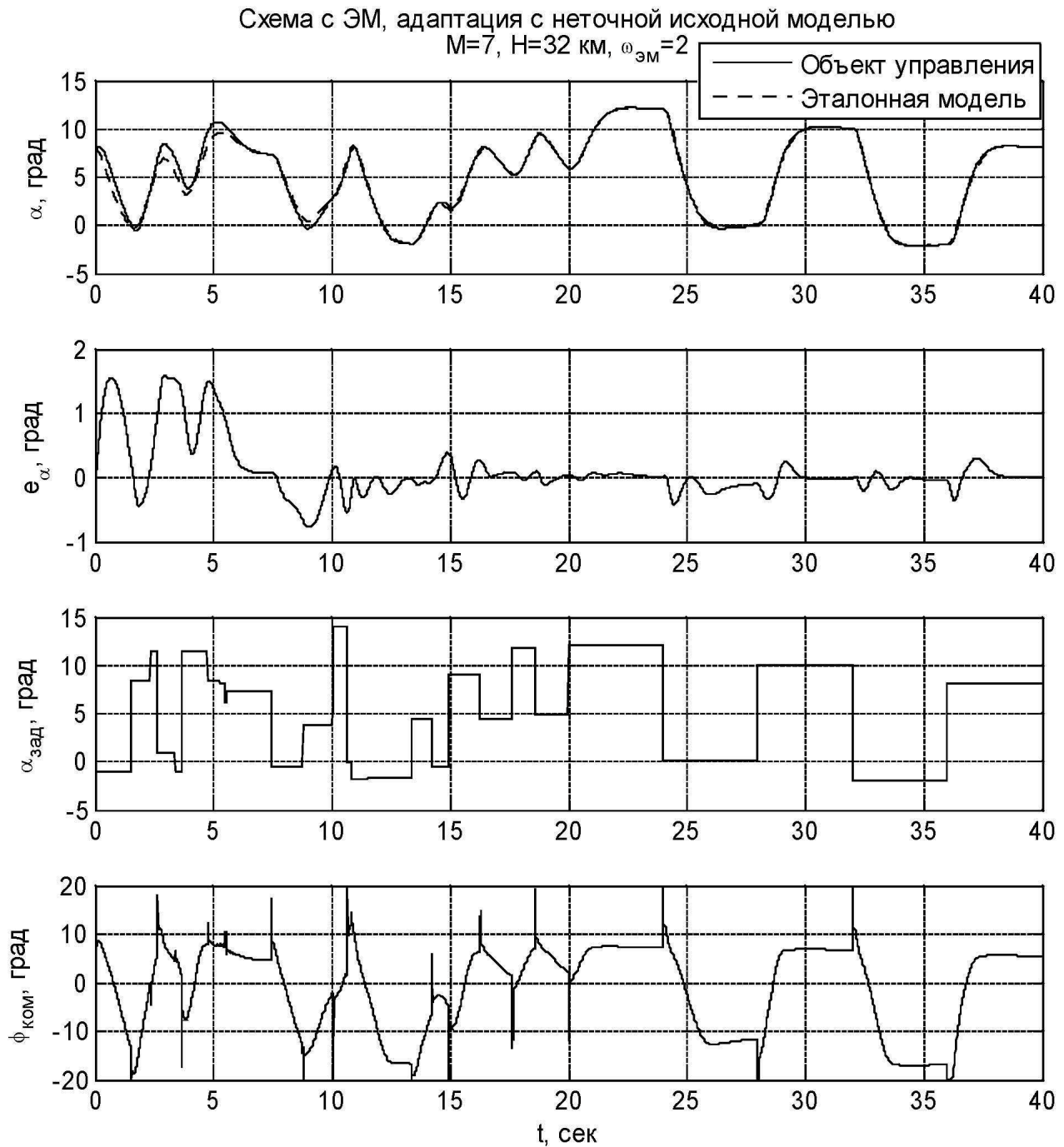


Рис. А.119. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 7, H = 32 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

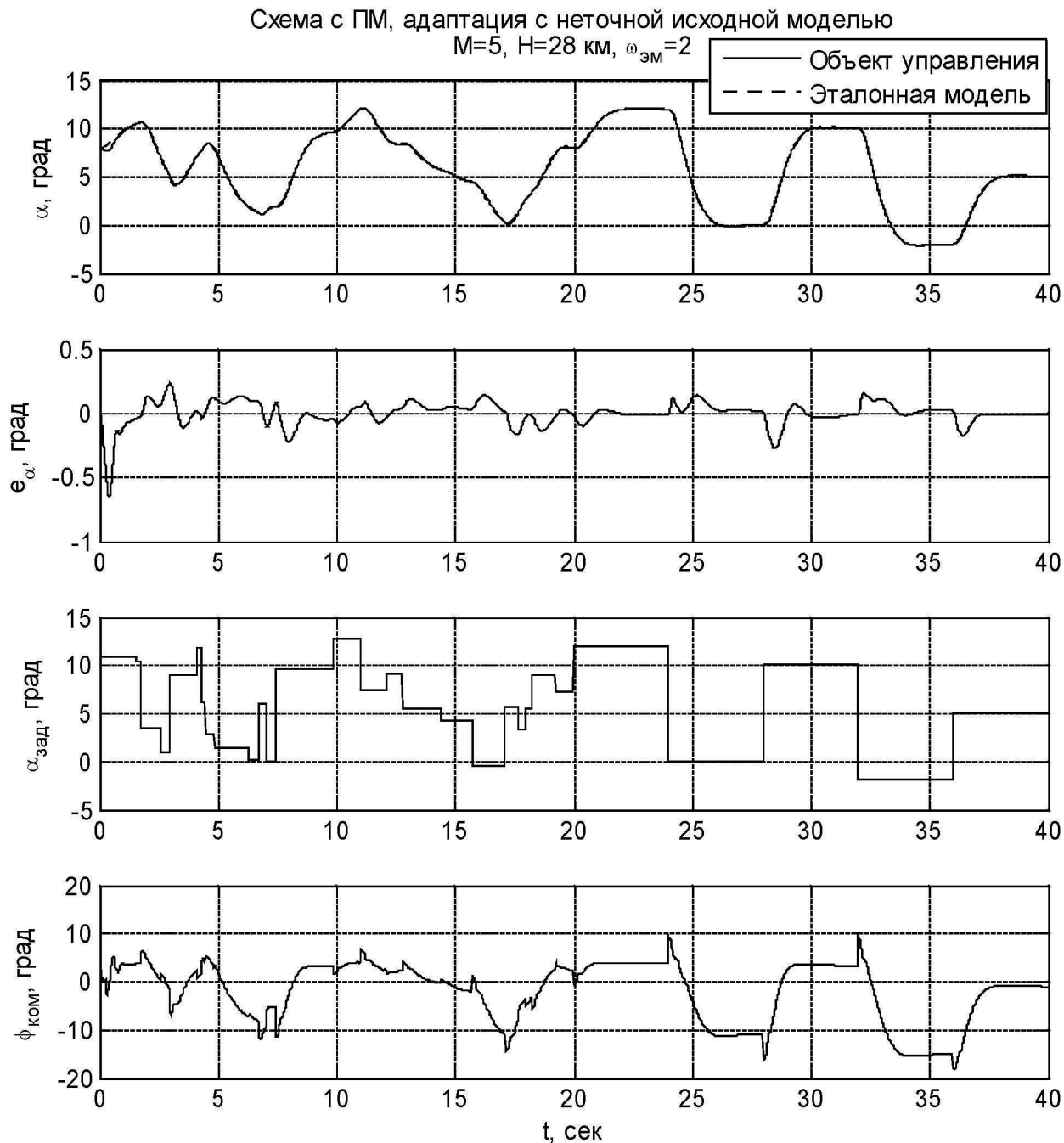


Рис. А.120. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 5, H = 28 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

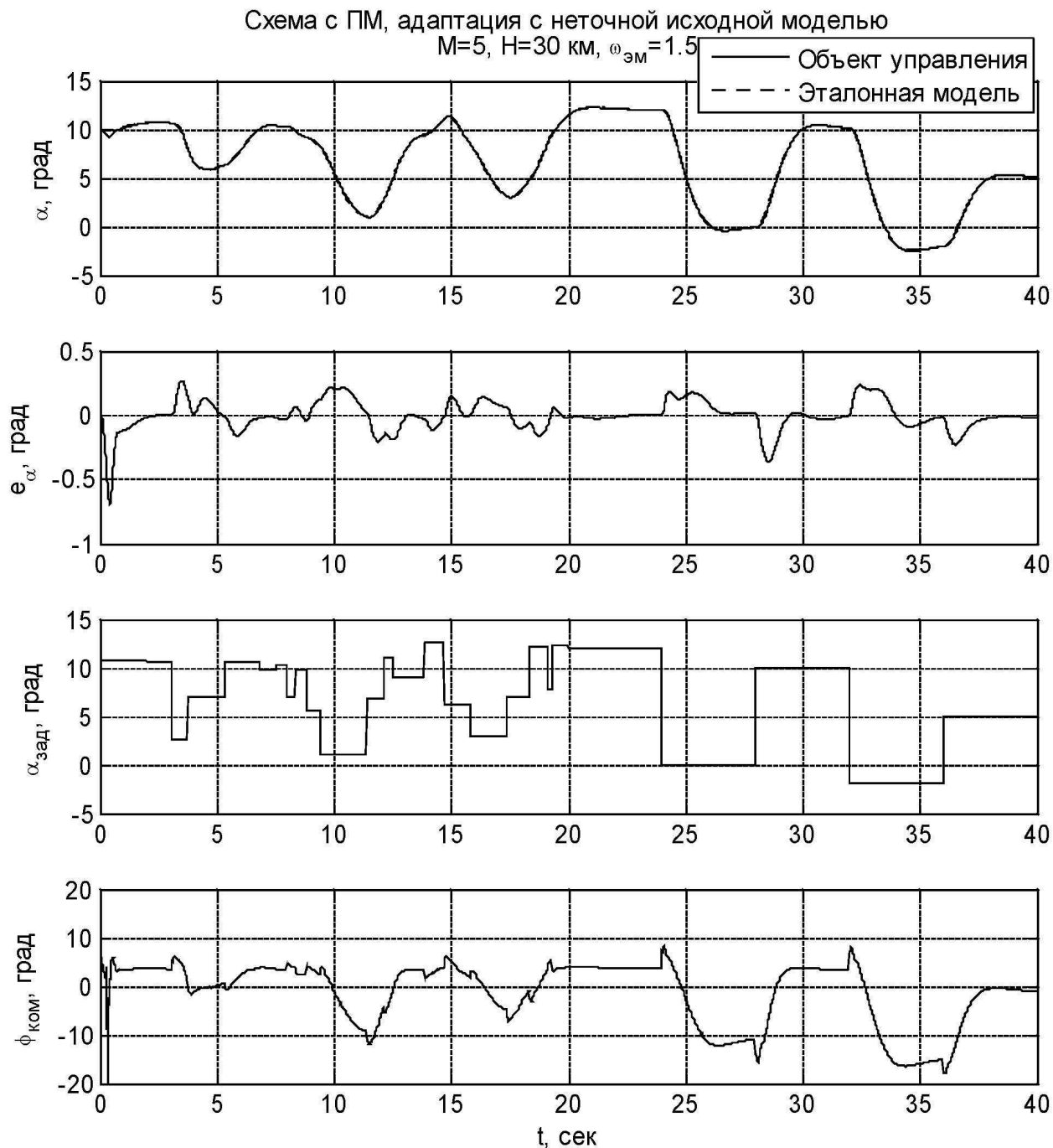


Рис. А.121. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 5, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

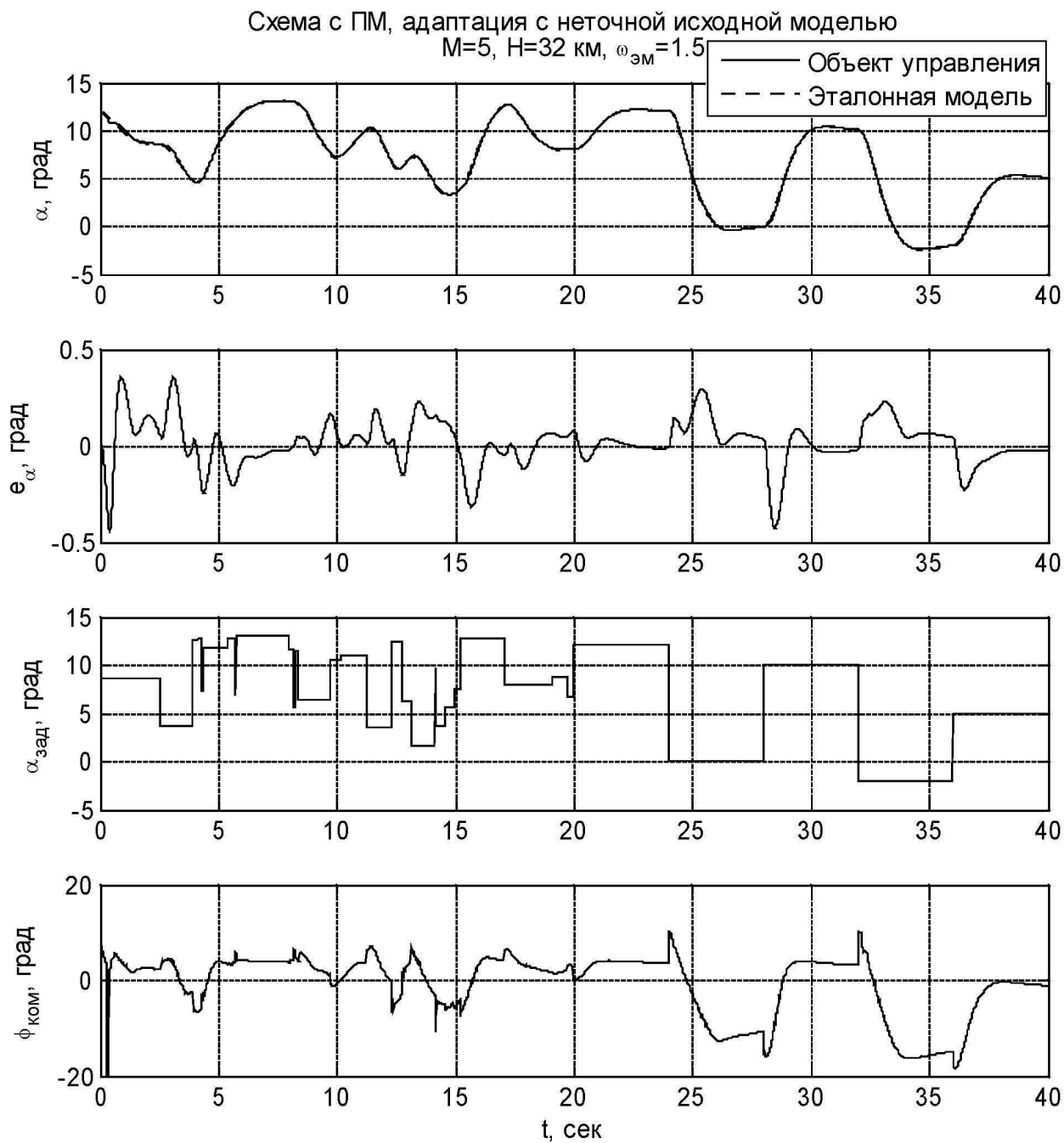


Рис. А.122. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 5, H = 32 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

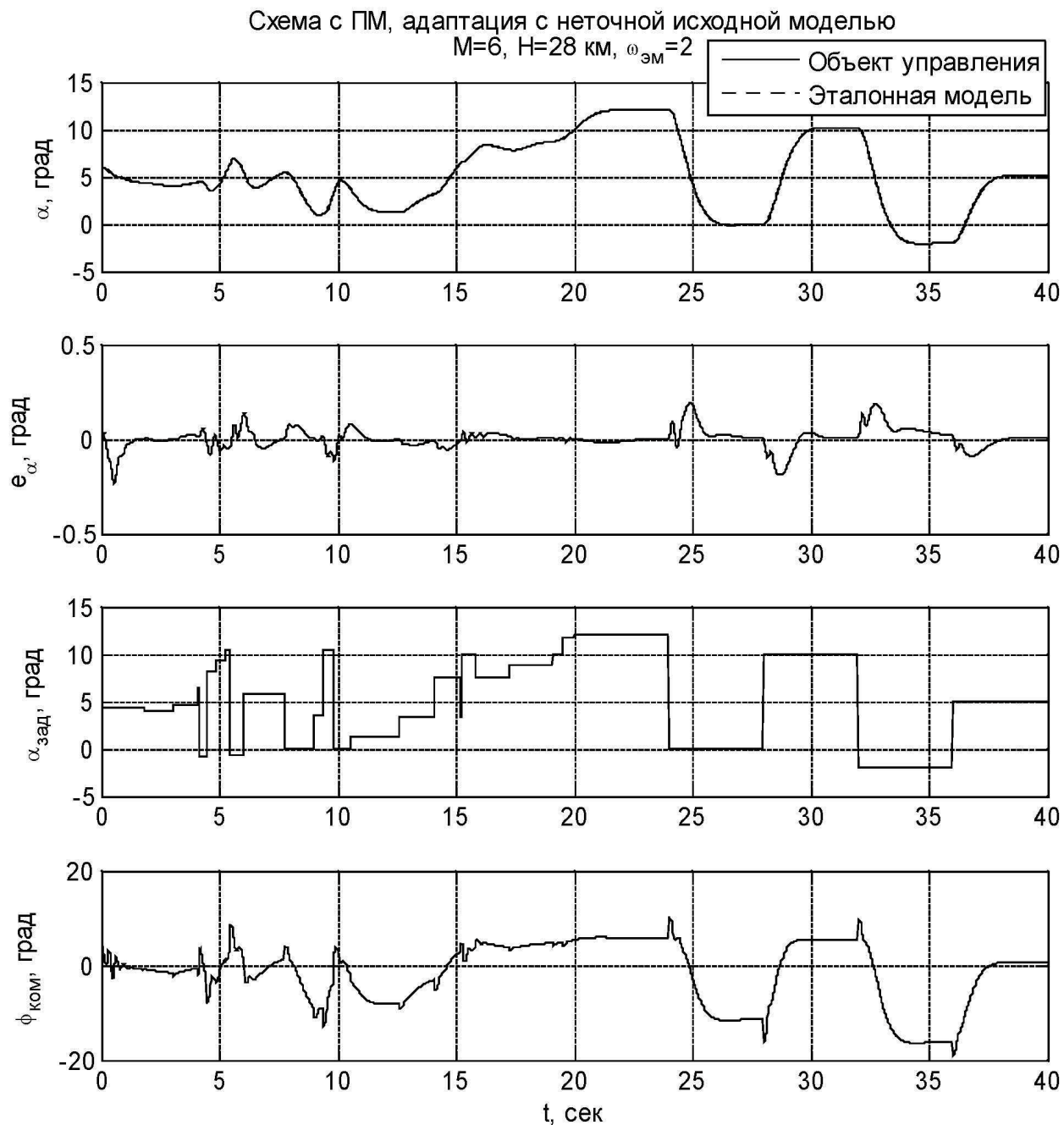


Рис. А.123. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 6, H = 28 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{\text{зад}}$ — задающий сигнал по углу атаки, град.; $\phi_{\text{ком}}$ — командный сигнал для привода элевонов, град; t — время, с

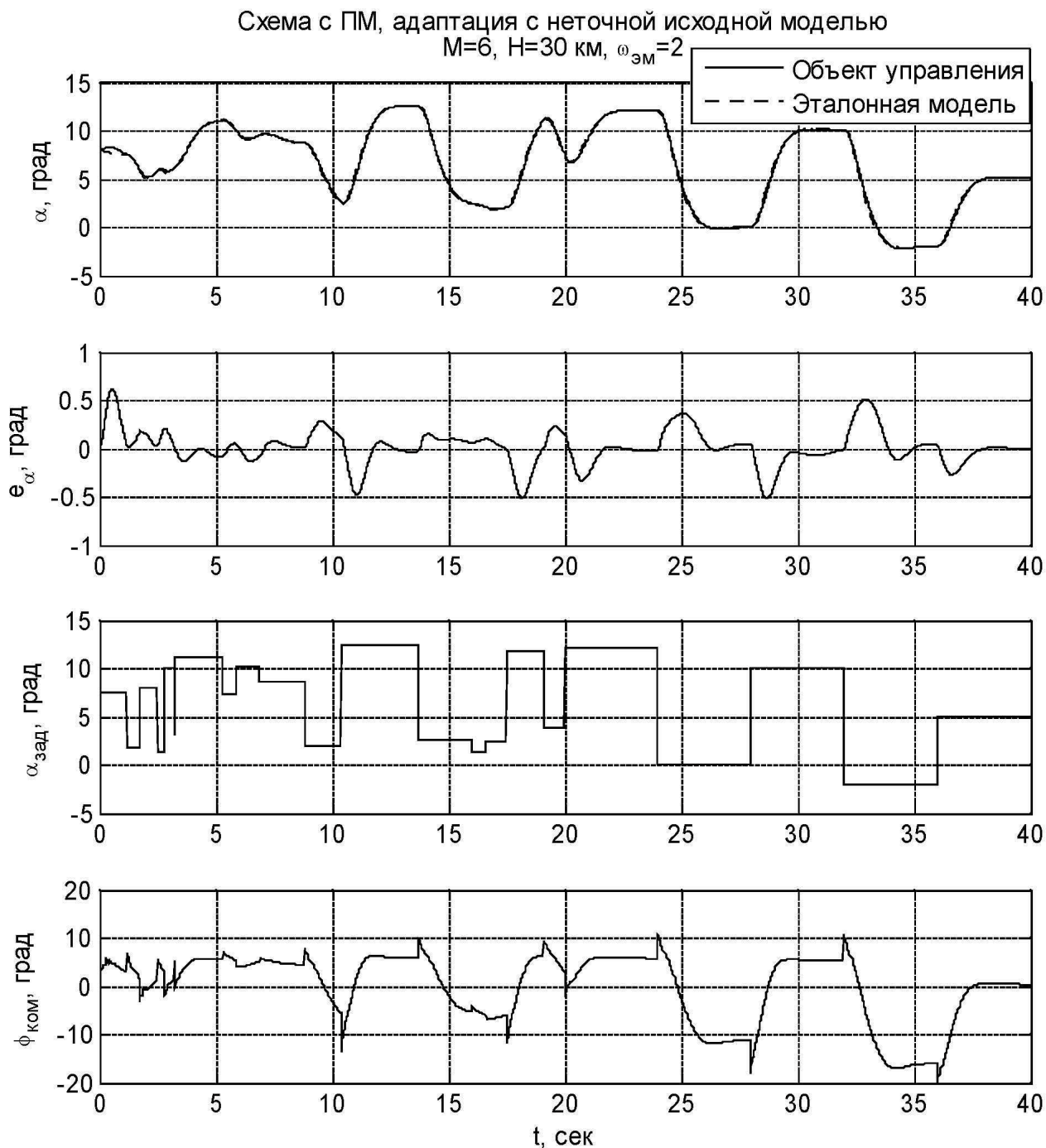


Рис. А.124. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 6, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

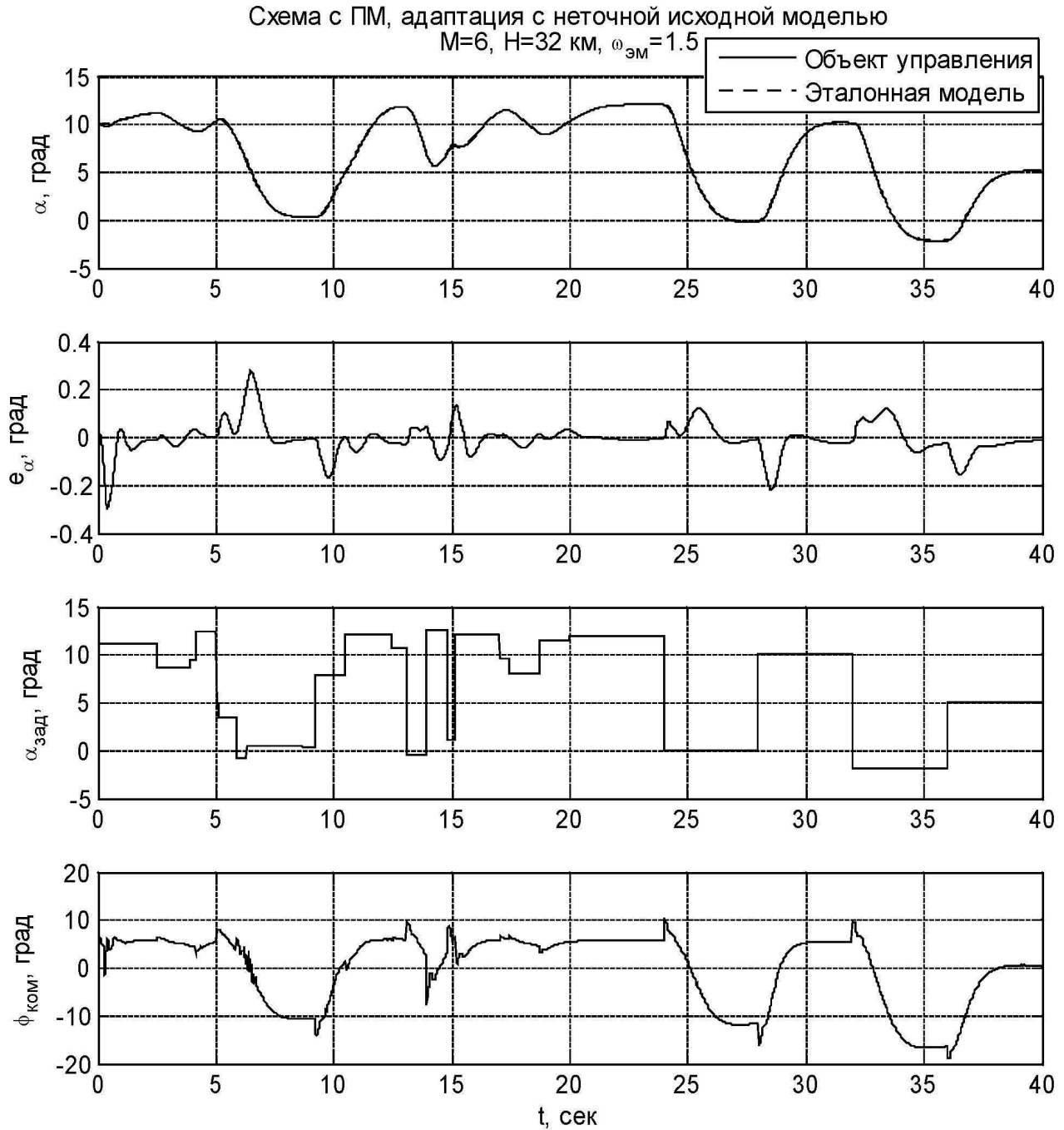


Рис. А.125. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 6, H = 32 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{\text{зад}}$ — задающий сигнал по углу атаки, град.; $\phi_{\text{ком}}$ — командный сигнал для привода элевонов, град; t — время, с

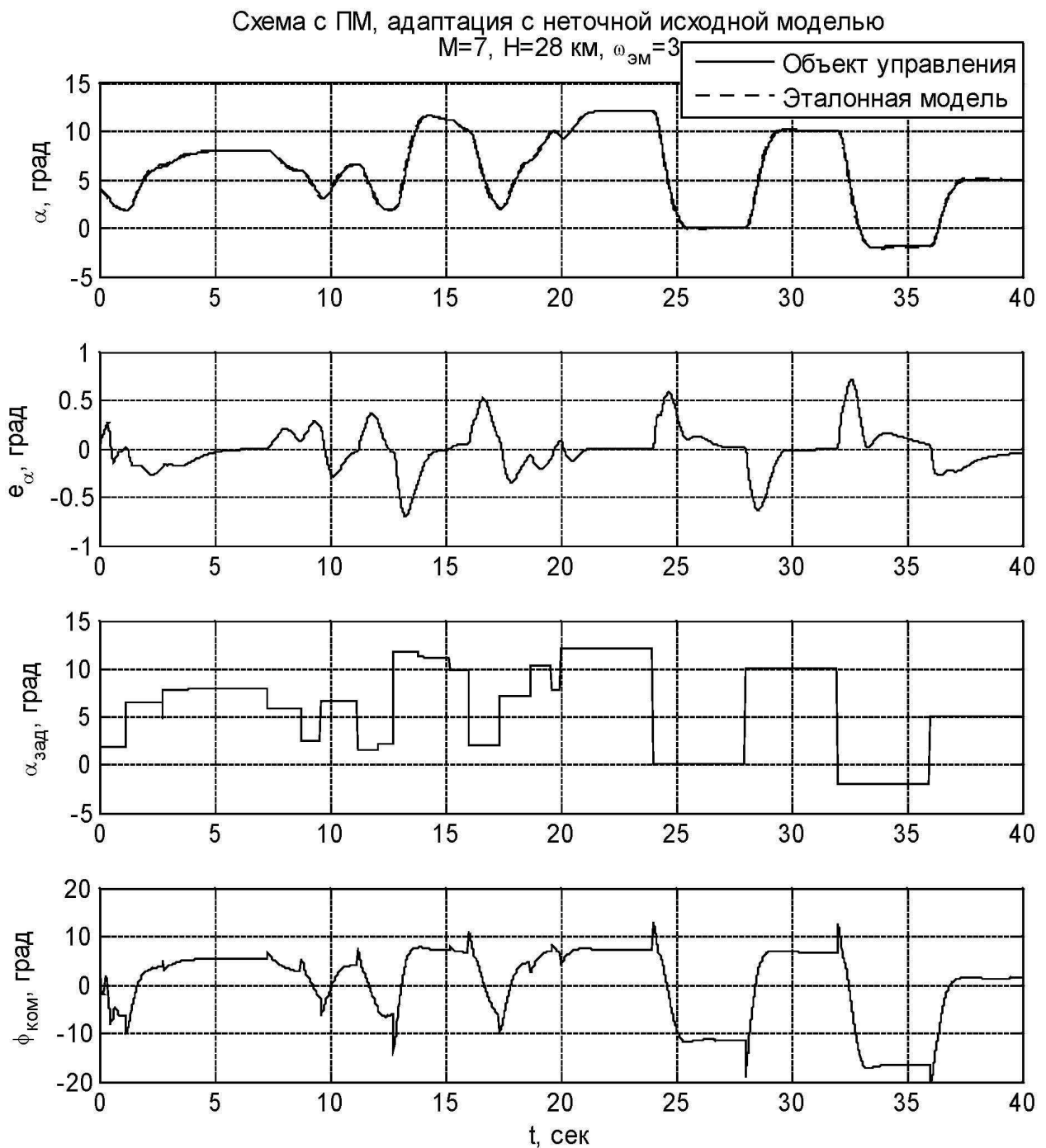


Рис. А.126. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 7, H = 28 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

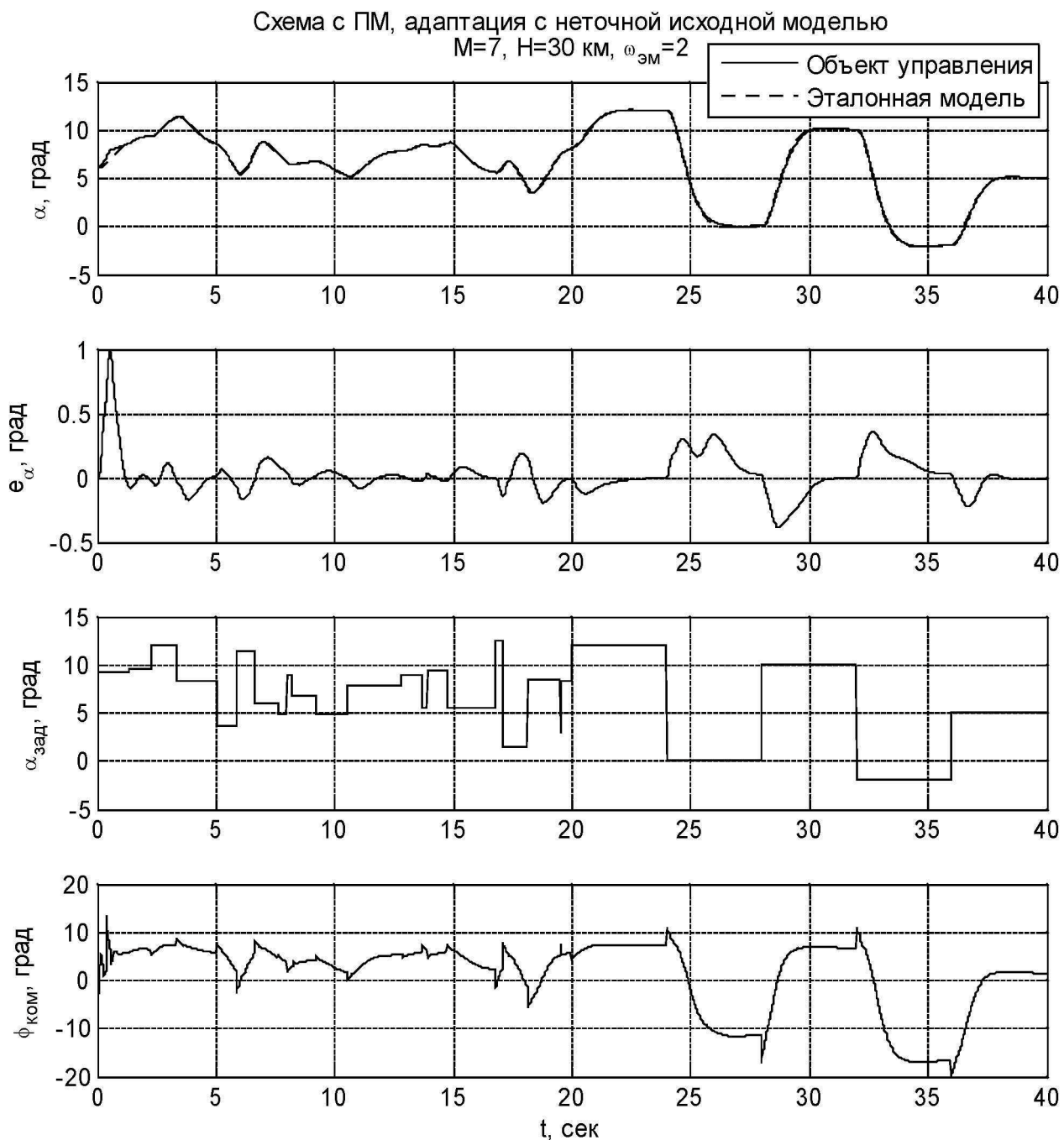


Рис. А.127. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 7, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{\text{зад}}$ — задающий сигнал по углу атаки, град.; $\phi_{\text{ком}}$ — командный сигнал для привода элевонов, град; t — время, с

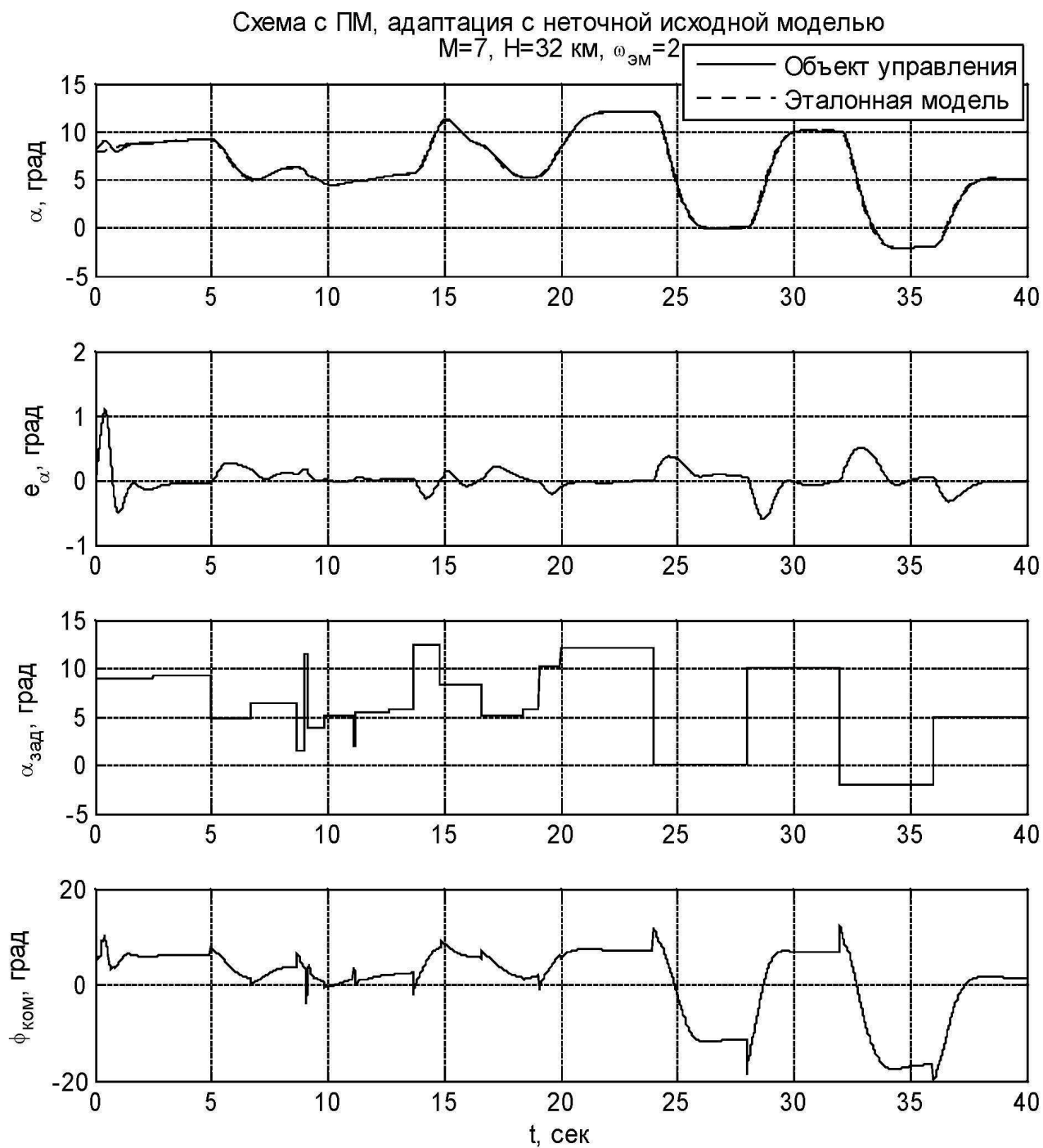


Рис. А.128. Результаты вычислительного эксперимента по адаптации к неопределенности в исходных данных для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 7, H = 32 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

А.8. Оценка значимости механизмов адаптации в задаче управления угловым движением ЛА

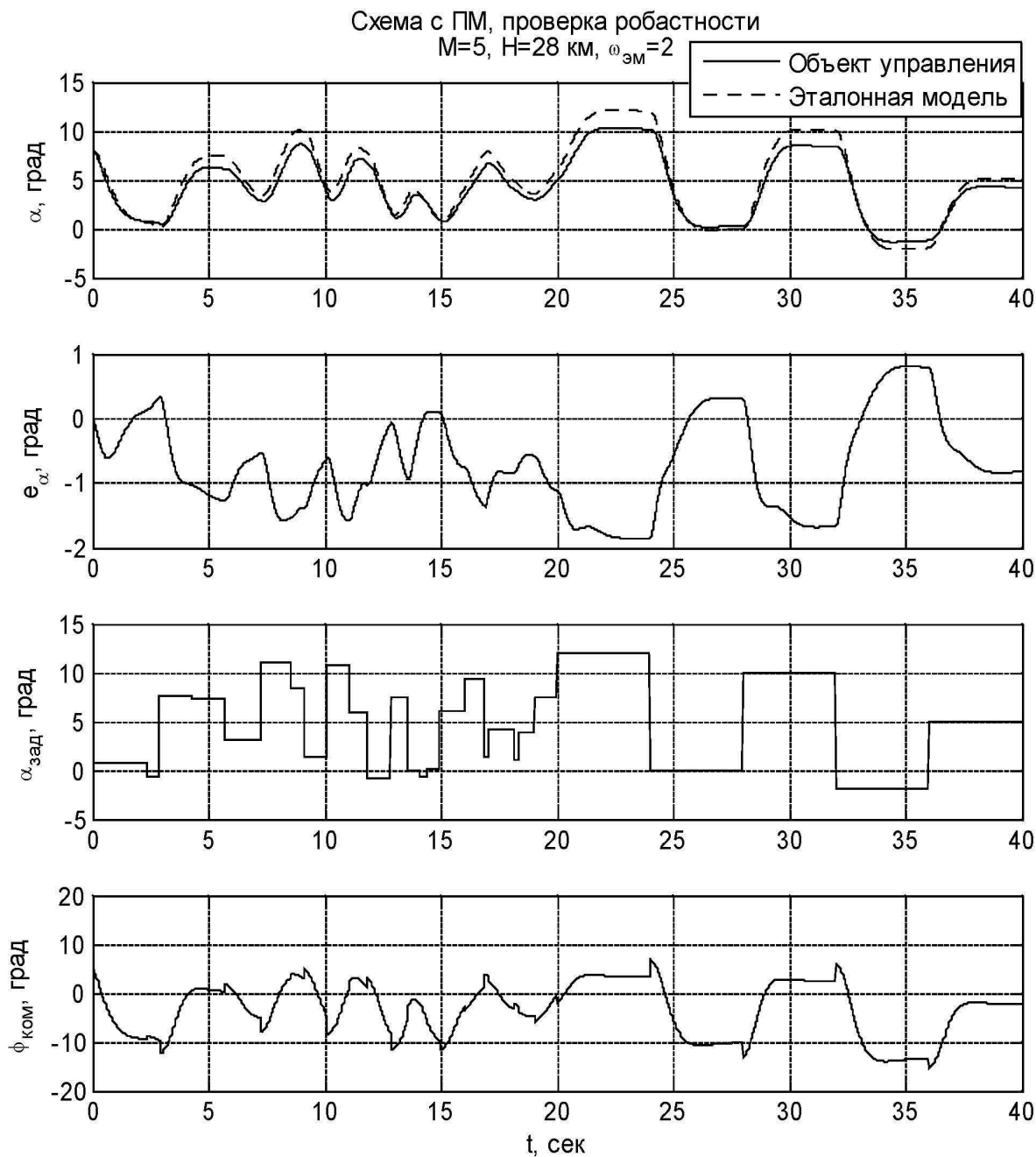


Рис. А.129. Результаты вычислительного эксперимента по оценке значимости механизмов адаптации для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 5$, $H = 28$ км). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

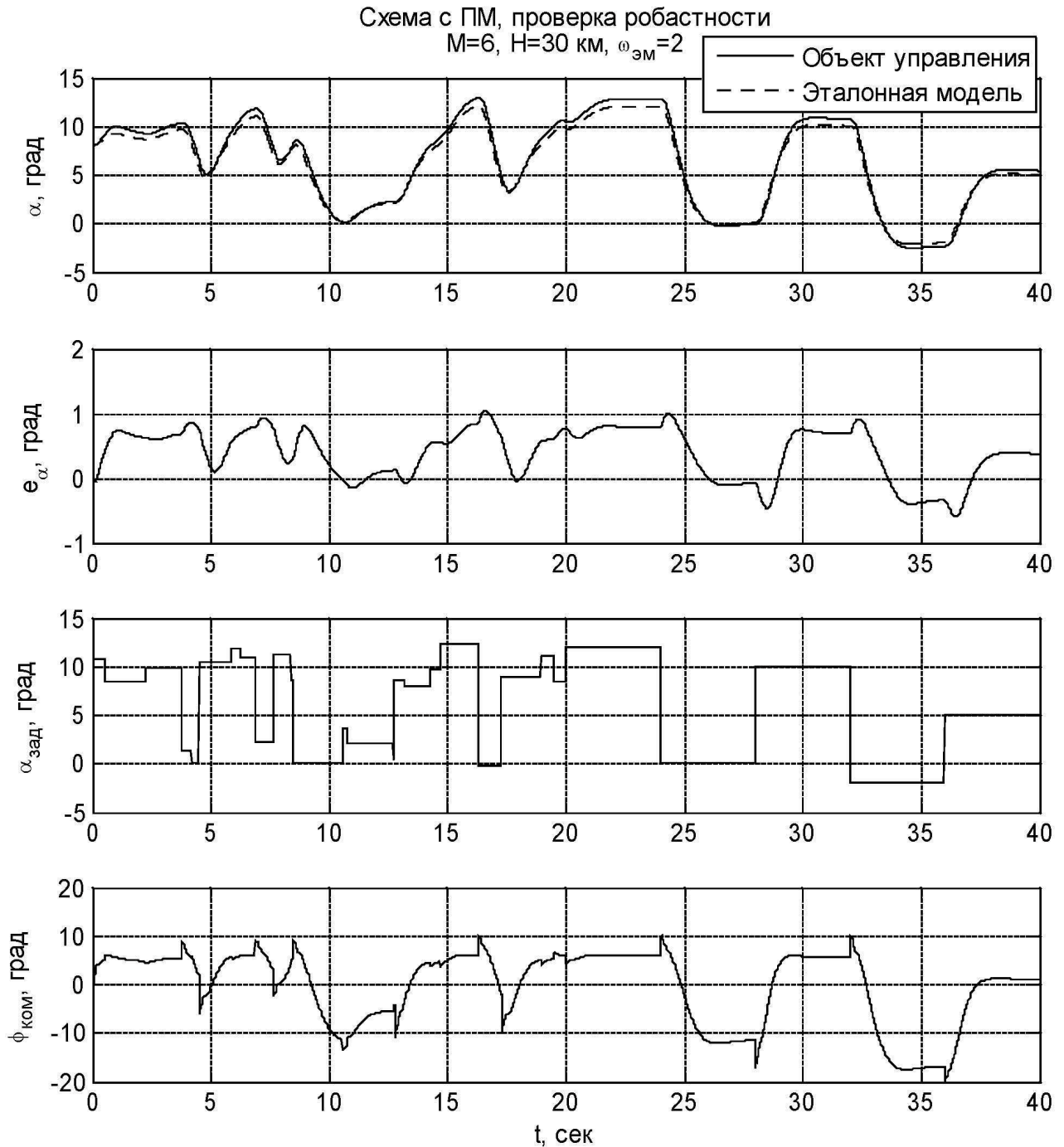


Рис. А.130. Результаты вычислительного эксперимента по оценке значимости механизмов адаптации для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 6, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

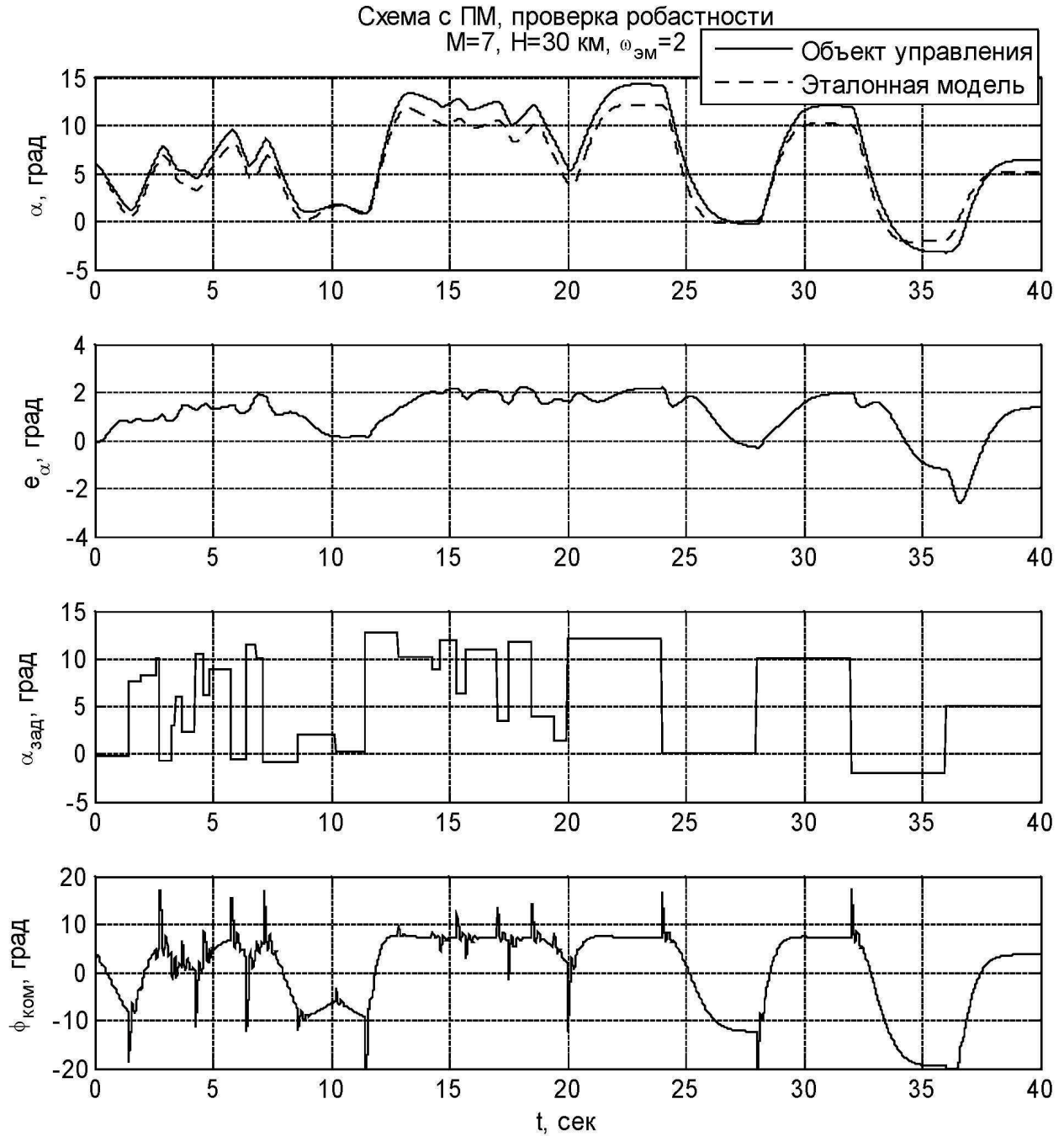


Рис. А.131. Результаты вычислительного эксперимента по оценке значимости механизмов адаптации для системы управления с прогнозирующей моделью применительно к перспективному ГЗЛА (режим полета $M = 7, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_{α} — ошибка отслеживания заданного угла атаки, град; $\alpha_{\text{зад}}$ — задающий сигнал по углу атаки, град.; $\phi_{\text{ком}}$ — командный сигнал для привода элевонов, град; t — время, с

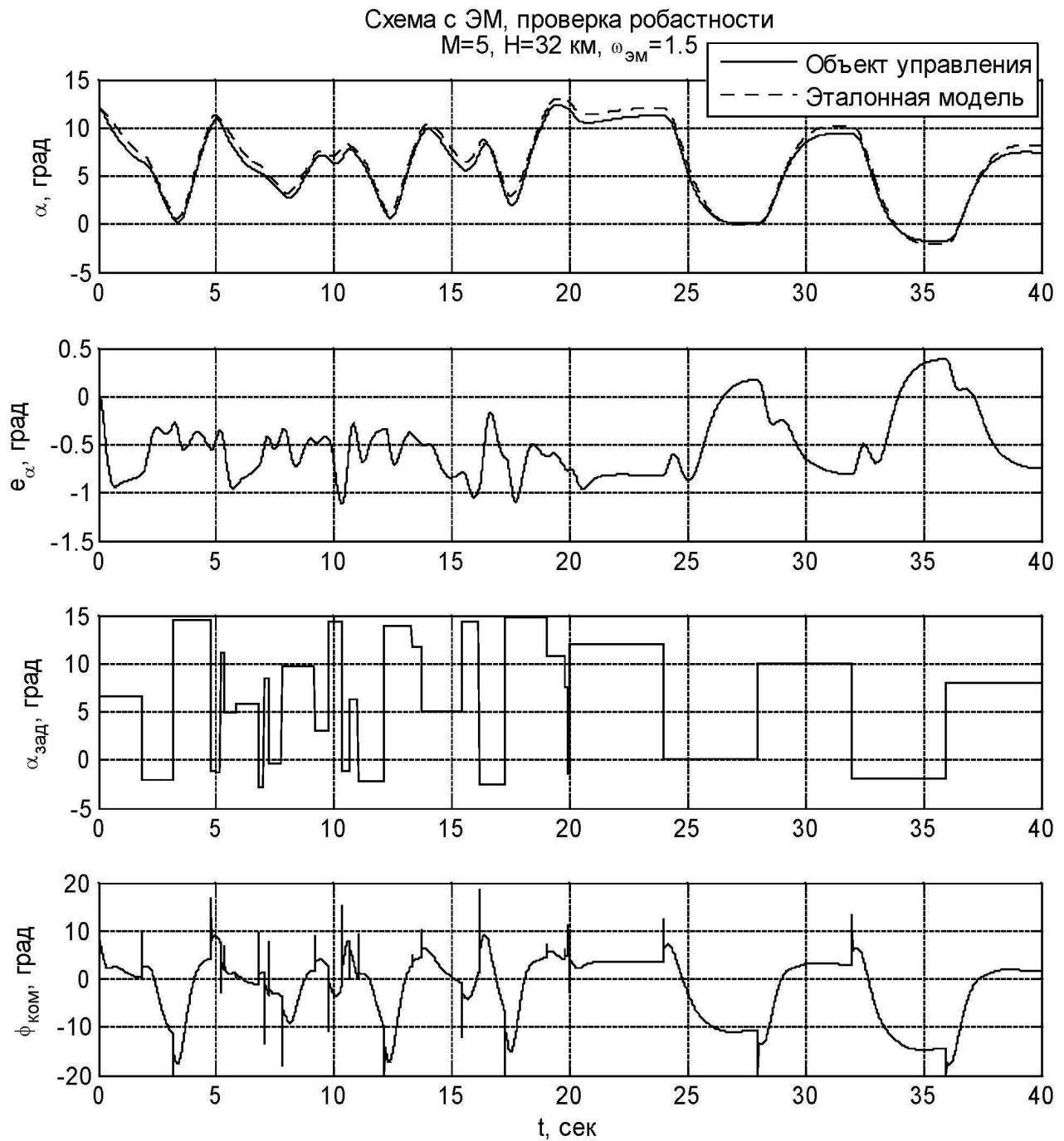


Рис. А.132. Результаты вычислительного эксперимента по оценке значимости механизмов адаптации для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 5, H = 32 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

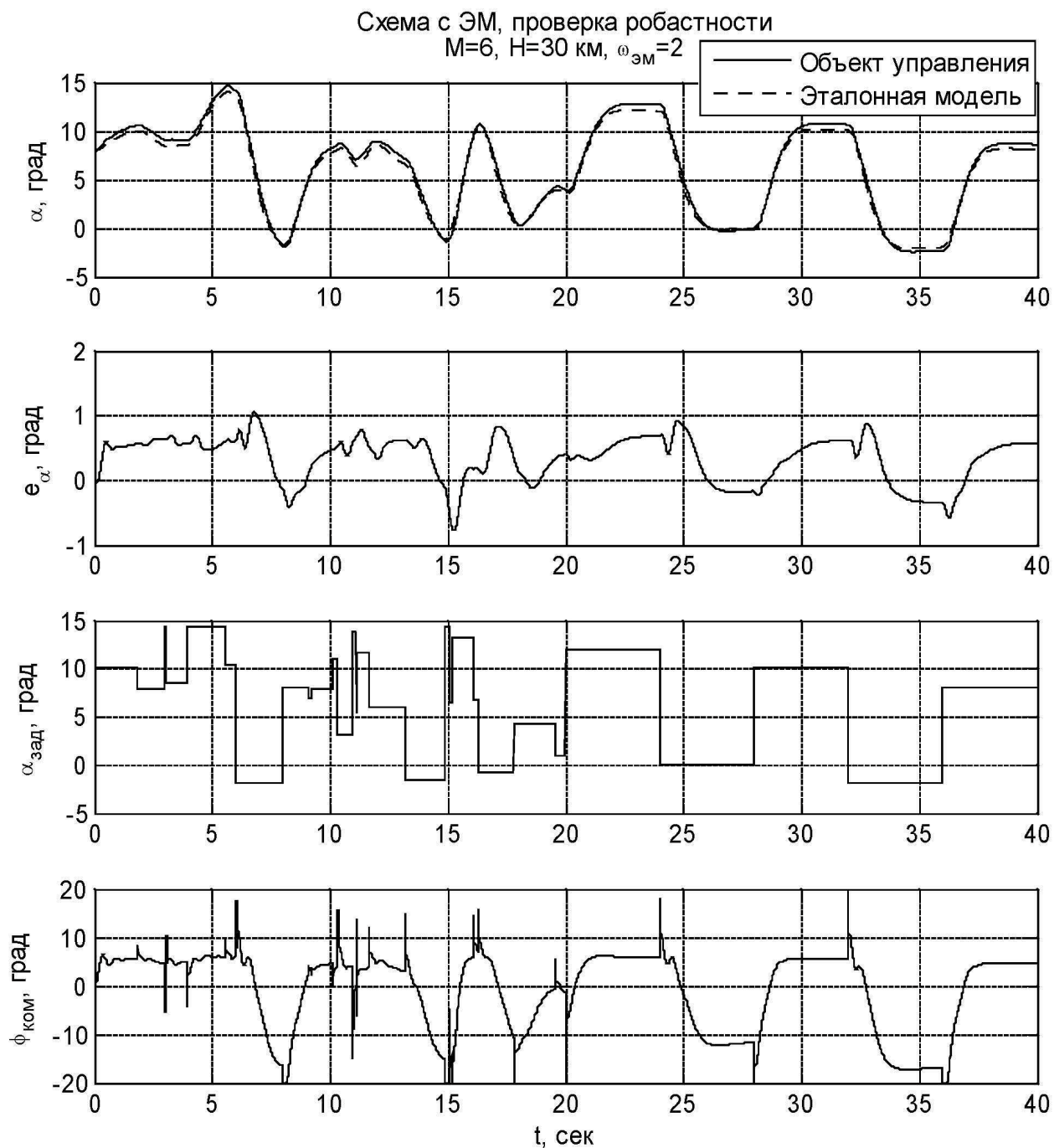


Рис. А.133. Результаты вычислительного эксперимента по оценке значимости механизмов адаптации для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 6, H = 30 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{зад}$ — задающий сигнал по углу атаки, град.; $\phi_{ком}$ — командный сигнал для привода элевонов, град; t — время, с

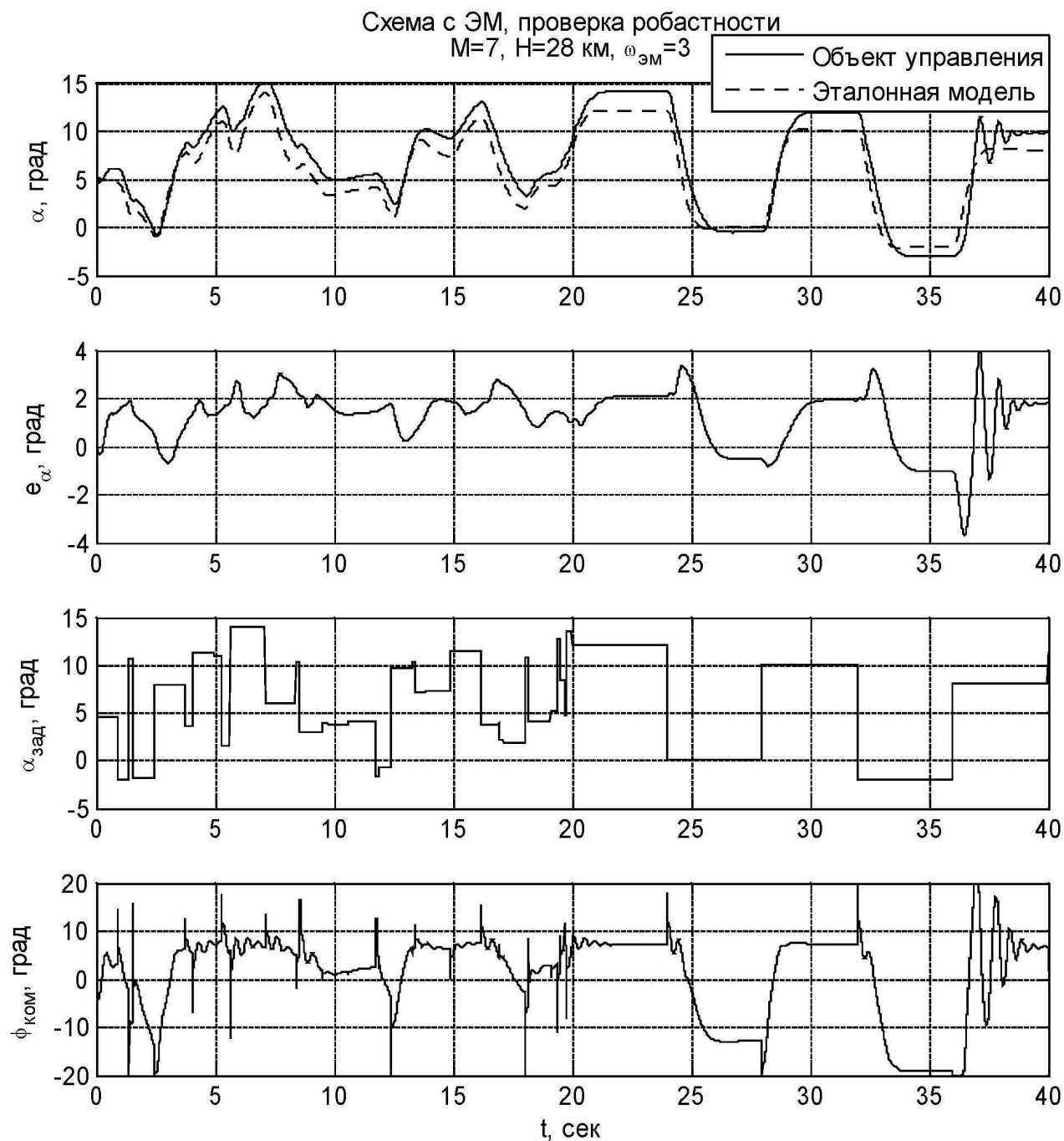


Рис. А.134. Результаты вычислительного эксперимента по оценке значимости механизмов адаптации для системы управления с эталонной моделью применительно к перспективному ГЗЛА (режим полета $M = 7, H = 28 \text{ км}$). Обозначения: α — угол атаки, град; e_α — ошибка отслеживания заданного угла атаки, град; $\alpha_{\text{зад}}$ — задающий сигнал по углу атаки, град.; $\phi_{\text{ком}}$ — командный сигнал для привода элевонов, град; t — время, с

Приложения

Б Диаграммы информативности обучающих наборов для задачи моделирования продольного углового движения маневренного самолета

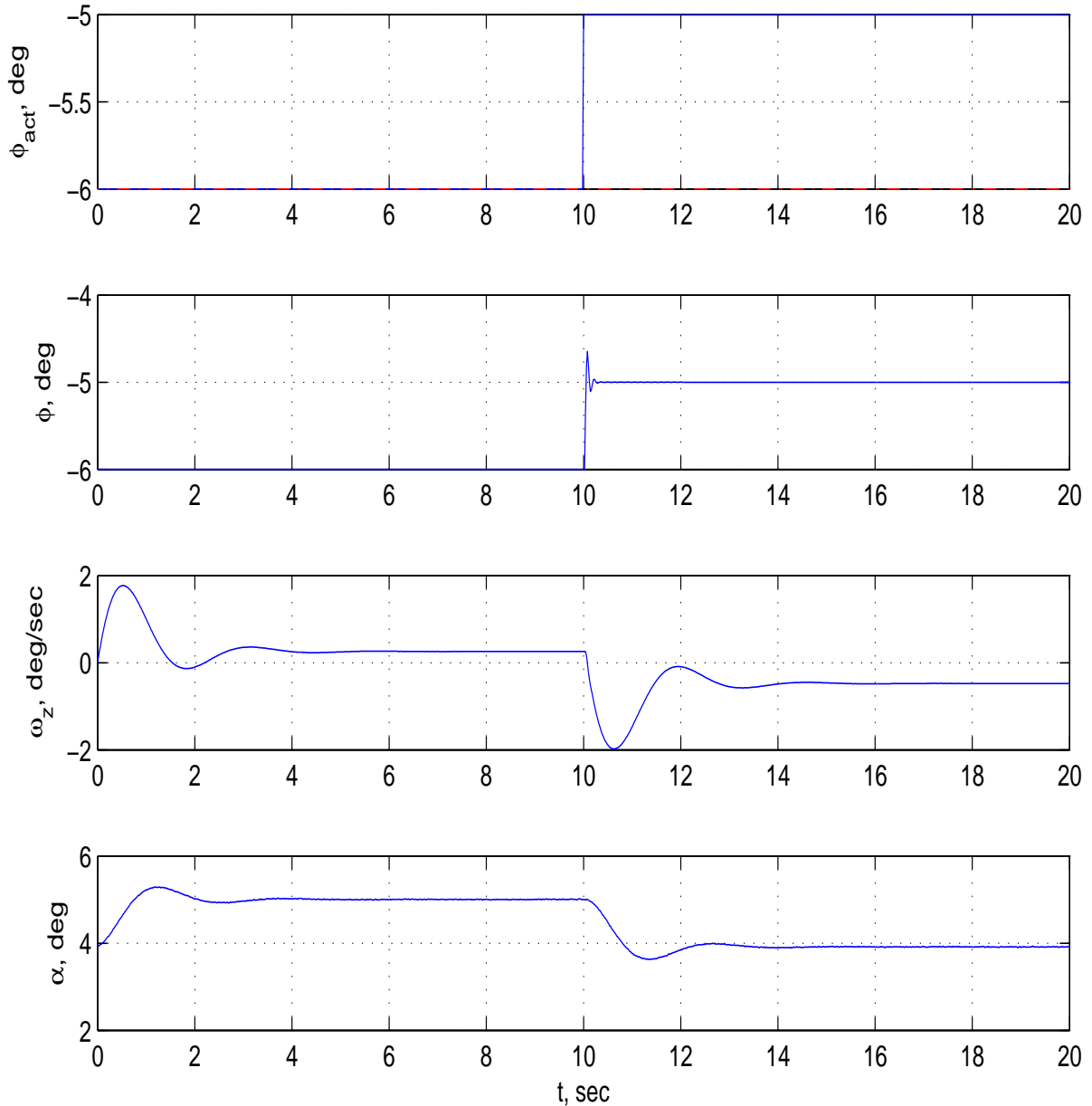
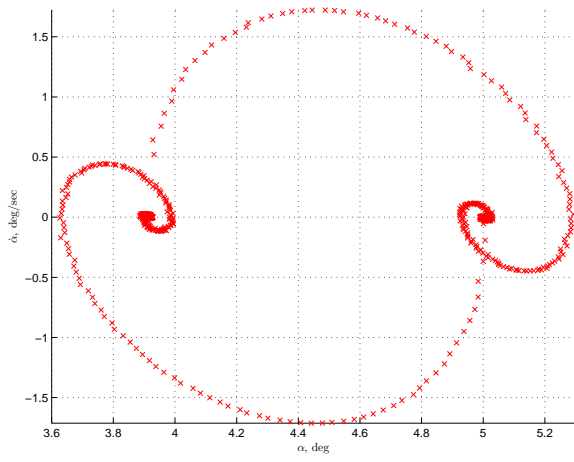
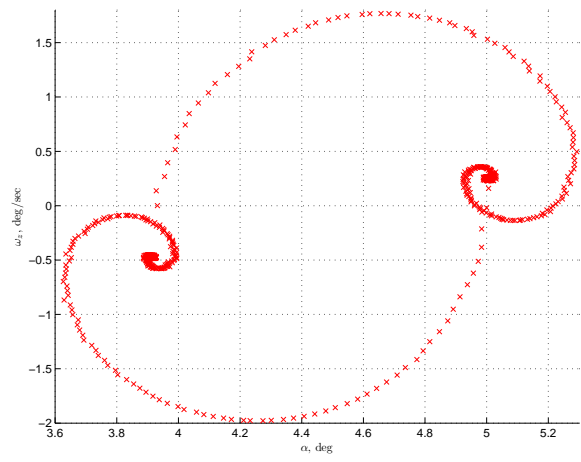


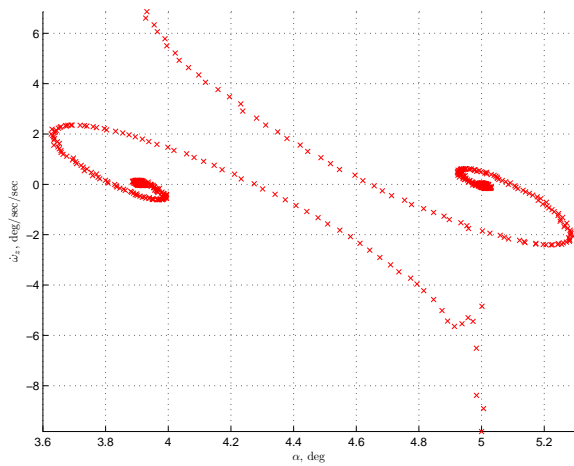
Рис. Б.1. Фазовые траектории, порождающие обучающий набор при ступенчатом тестовом воздействии (точечный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



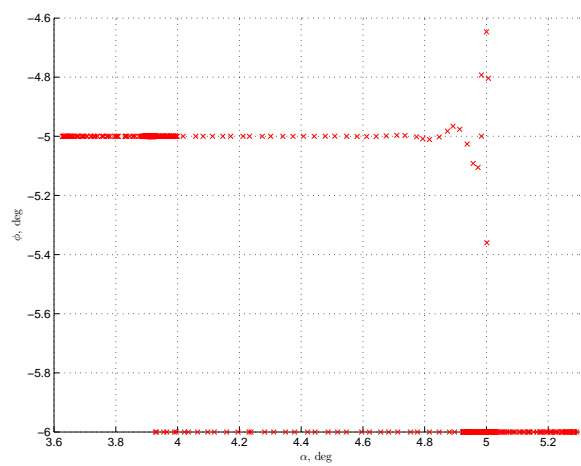
(a)



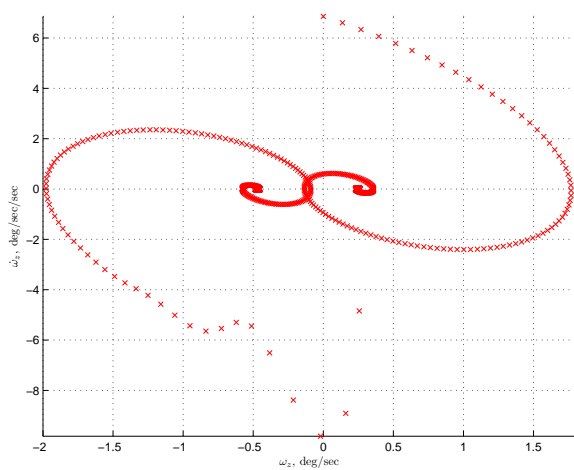
(b)



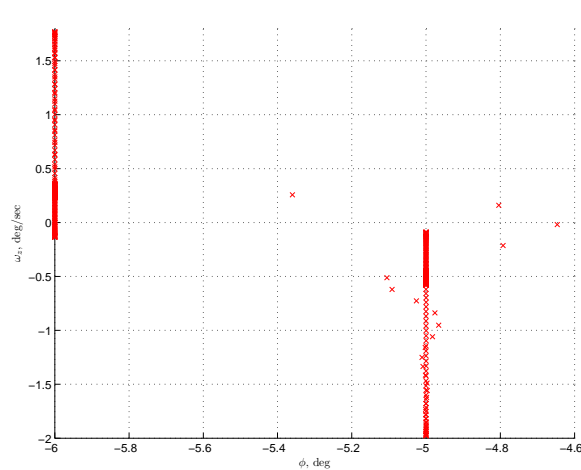
(c)



(d)



(e)



(f)

Рис. Б.2. Диаграммы информативности обучающего набора для ступенчатого тестового воздействия (точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

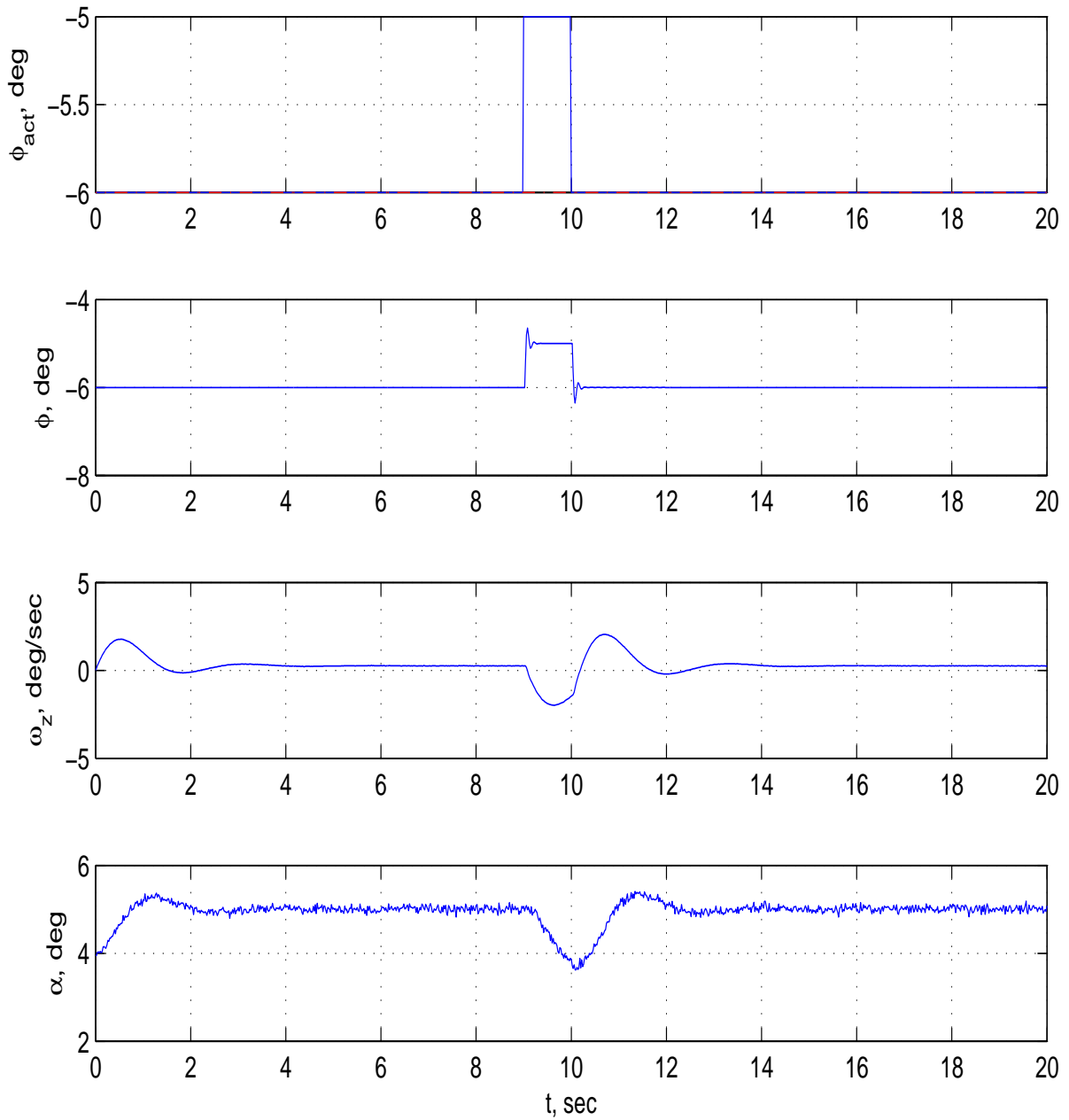
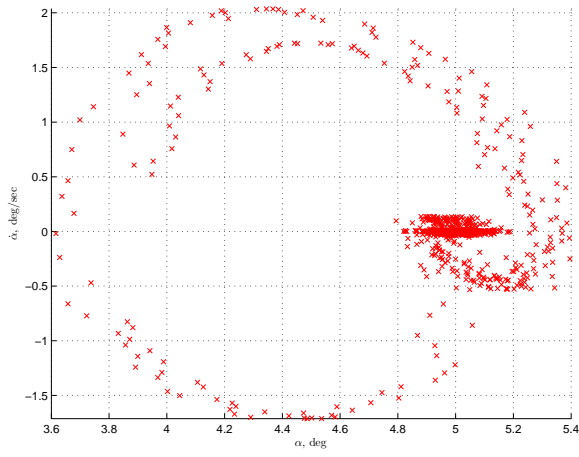
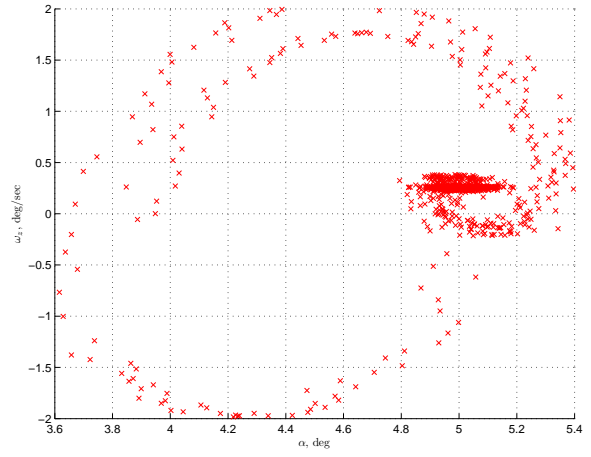


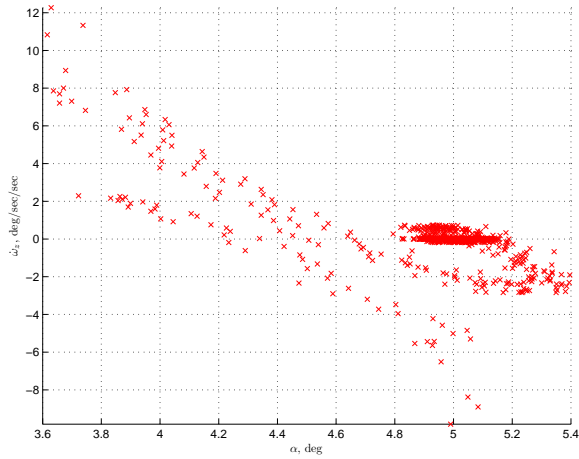
Рис. Б.3. Фазовые траектории, порождающие обучающий набор при импульсном тестовом воздействии (короткий импульс, точечный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



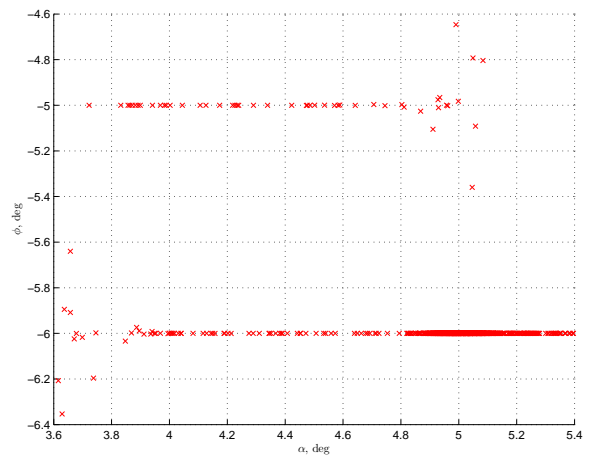
(a)



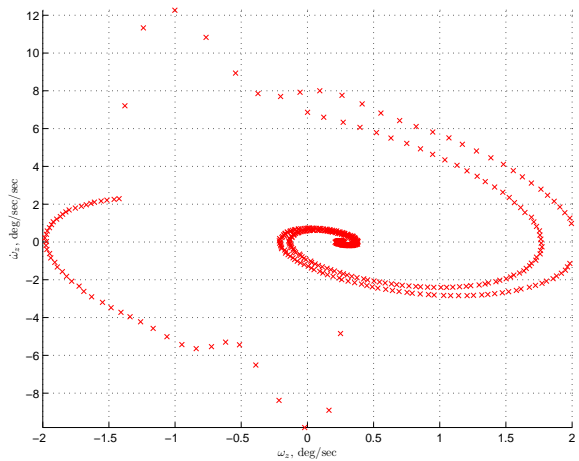
(b)



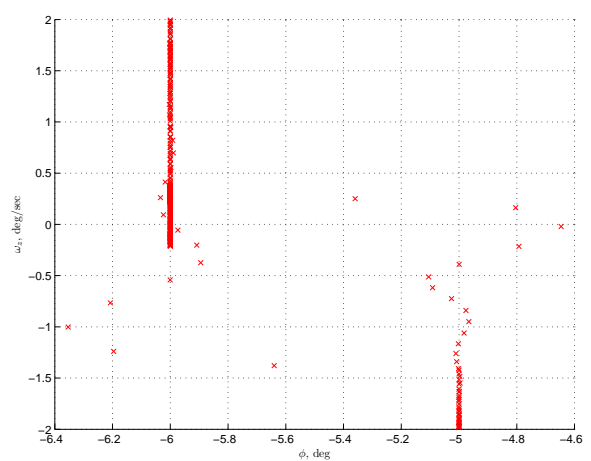
(c)



(d)



(e)



(f)

Рис. Б.4. Диаграммы информативности обучающего набора для импульсного тестового воздействия (короткий импульс, точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

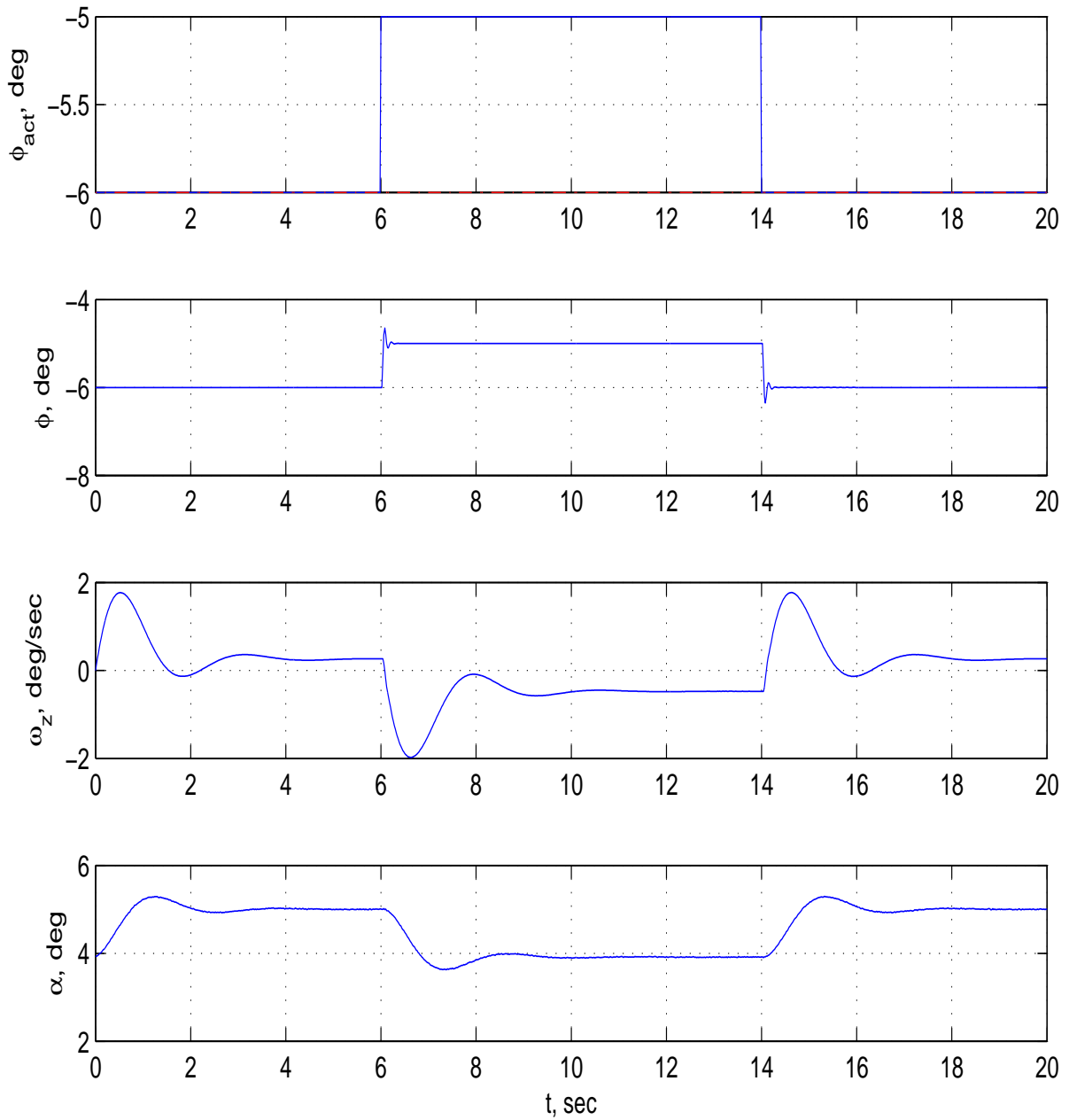
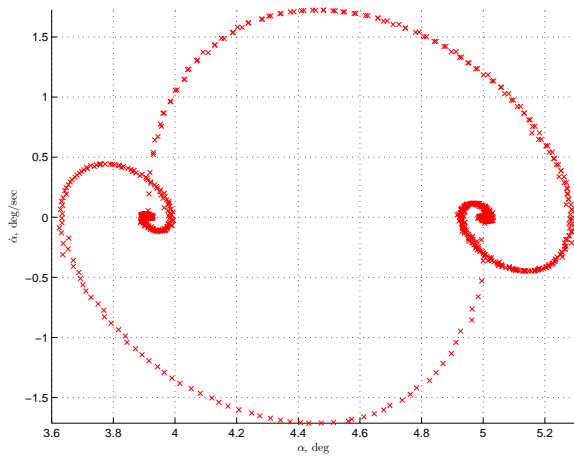
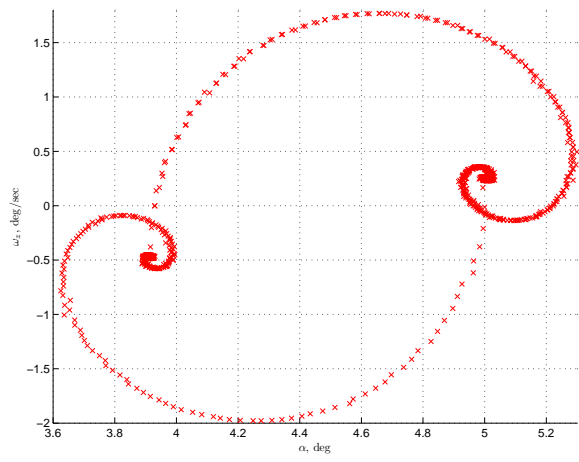


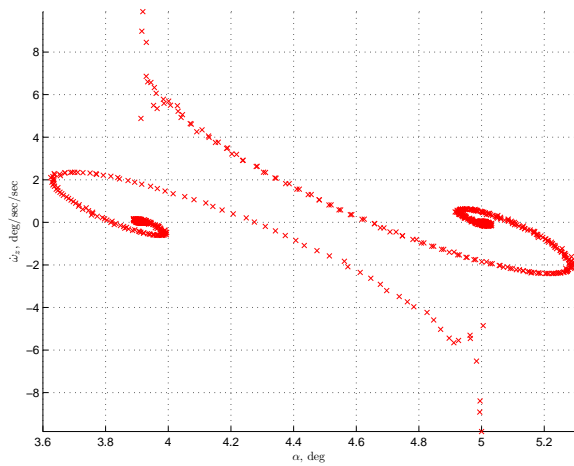
Рис. Б.5. Фазовые траектории, порождающие обучающий набор при импульсном тестовом воздействии (длинный импульс, точечный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



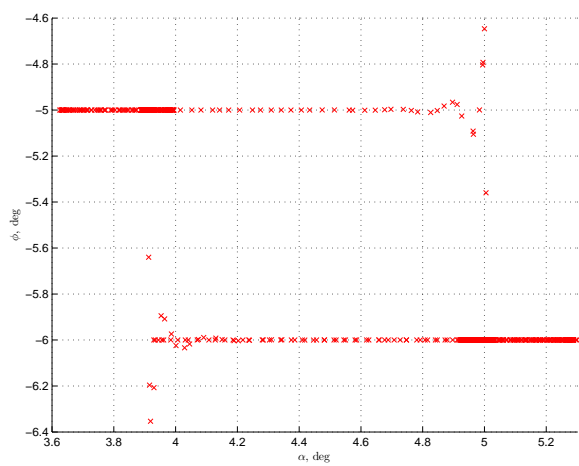
(a)



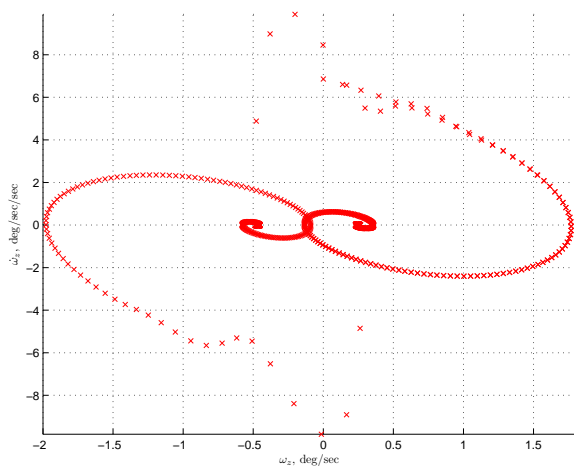
(b)



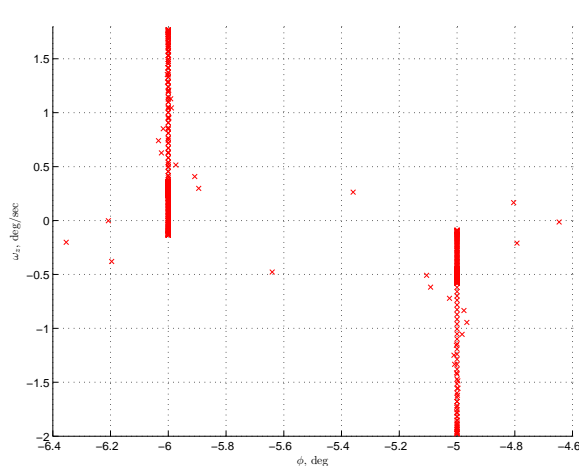
(c)



(d)



(e)



(f)

Рис. Б.6. Диаграммы информативности обучающего набора для импульсного тестового воздействия (длинный импульс, точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

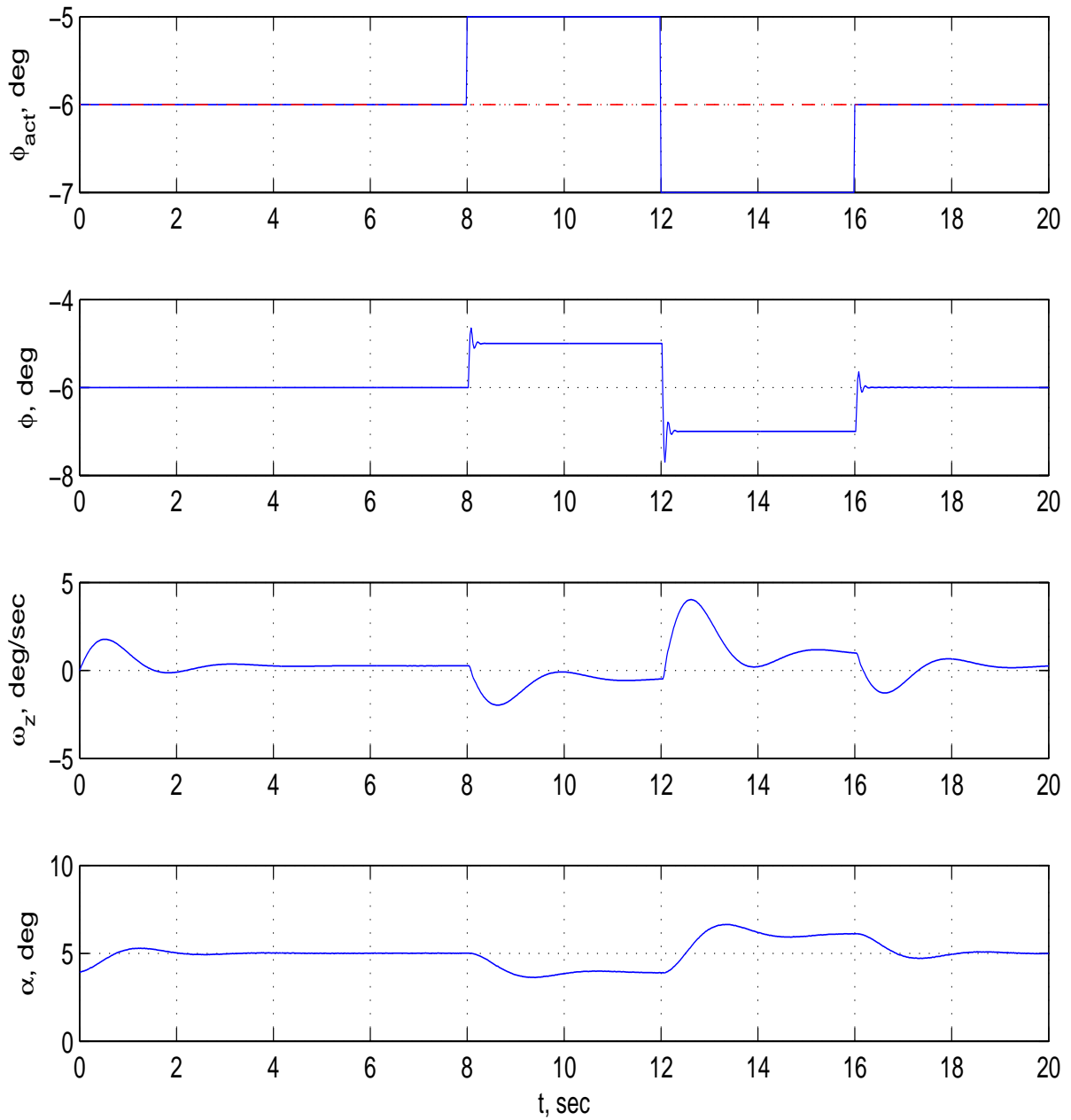
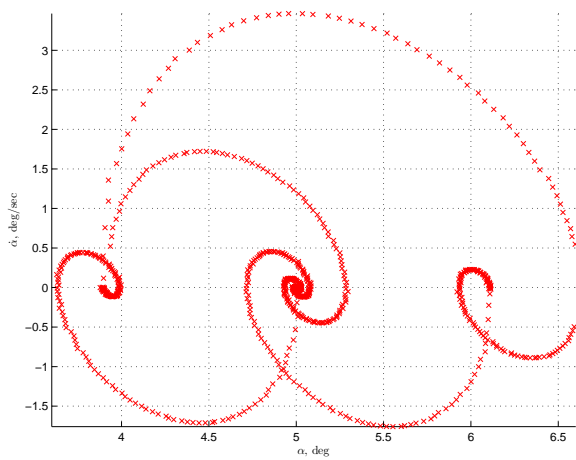
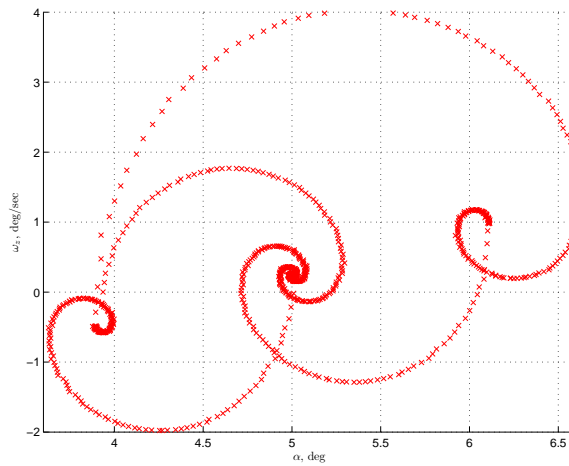


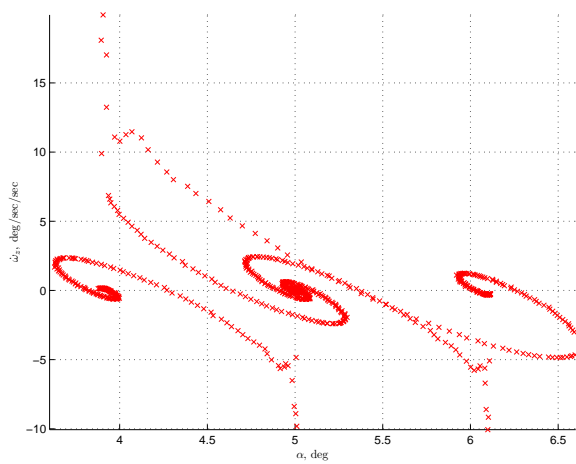
Рис. Б.7. Фазовые траектории, порождающие обучающий набор при тестовом воздействии типа дублет (точечный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



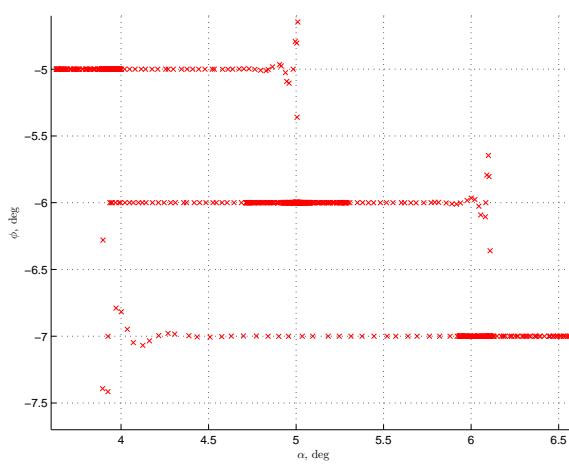
(a)



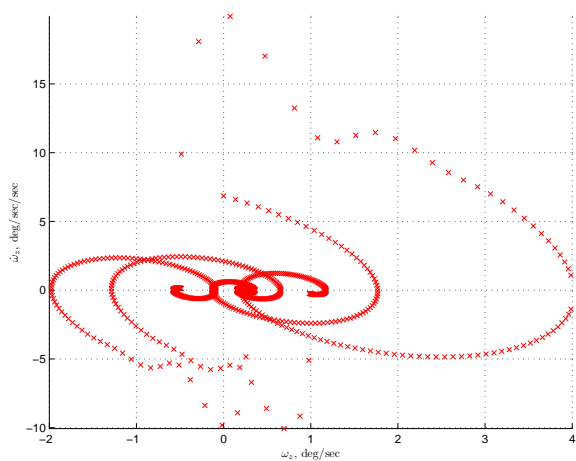
(b)



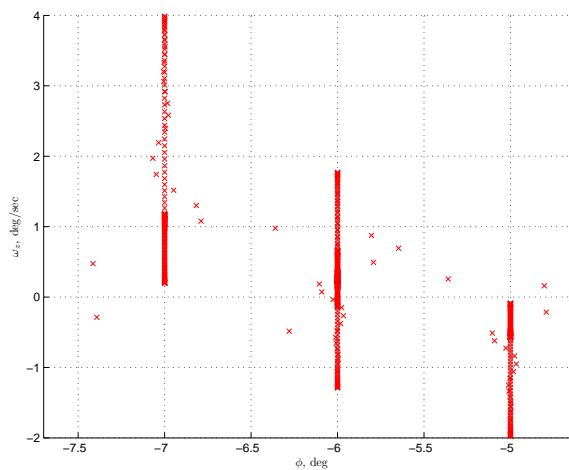
(c)



(d)



(e)



(f)

Рис. Б.8. Диаграммы информативности обучающего набора для тестового воздействия типа дублет (точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

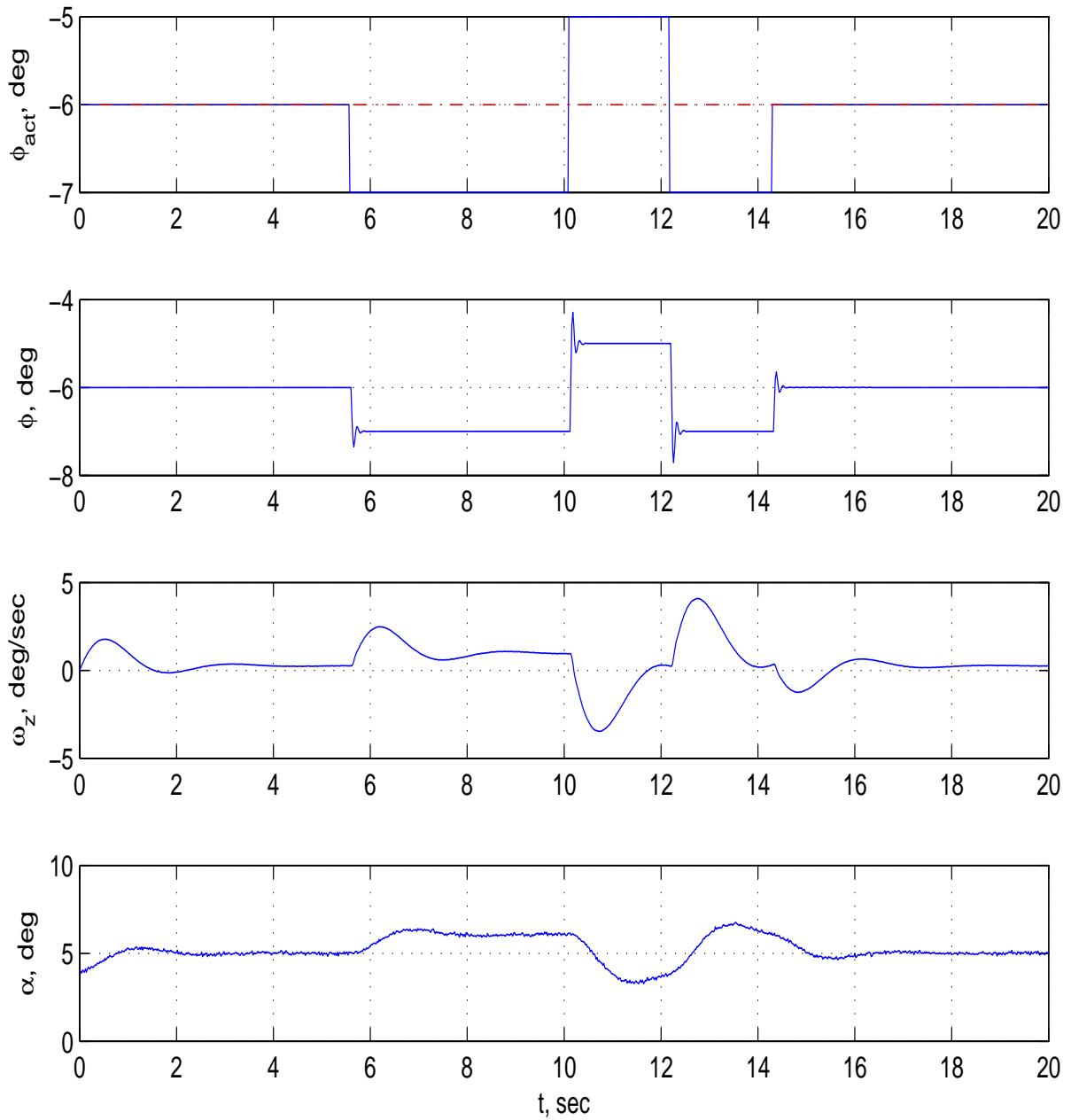
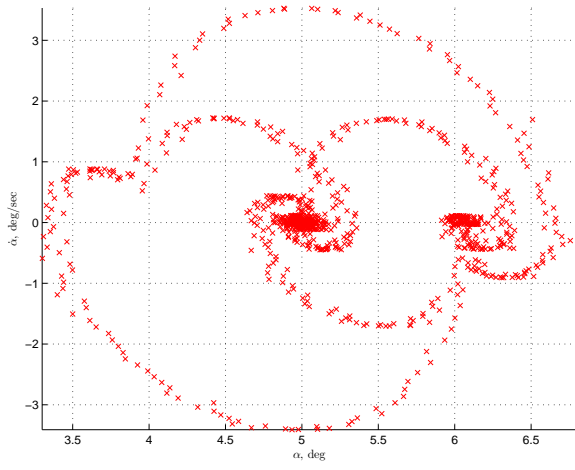
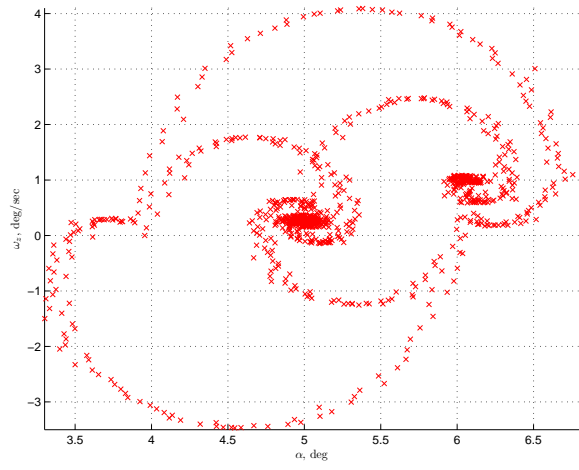


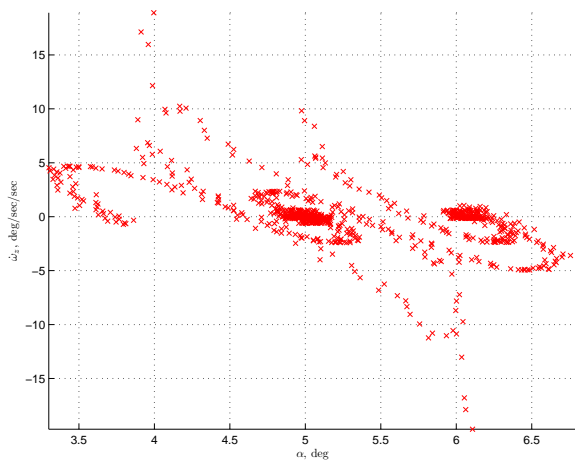
Рис. Б.9. Фазовые траектории, порождающие обучающий набор при тестовом воздействии типа 2-1-1 (точный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



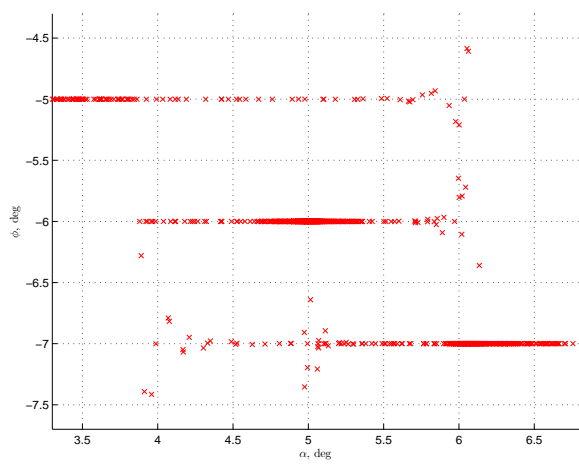
(a)



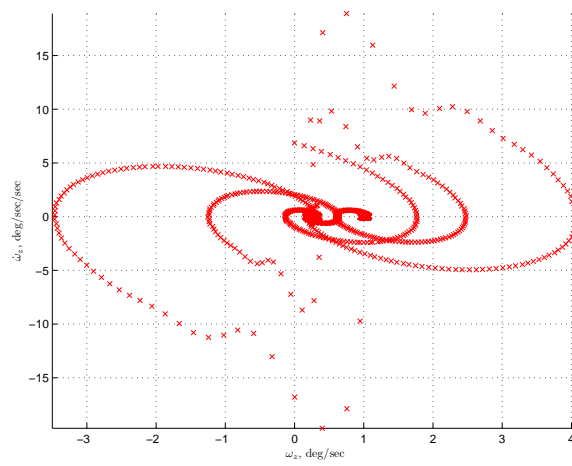
(b)



(c)



(d)



(e)

Рис. Б.10. Диаграммы информативности обучающего набора для тестового воздействия типа 2-1-1 (точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$

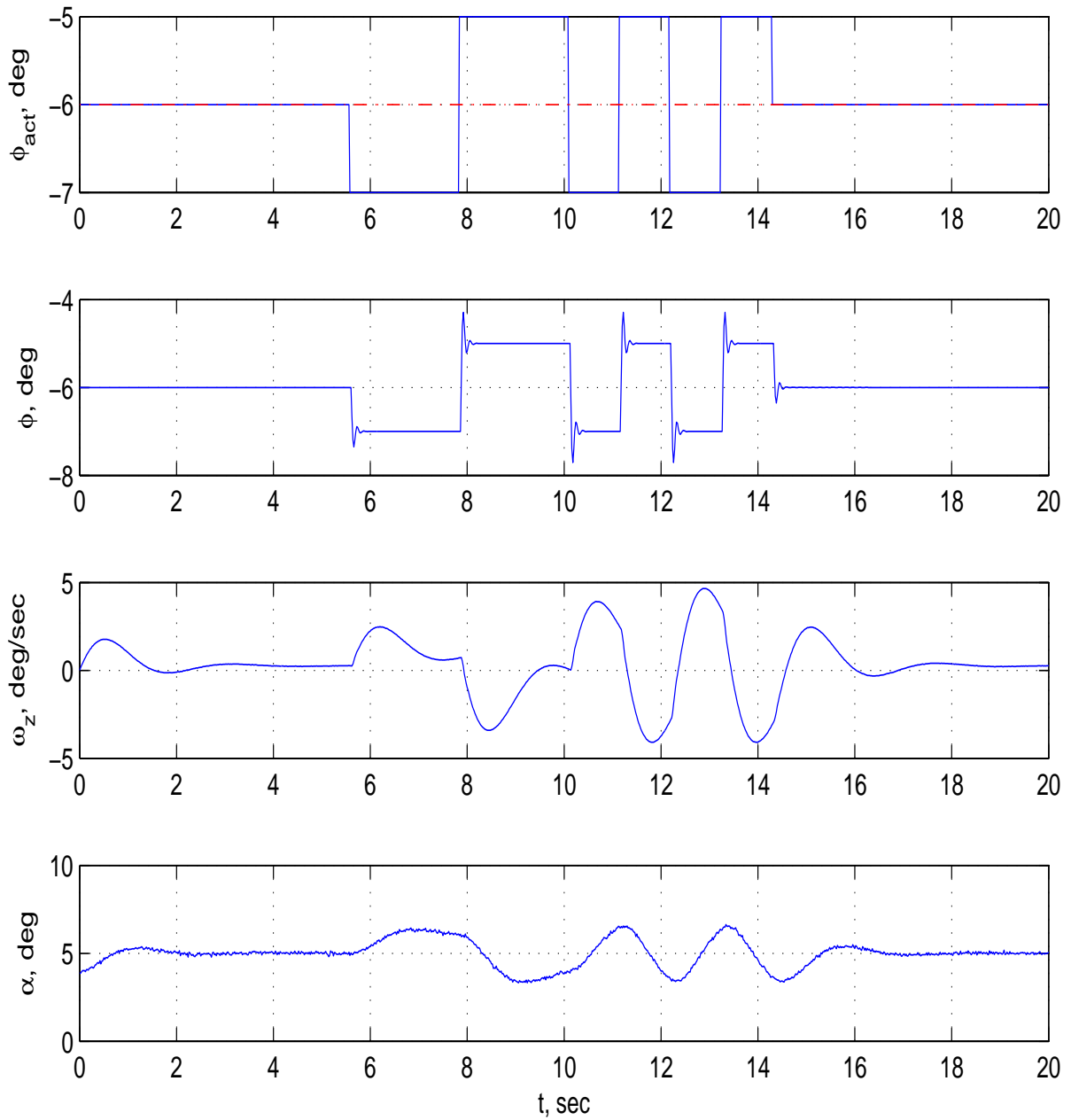
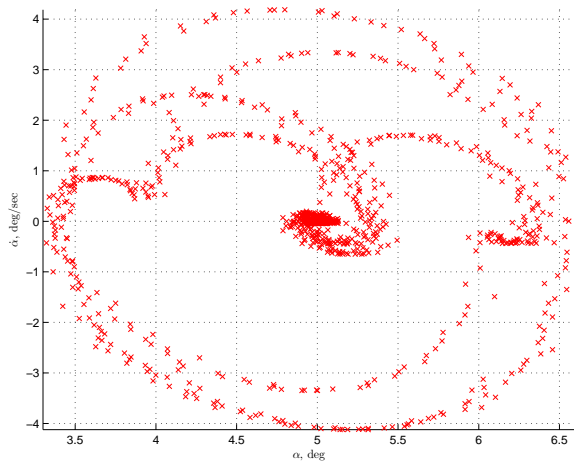
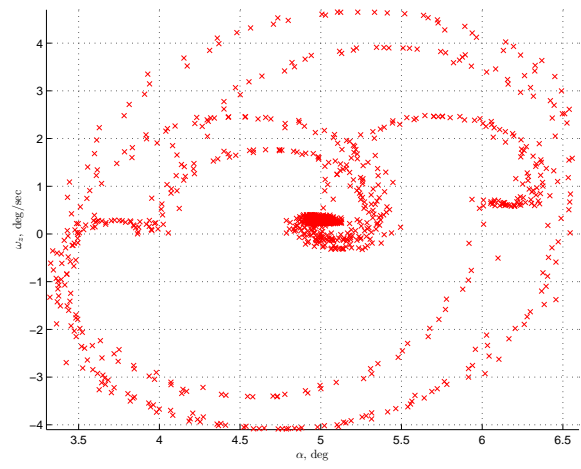


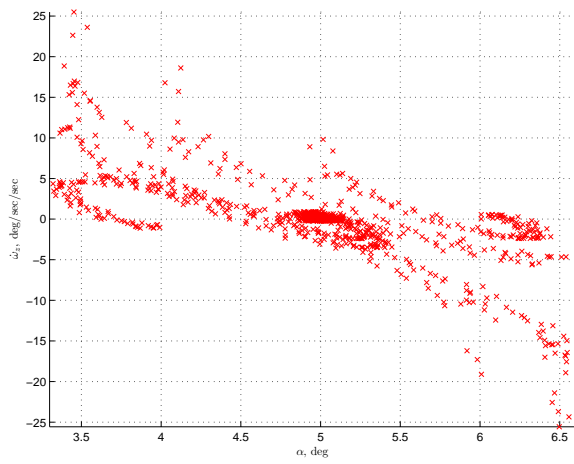
Рис. Б.11. Фазовые траектории, порождающие обучающий набор при тестовом воздействии типа 2-2-1-1-1-1 (точечный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



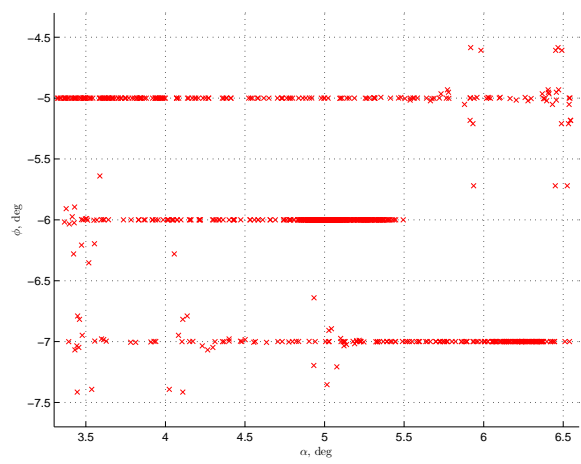
(a)



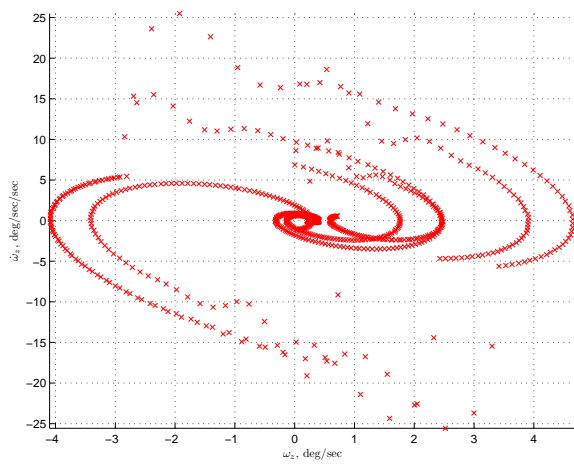
(b)



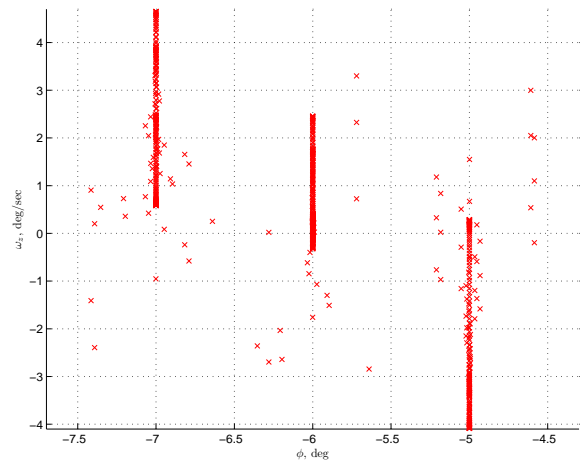
(c)



(d)



(e)



(f)

Рис. Б.12. Диаграммы информативности обучающего набора для тестового воздействия типа 2-2-1-1-1 (точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \ddot{\alpha})$; (d) — (α, ϕ) ; (e) — $(\omega_z, \ddot{\alpha})$; (f) — (ϕ, ω_z)

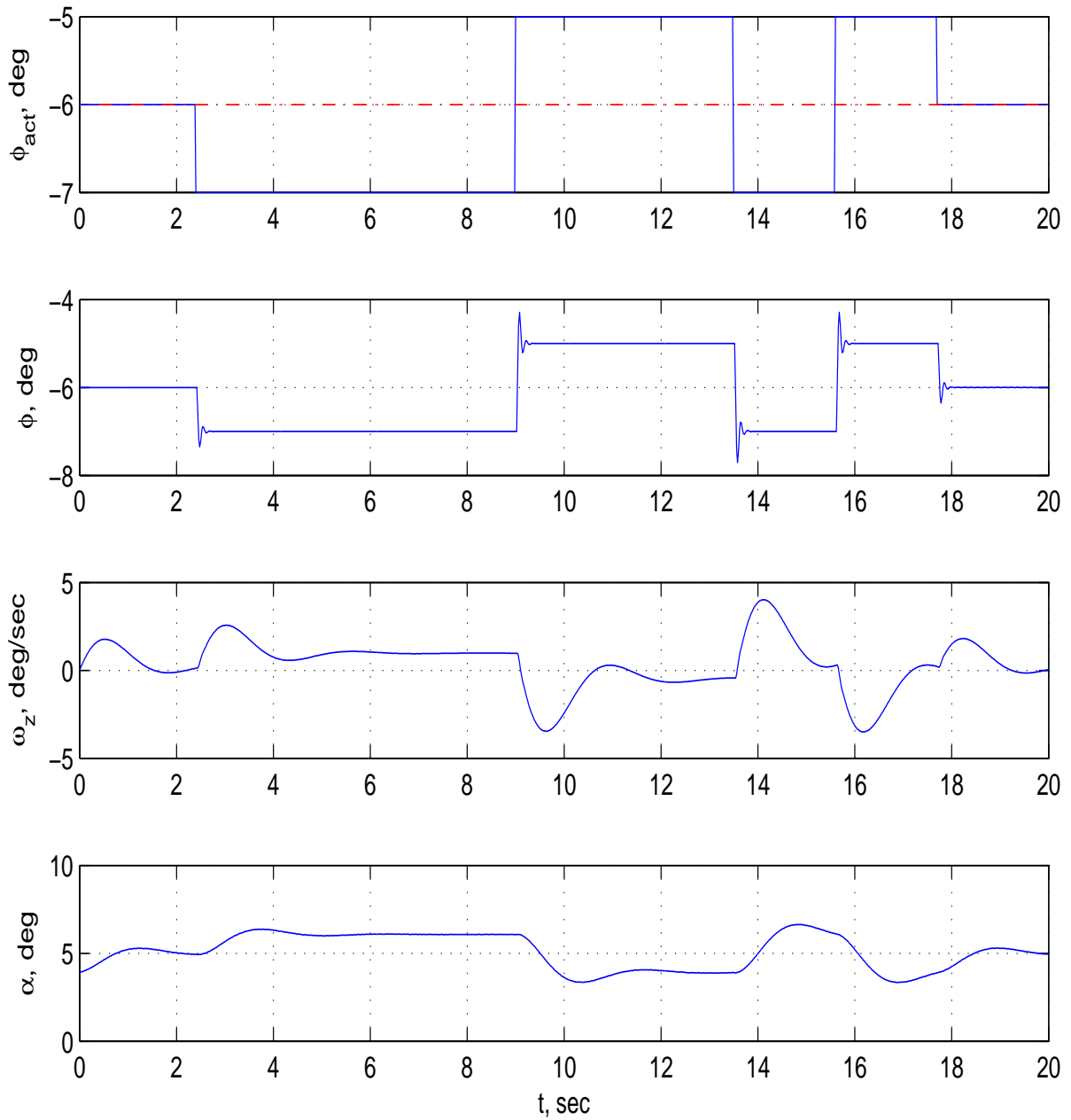
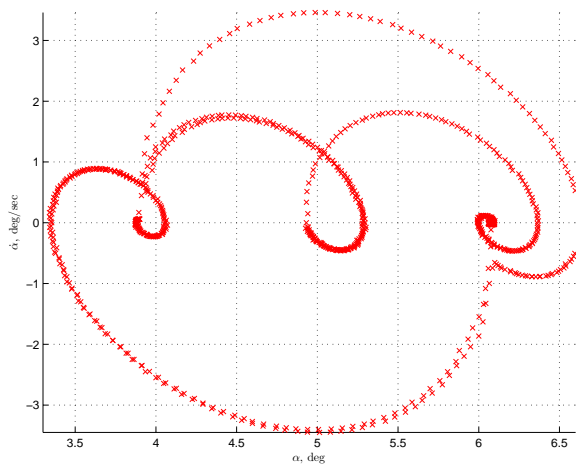
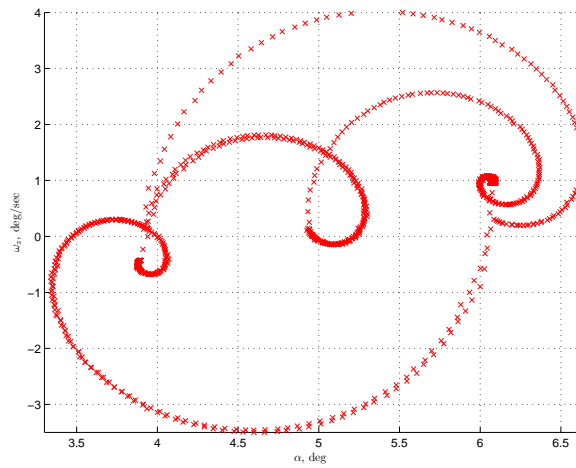


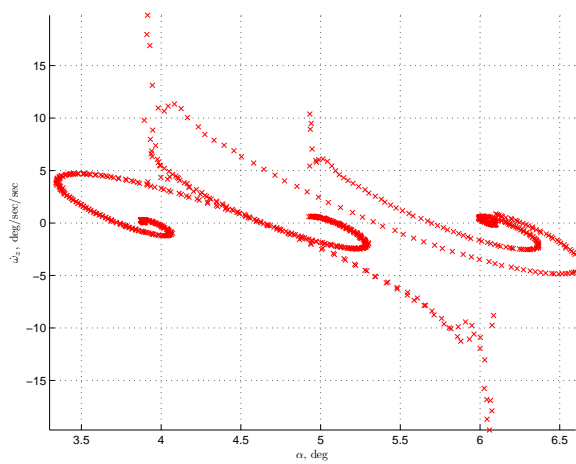
Рис. Б.13. Фазовые траектории, порождающие обучающий набор при тестовом воздействии типа 3-2-1-1 (точечный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



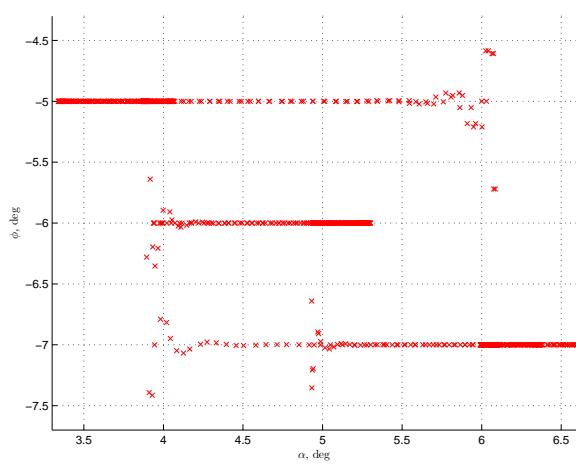
(a)



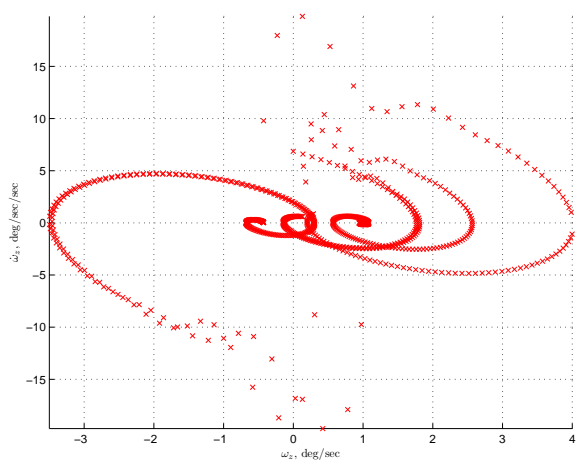
(b)



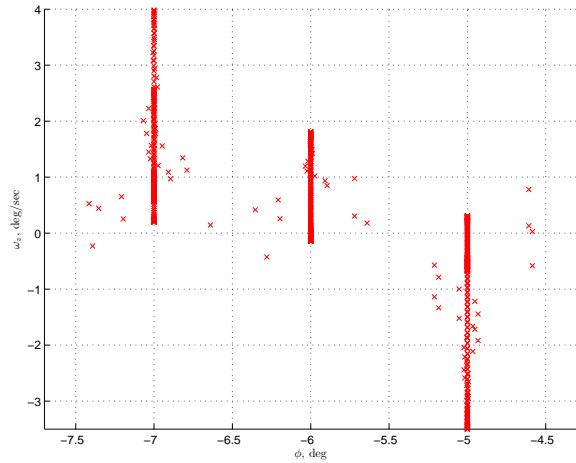
(c)



(d)



(e)



(f)

Рис. Б.14. Диаграммы информативности обучающего набора для тестового воздействия типа 3-2-1-1 (точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \ddot{\alpha})$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

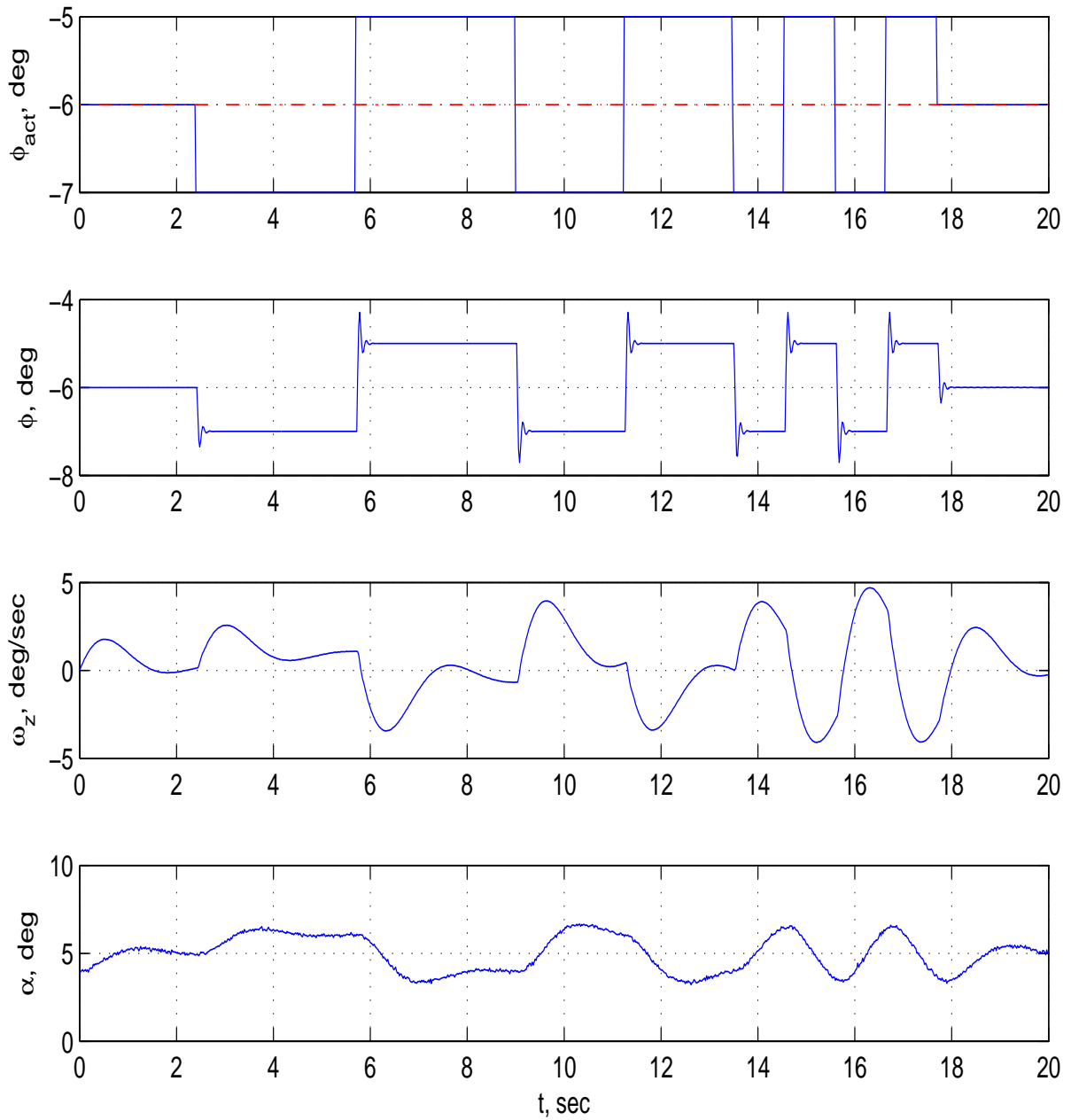
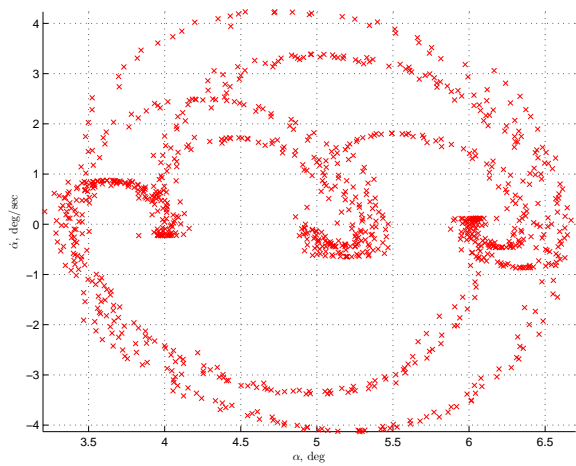
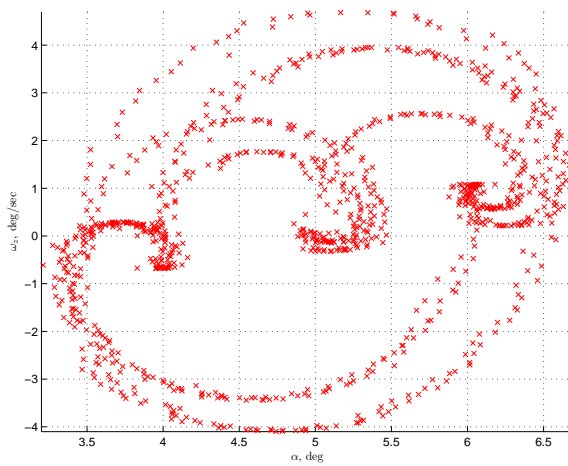


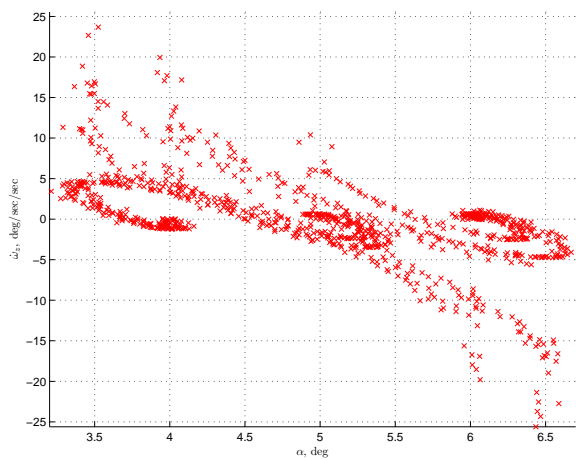
Рис. Б.15. Фазовые траектории, порождающие обучающий набор при тестовом воздействии типа 3-3-2-2-1-1-1-1-1 (точечный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



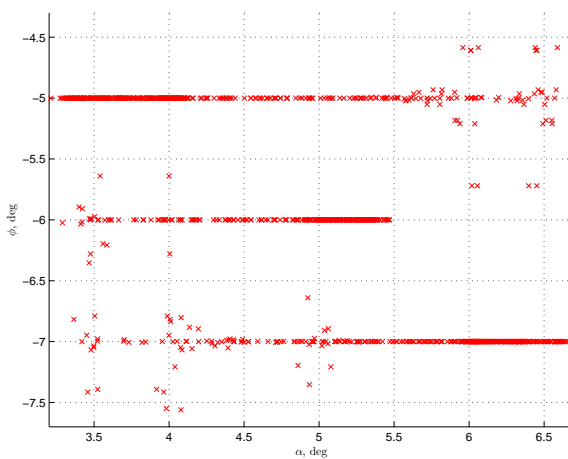
(a)



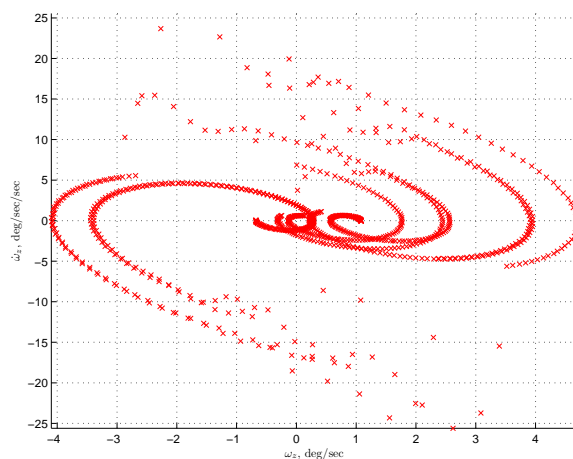
(b)



(c)



(d)



(e)

Рис. Б.16. Диаграммы информативности обучающего набора для тестового воздействия типа 3-3-2-2-1-1-1-1-1 (точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$

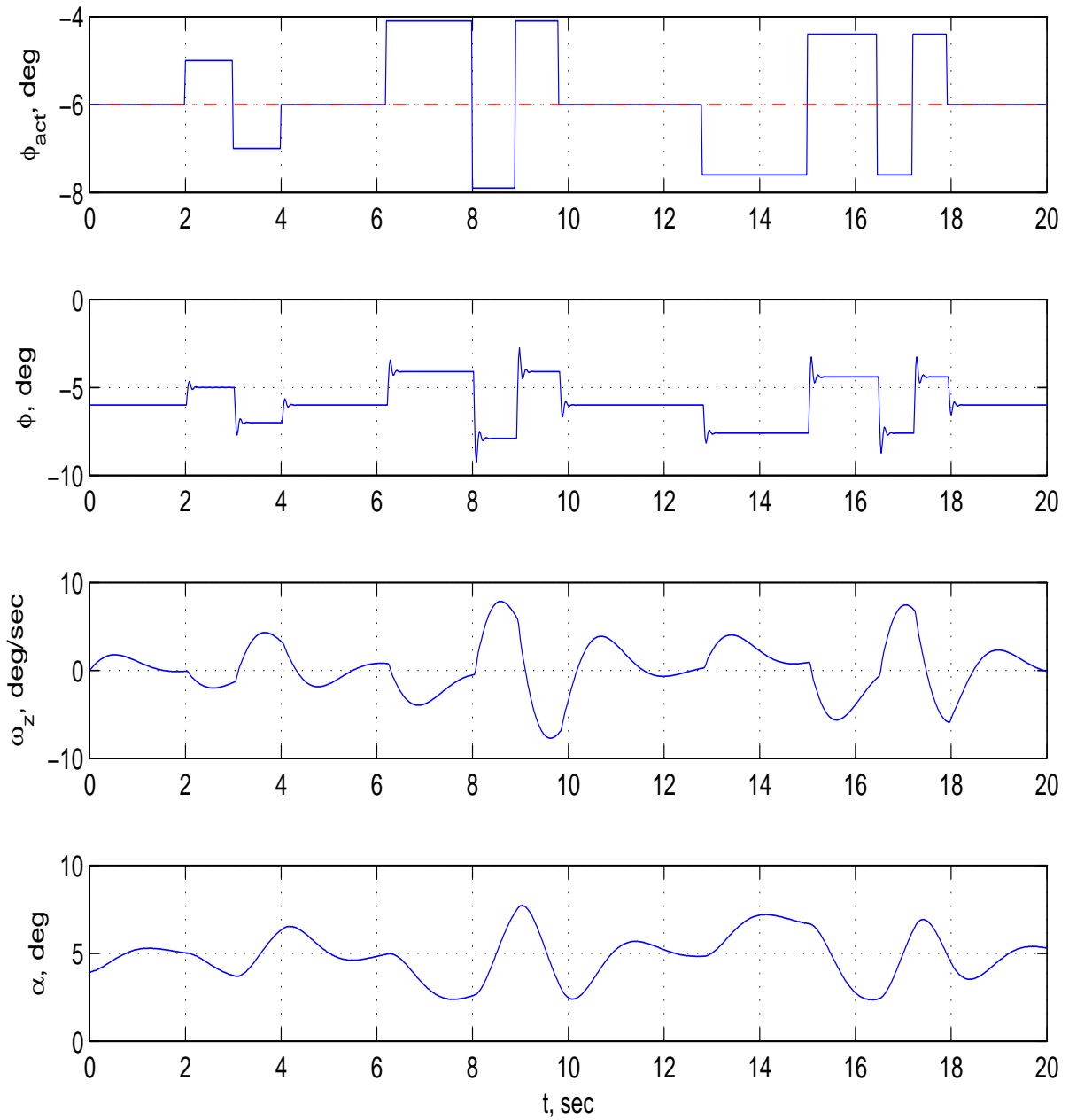


Рис. Б.17. Фазовые траектории, порождающие обучающий набор lkz комбинированного тестового воздействия типа дублет+(2-1-1)+(3-2-1-1) (точечный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО

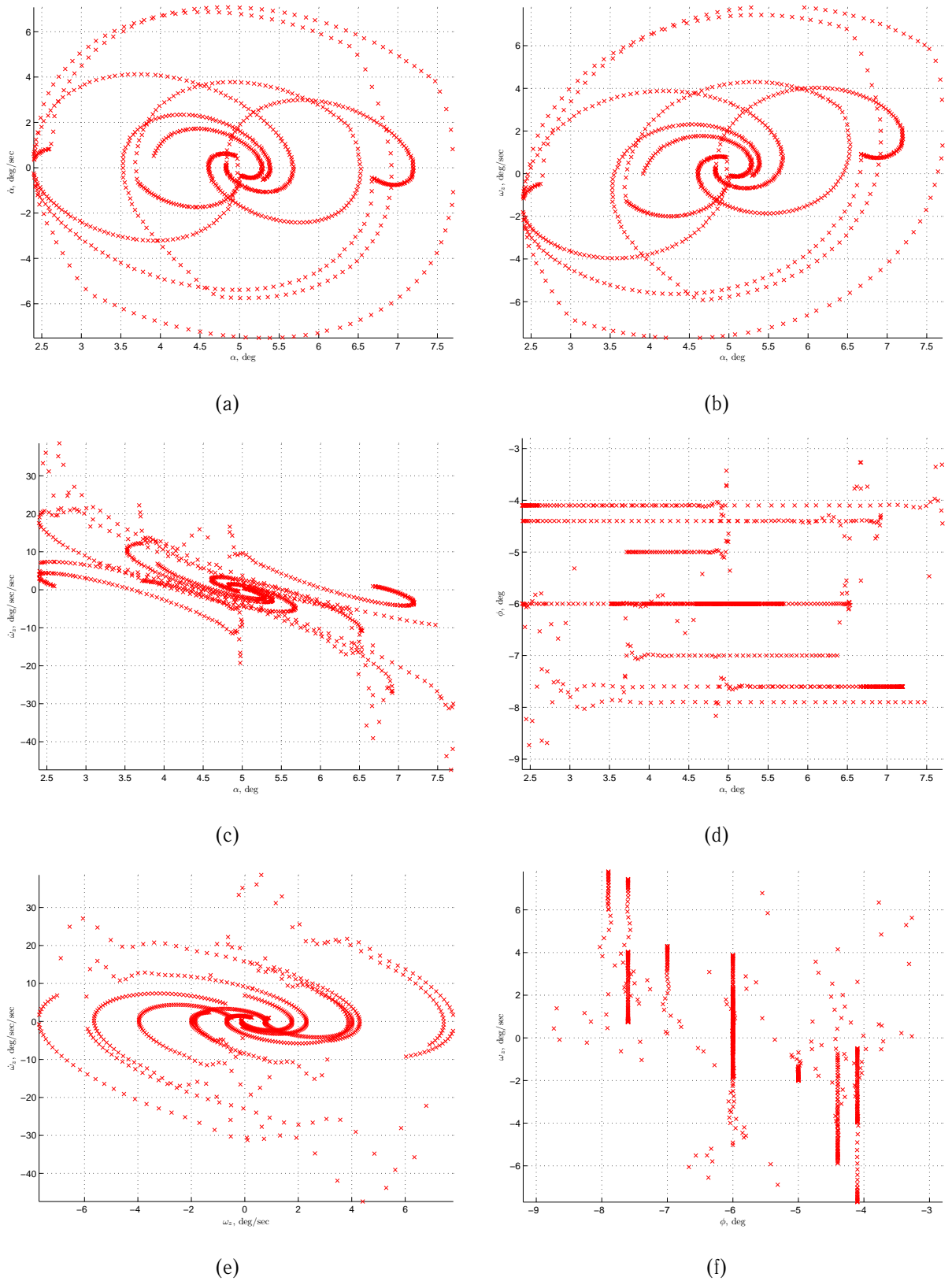


Рис. Б.18. Диаграммы информативности обучающего набора для комбинированного тестового воздействия типа дублет+(2-1-1)+(3-2-1-1) (точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — $(\alpha, \dot{\omega}_z)$; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

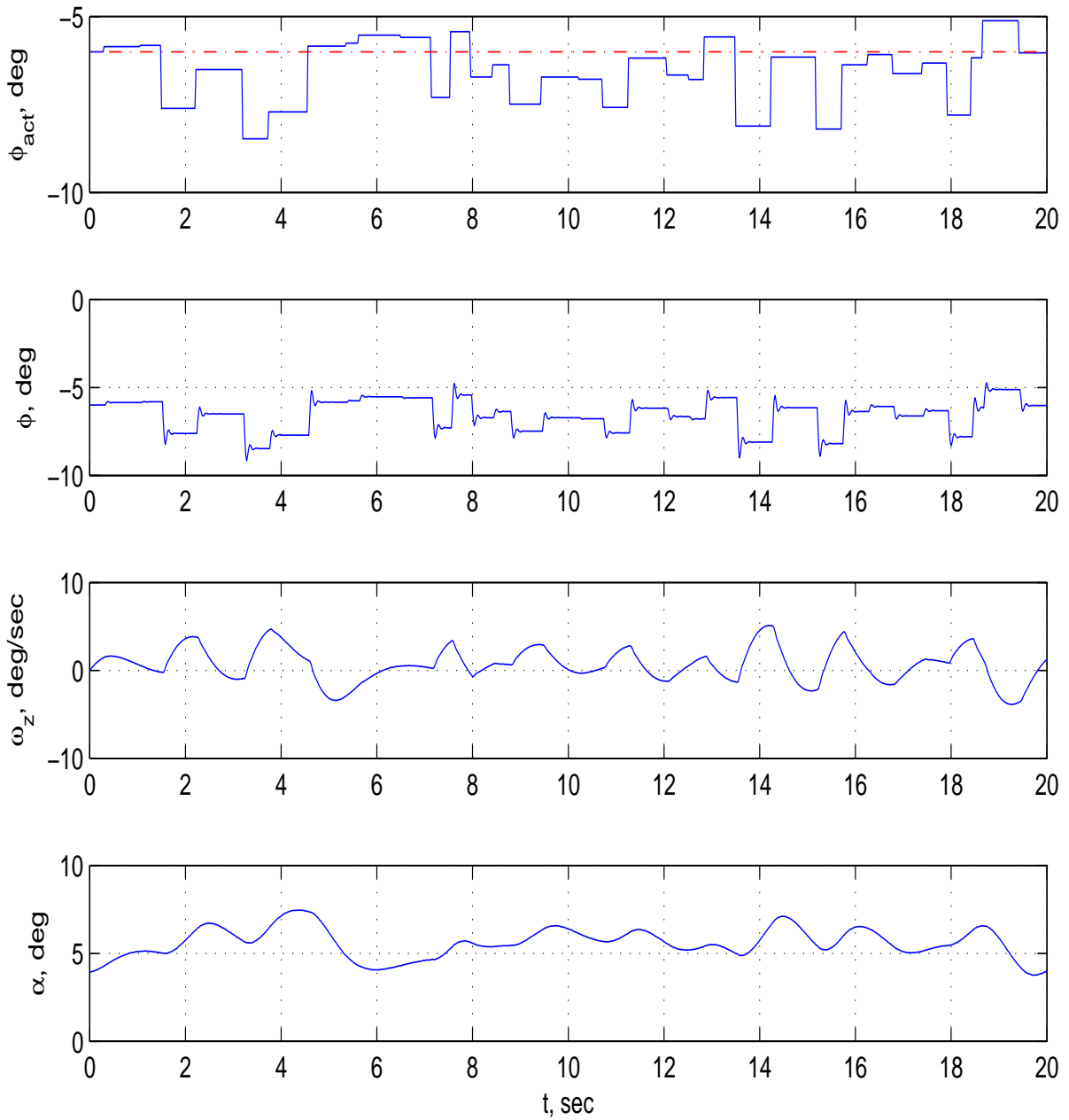
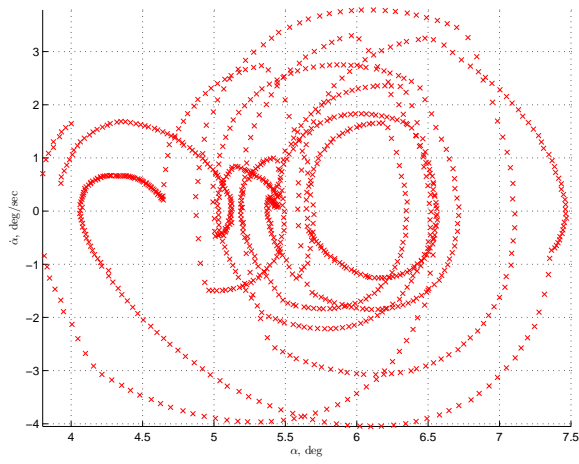
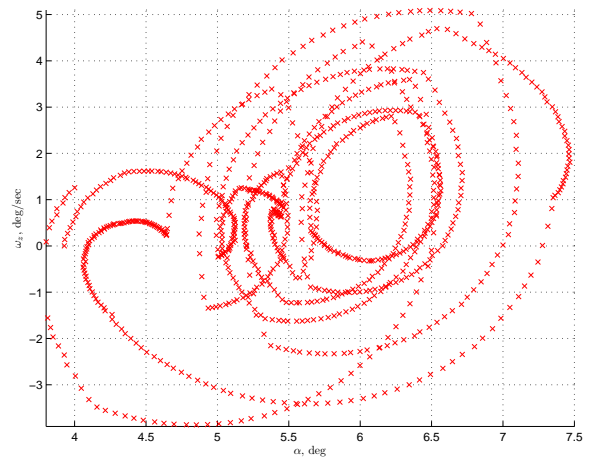


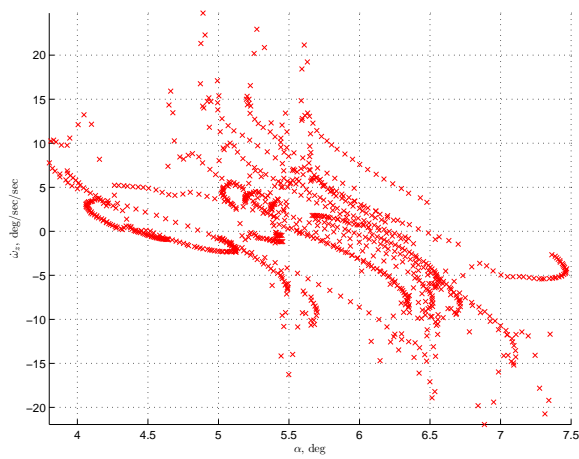
Рис. Б.19. Фазовые траектории, порождающие обучающий набор при случайном тестовом воздействии типа (точечный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



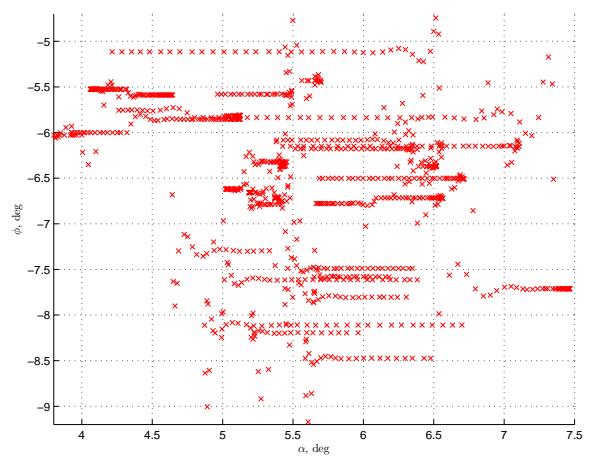
(a)



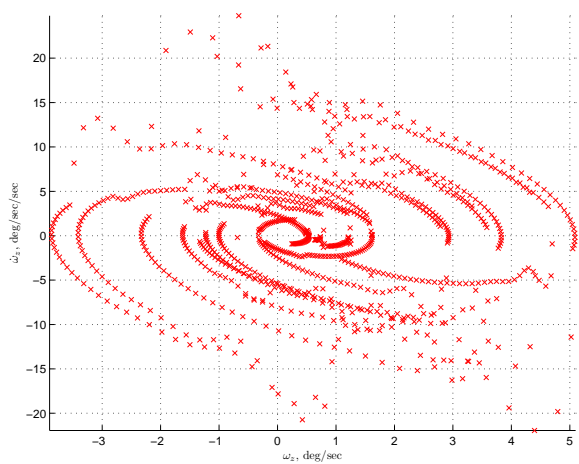
(b)



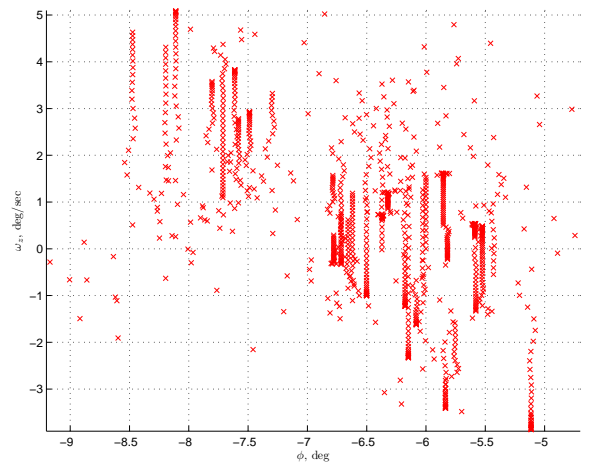
(c)



(d)



(e)



(f)

Рис. Б.20. Диаграммы информативности обучающего набора для случайного тестового воздействия (точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

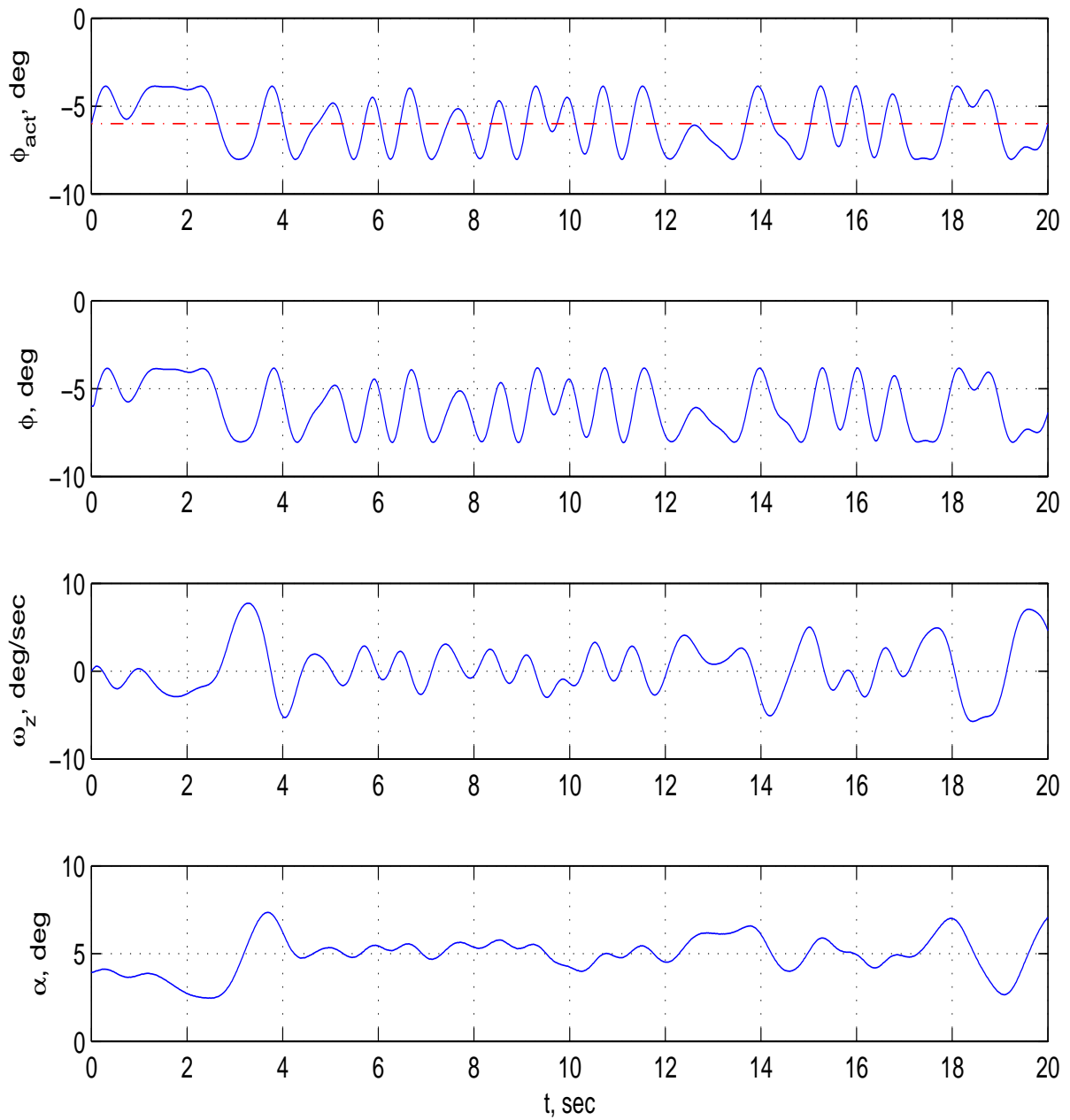
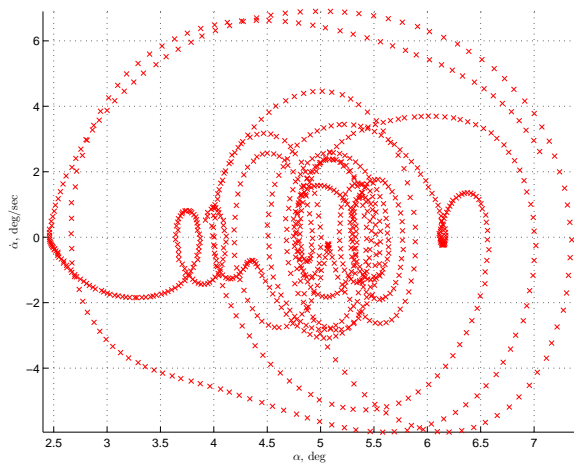
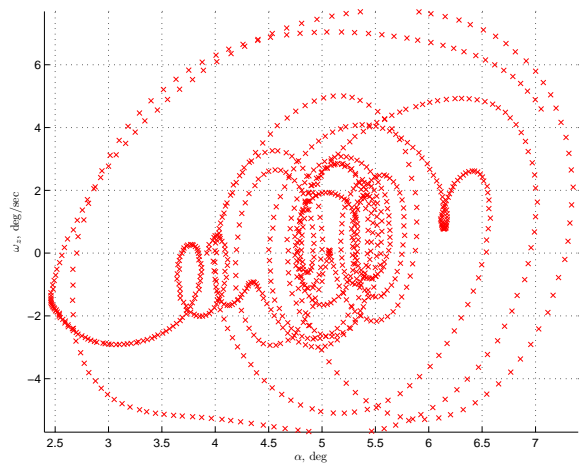


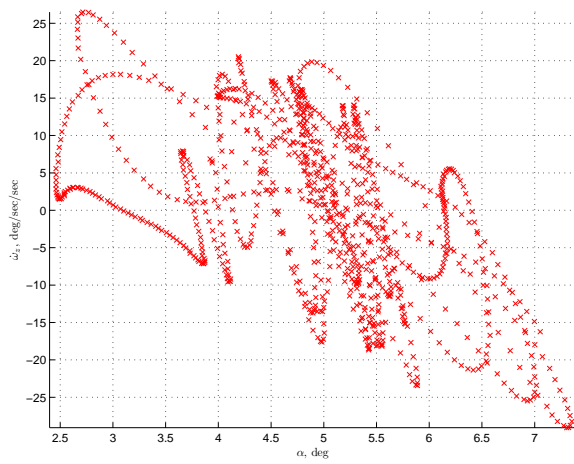
Рис. Б.21. Фазовые траектории, порождающие обучающий набор при полигармоническом тестовом воздействии типа (точечный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



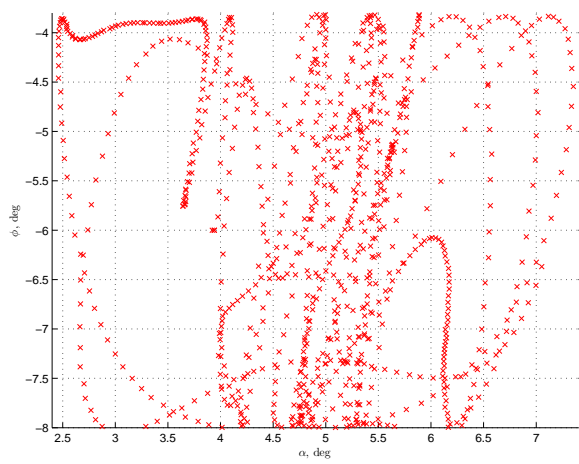
(a)



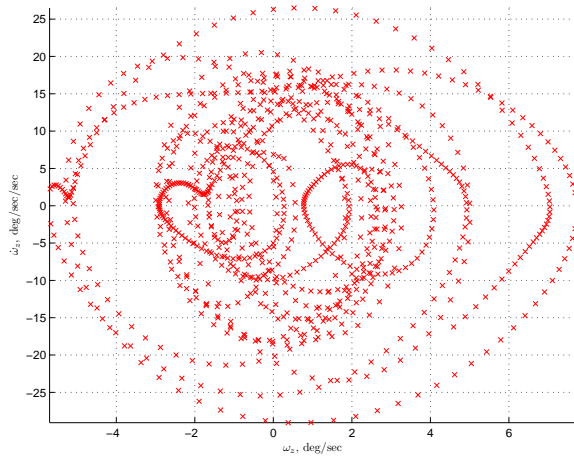
(b)



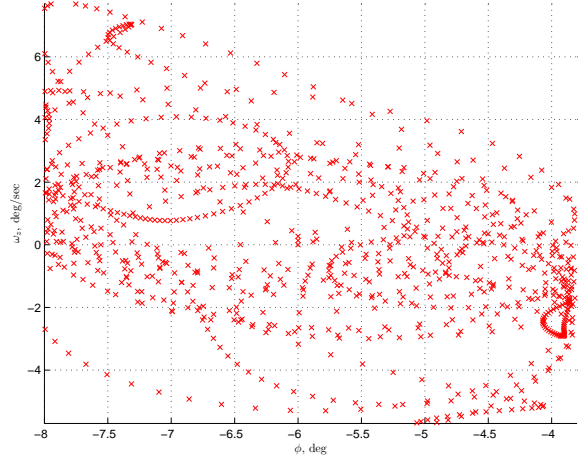
(c)



(d)



(e)



(f)

Рис. Б.22. Диаграммы информативности обучающего набора для полигармонического тестового воздействия (точечный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

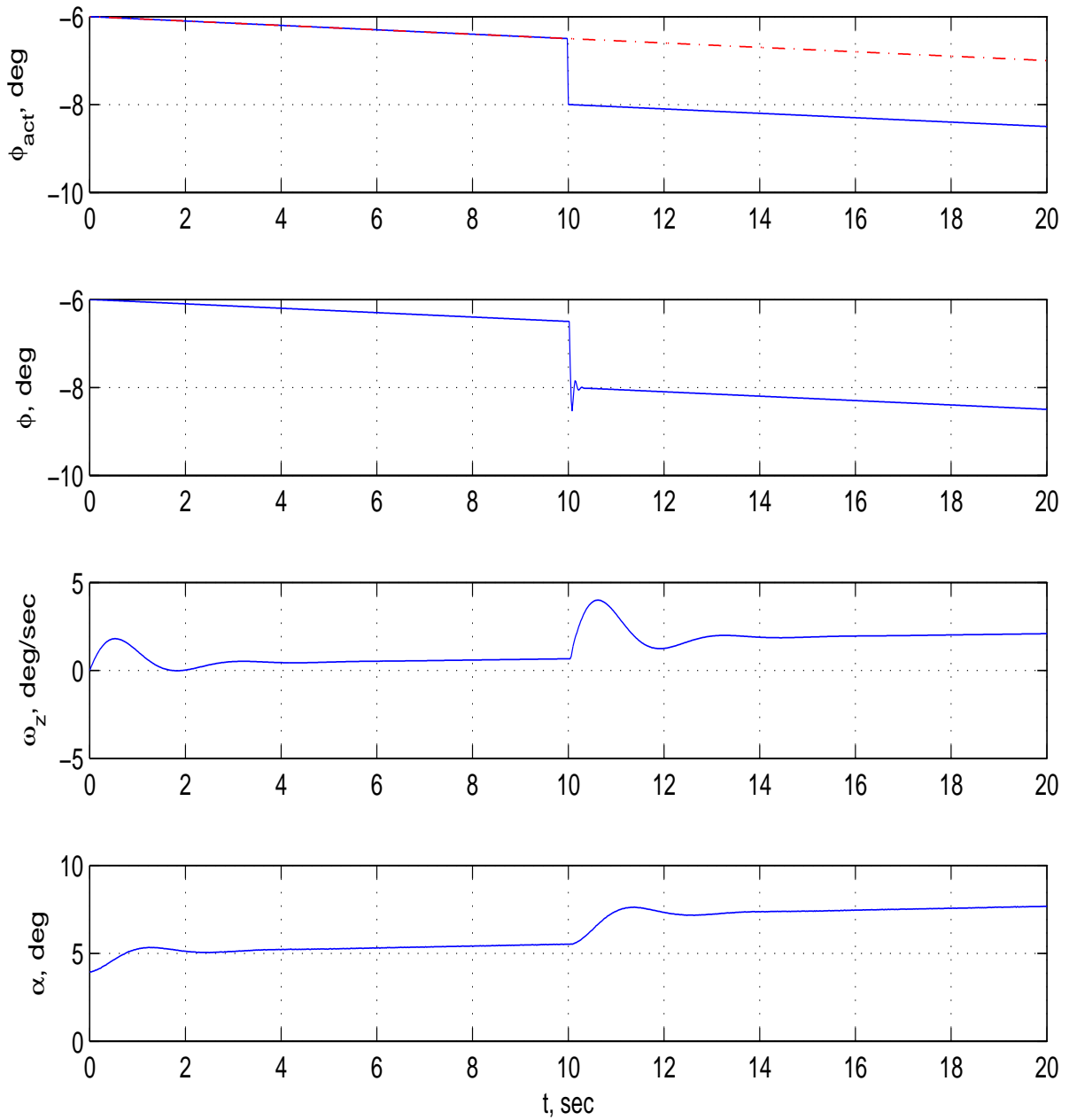
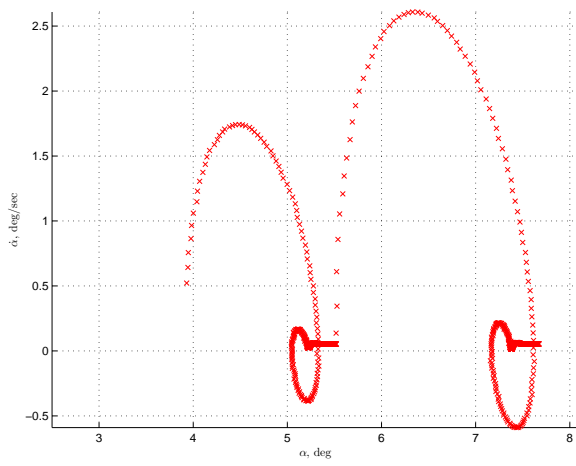
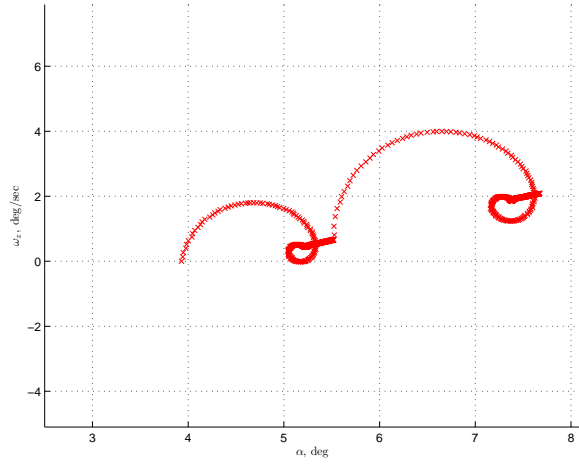


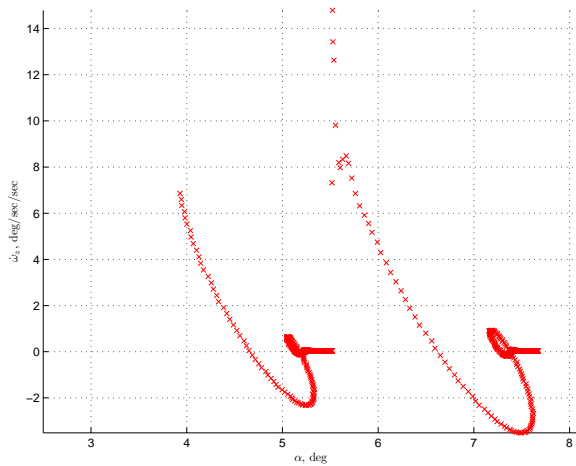
Рис. Б.23. Фазовые траектории, порождающие обучающий набор при ступенчатом тестовом воздействии (монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



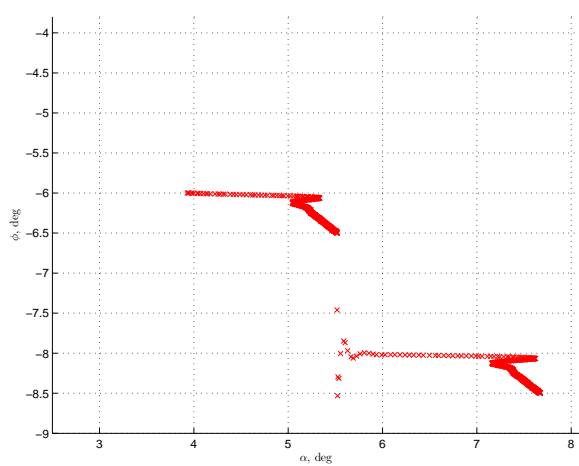
(a)



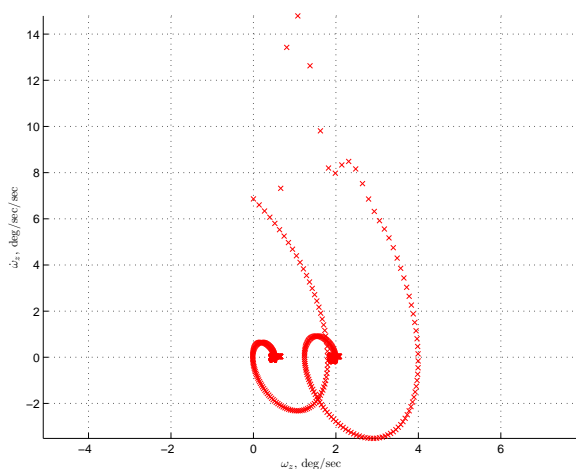
(b)



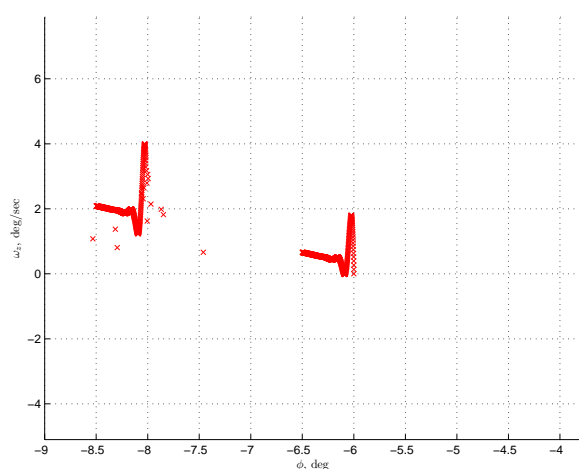
(c)



(d)



(e)



(f)

Рис. Б.24. Диаграммы информативности обучающего набора для ступенчатого тестового воздействия (монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

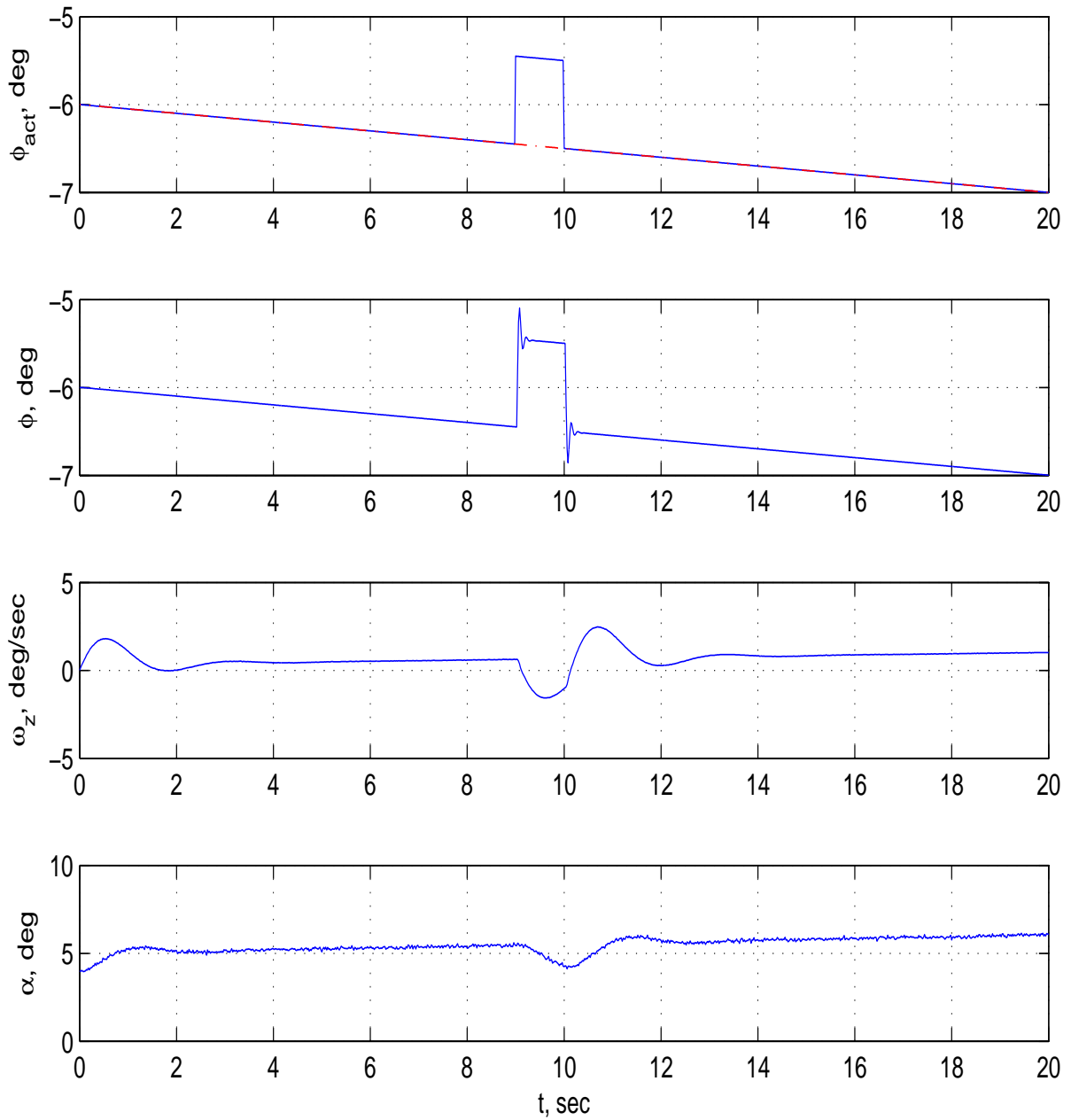
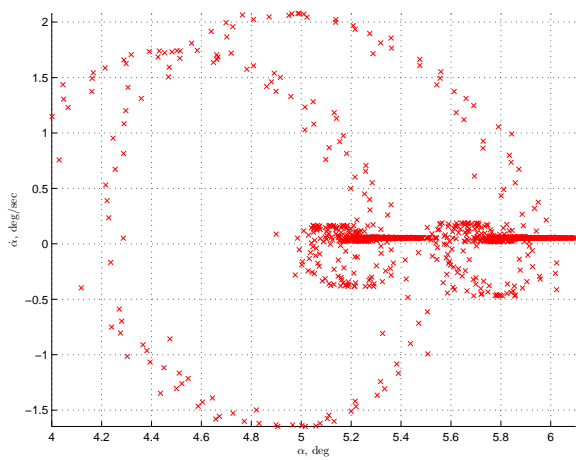
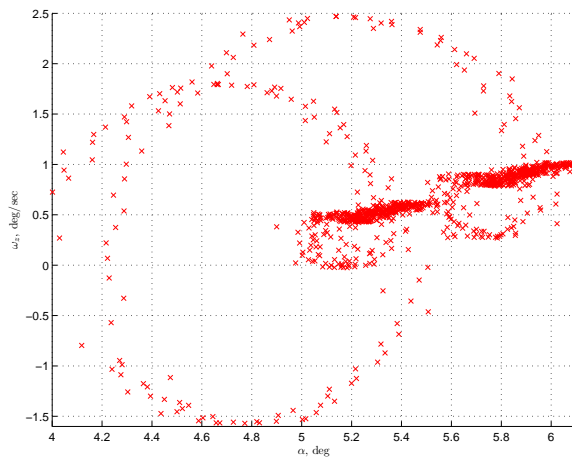


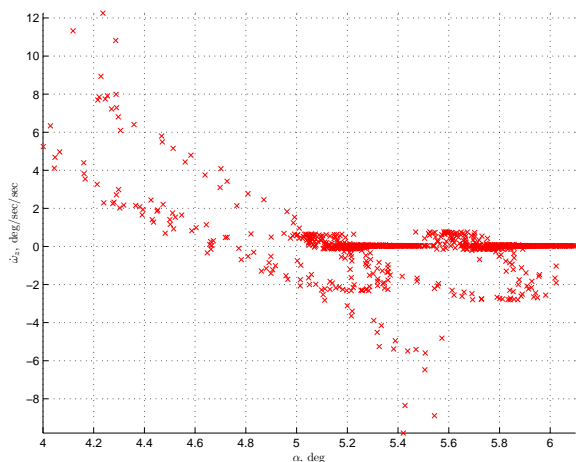
Рис. Б.25. Фазовые траектории, порождающие обучающий набор при импульсном тестовом воздействии (короткий импульс, монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



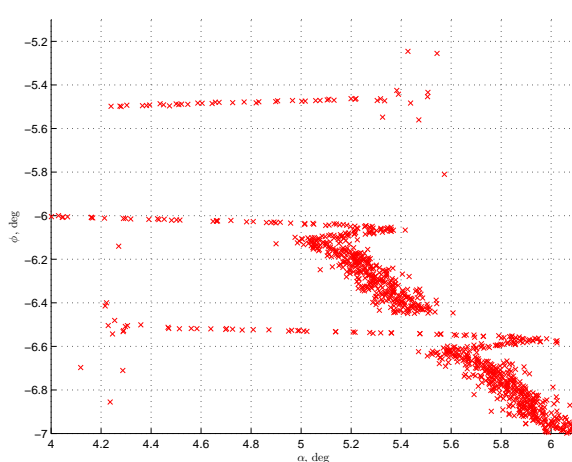
(a)



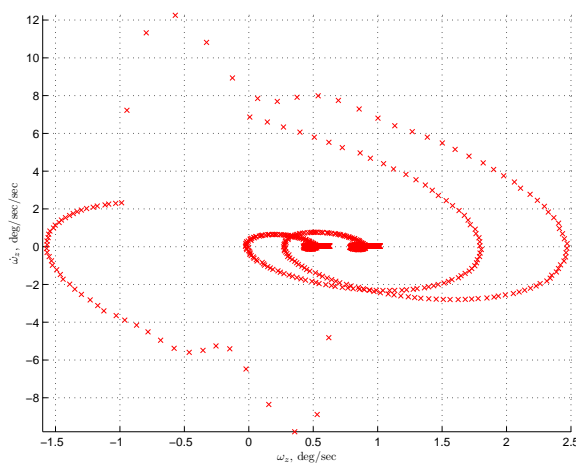
(b)



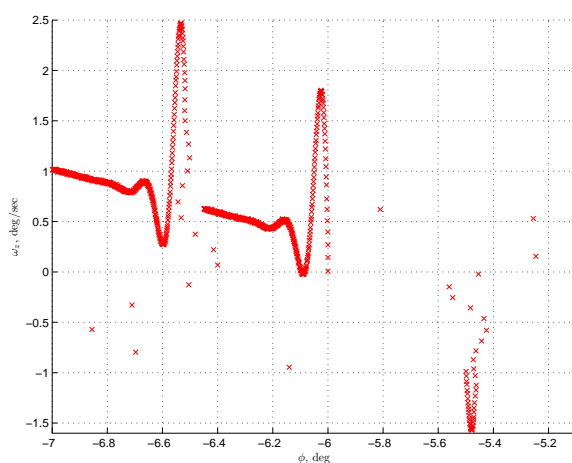
(c)



(d)



(e)



(f)

Рис. Б.26. Диаграммы информативности обучающего набора для импульсного тестового воздействия (короткий импульс, монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

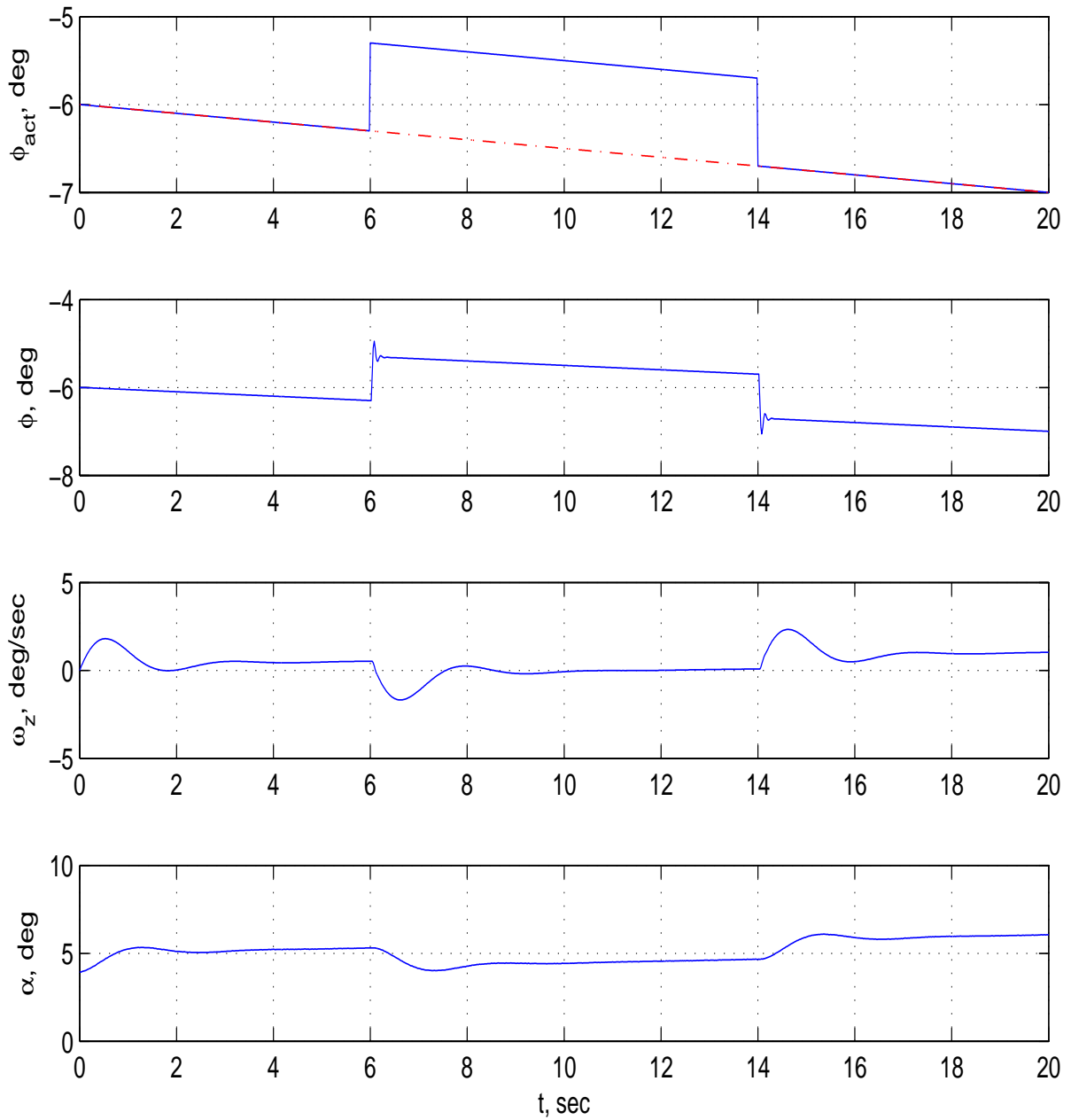


Рис. Б.27. Фазовые траектории, порождающие обучающий набор при импульсном тестовом воздействии (длинный импульс, монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО

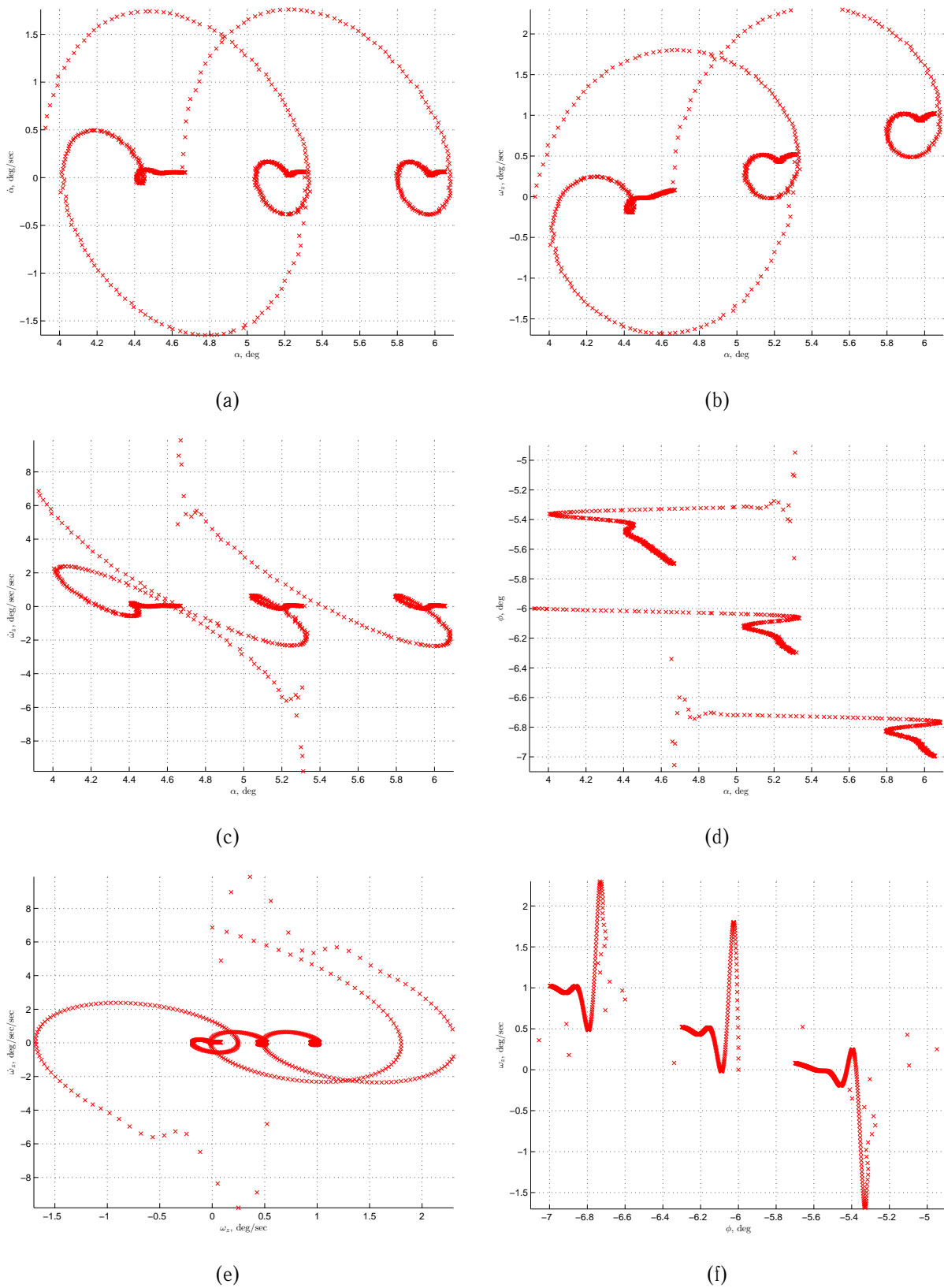


Рис. Б.28. Диаграммы информативности обучающего набора для импульсного тестового воздействия (длинный импульс, монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

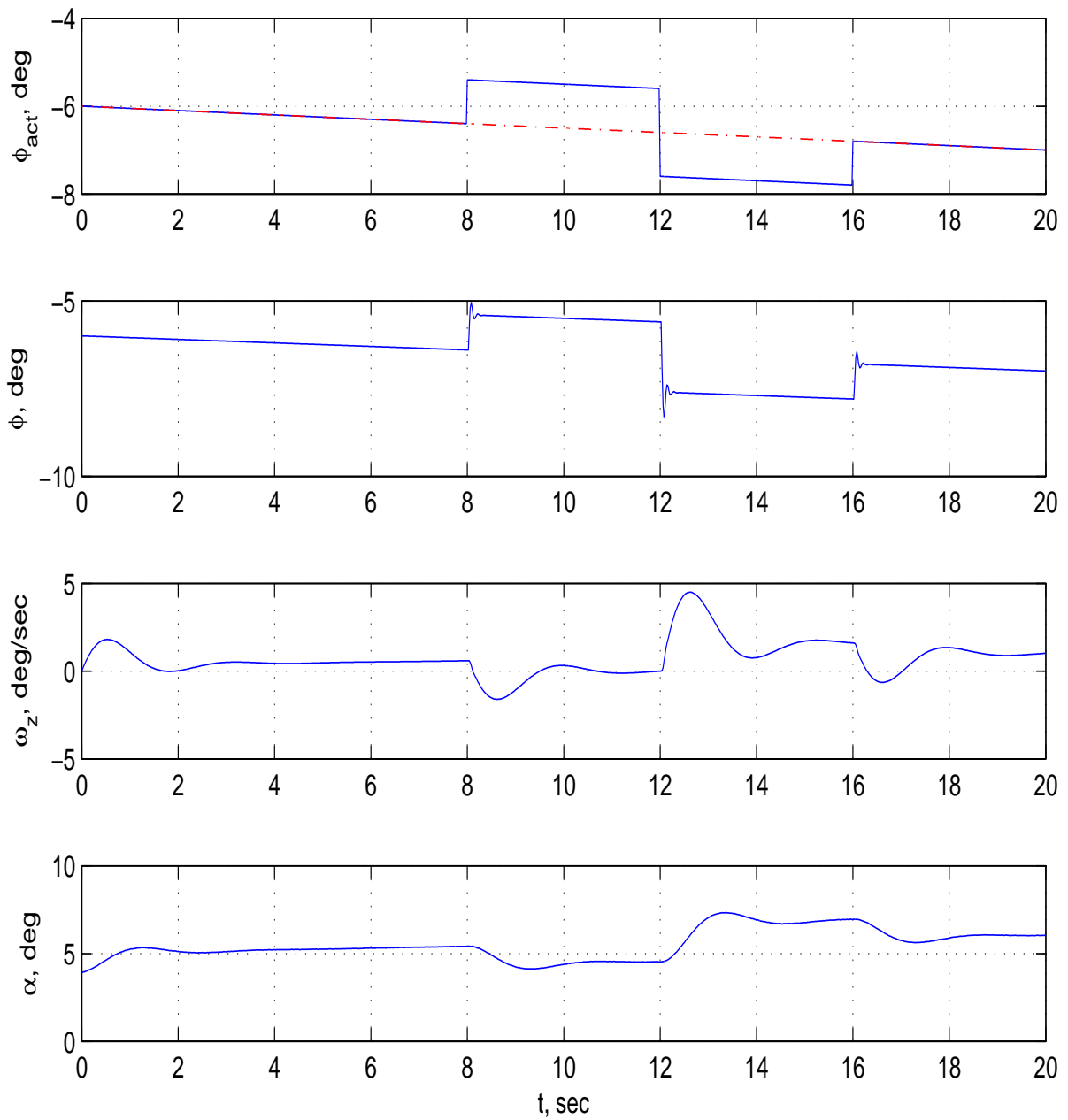
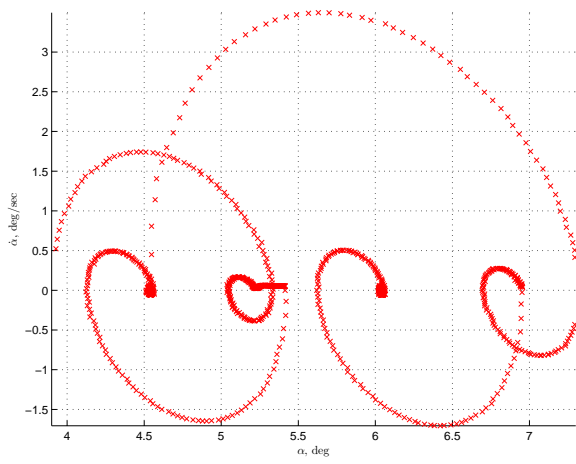
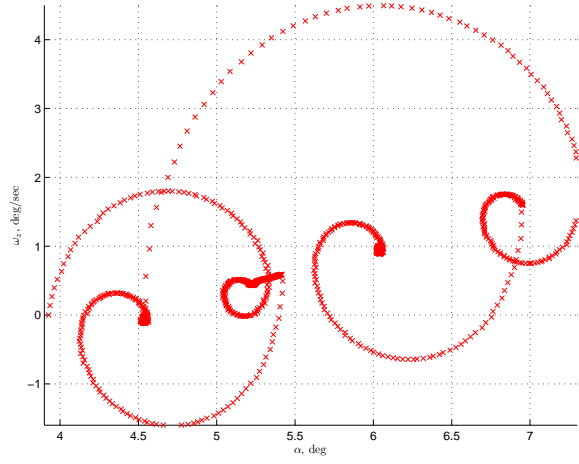


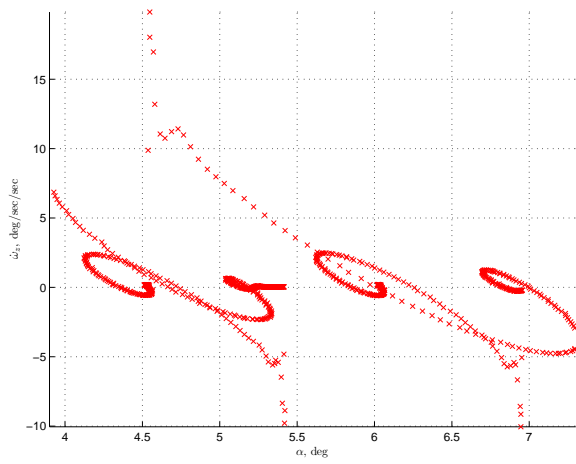
Рис. Б.29. Фазовые траектории, порождающие обучающий набор при тестовом воздействии типа дублет (монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



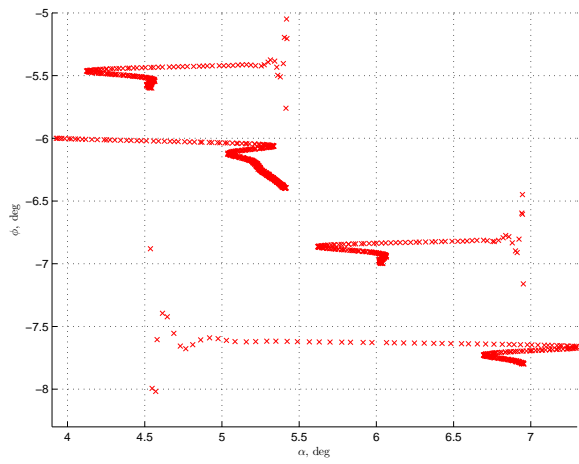
(a)



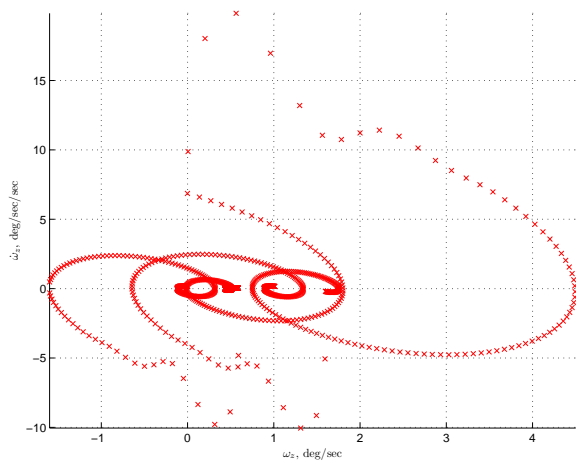
(b)



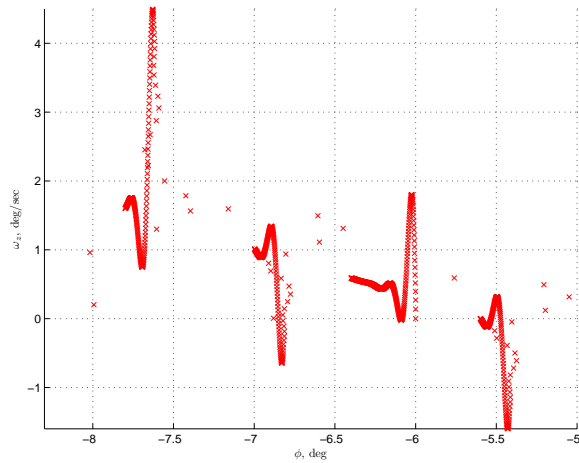
(c)



(d)



(e)



(f)

Рис. Б.30. Диаграммы информативности обучающего набора для тестового воздействия типа дублет (монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

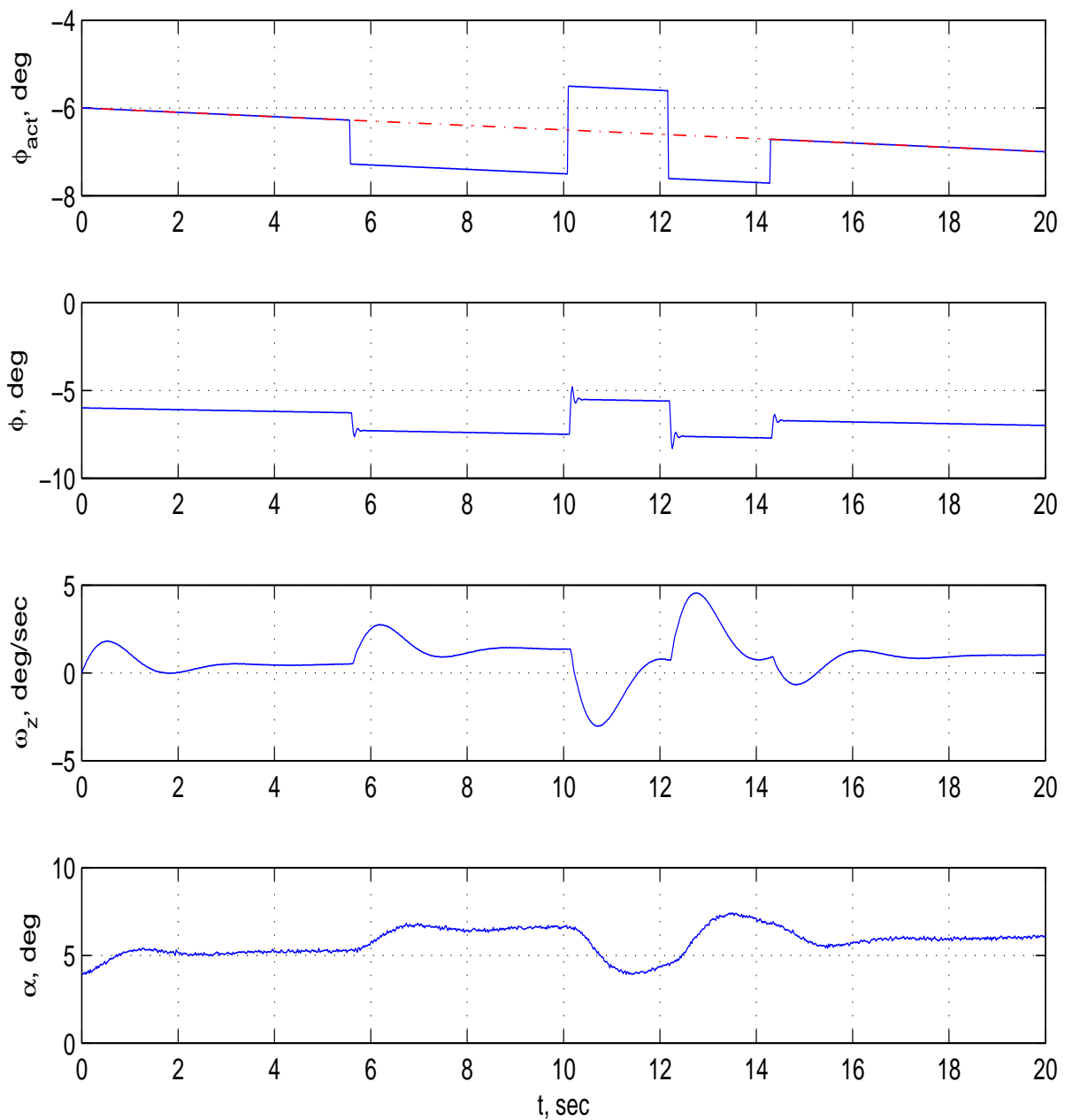
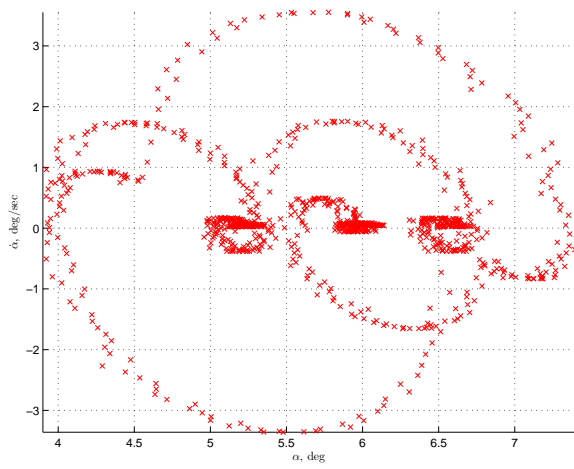
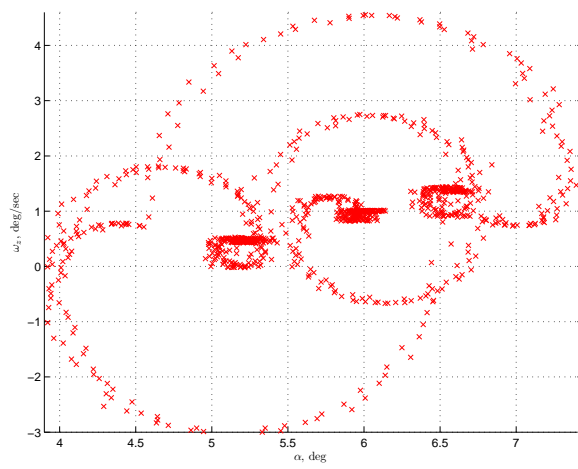


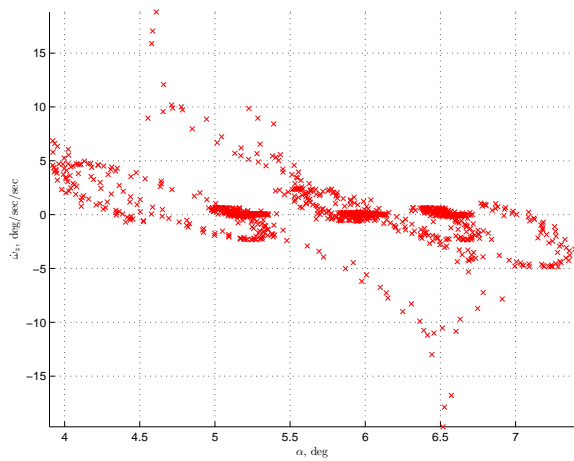
Рис. Б.31. Фазовые траектории, порождающие обучающий набор при тестовом воздействии типа 2-1-1 (монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



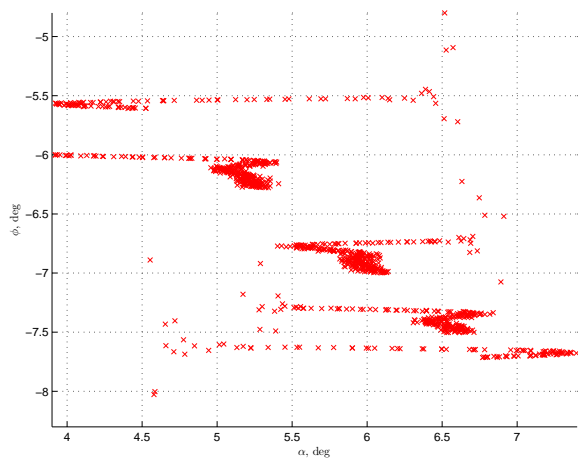
(a)



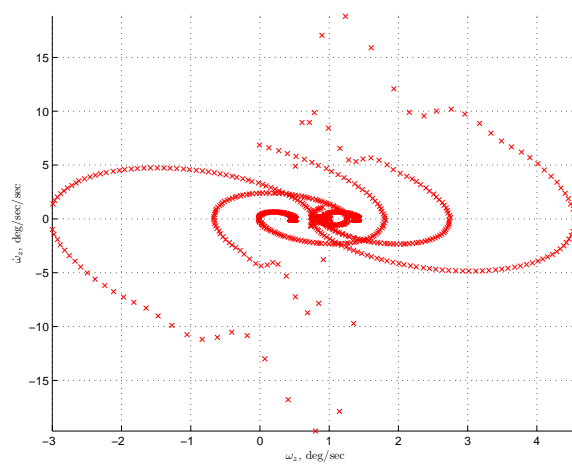
(b)



(c)



(d)



(e)

Рис. Б.32. Диаграммы информативности обучающего набора для тестового воздействия типа 2-1-1 (монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$

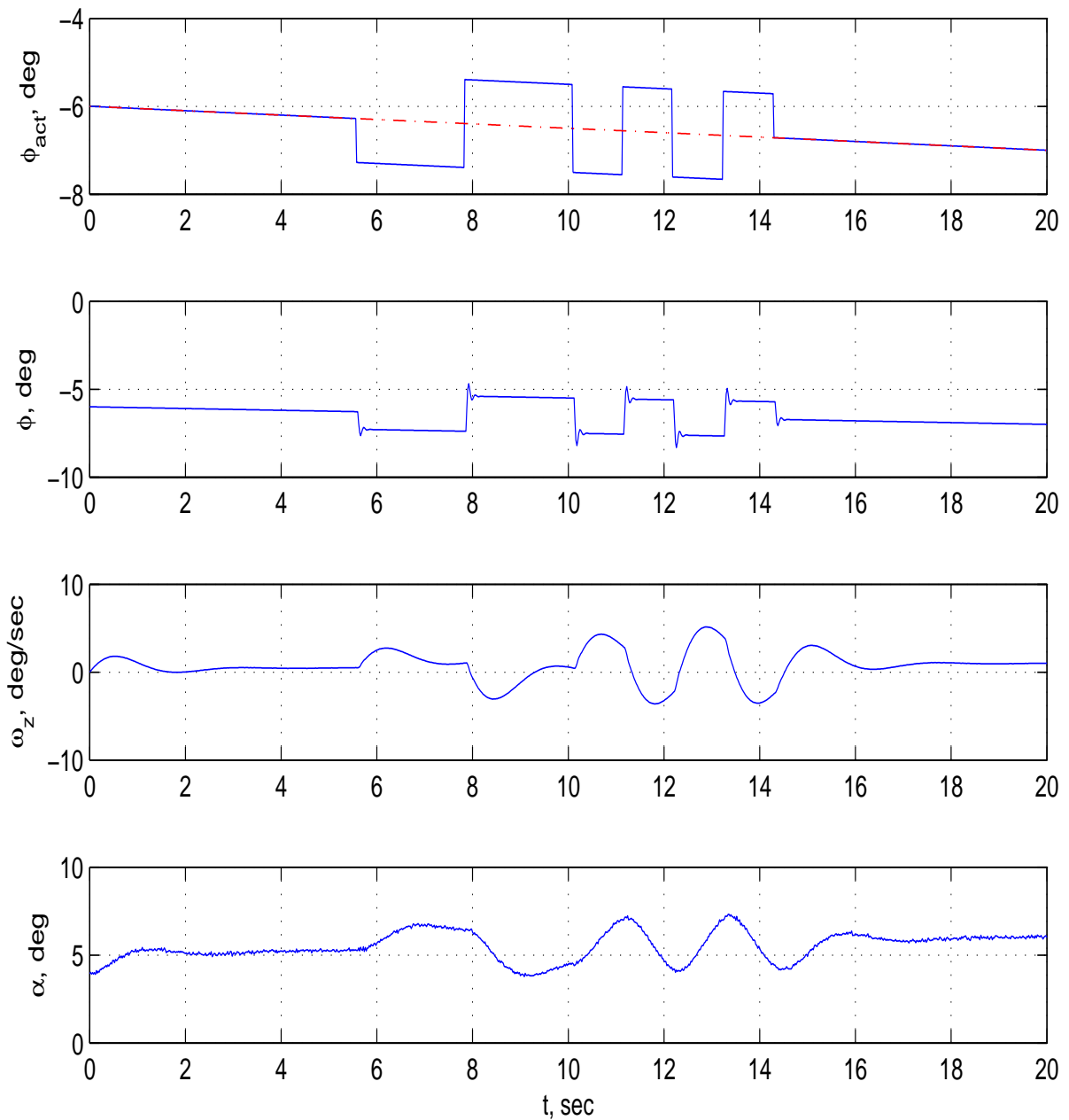
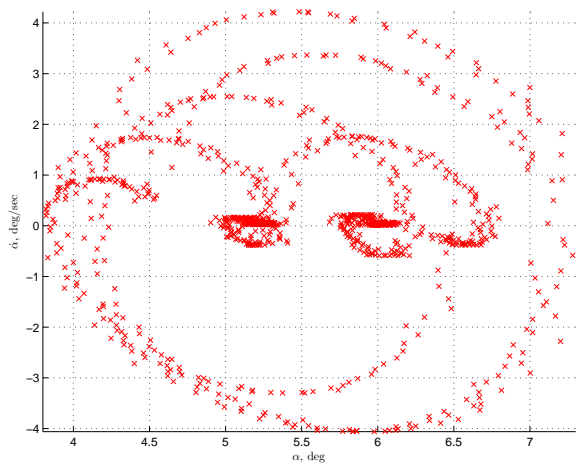
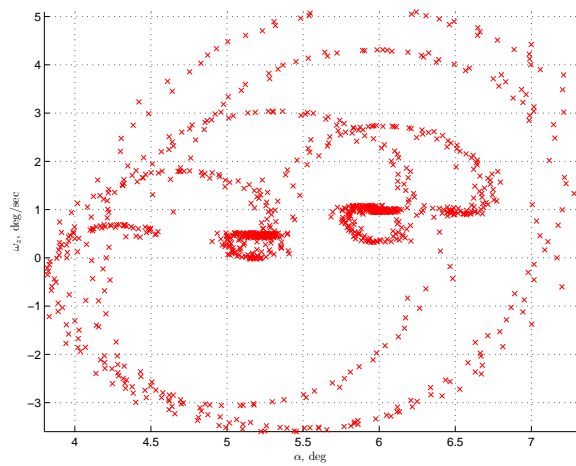


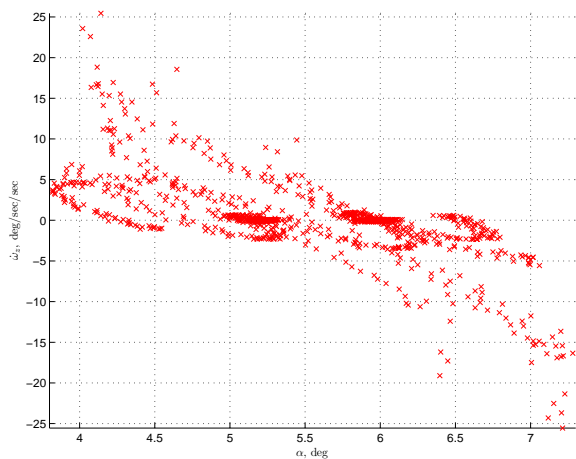
Рис. Б.33. Фазовые траектории, порождающие обучающий набор при тестовом воздействии типа 2-2-1-1-1-1 (монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



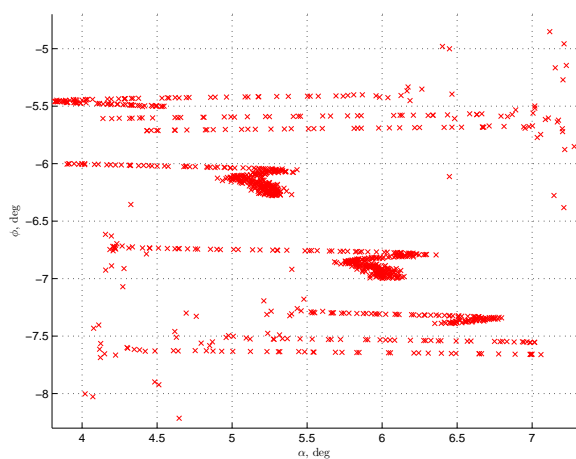
(a)



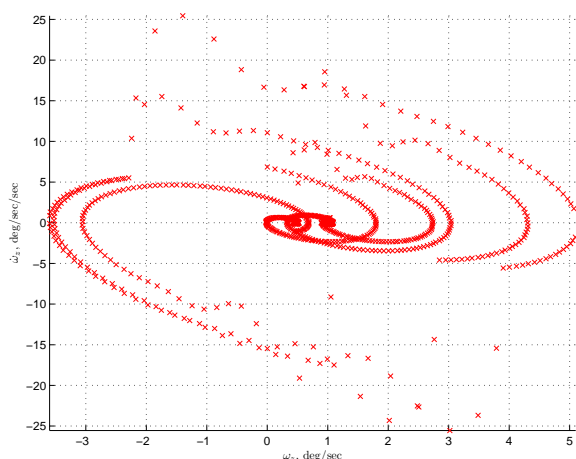
(b)



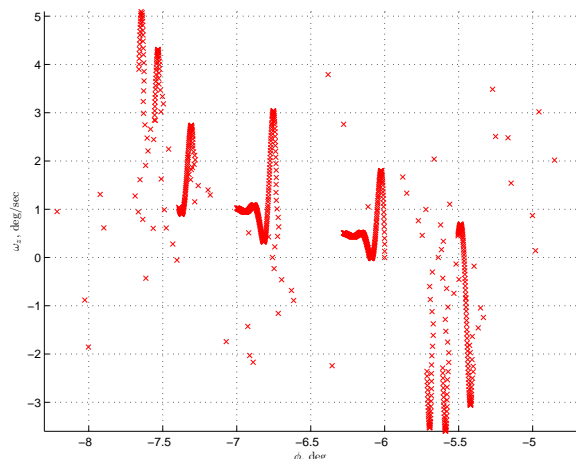
(c)



(d)



(e)



(f)

Рис. Б.34. Диаграммы информативности обучающего набора для тестового воздействия типа 2-2-1-1-1-1 (монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

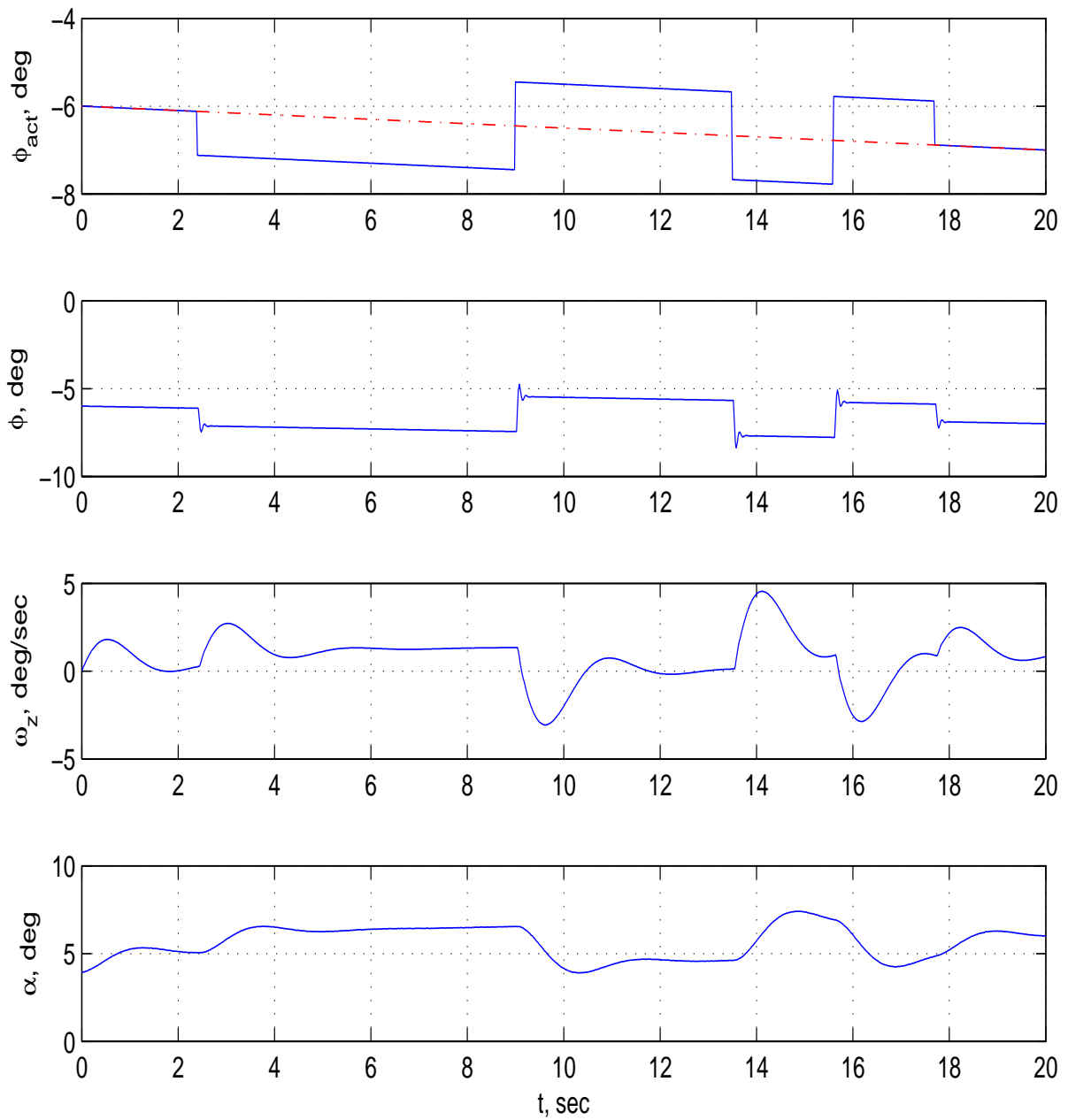
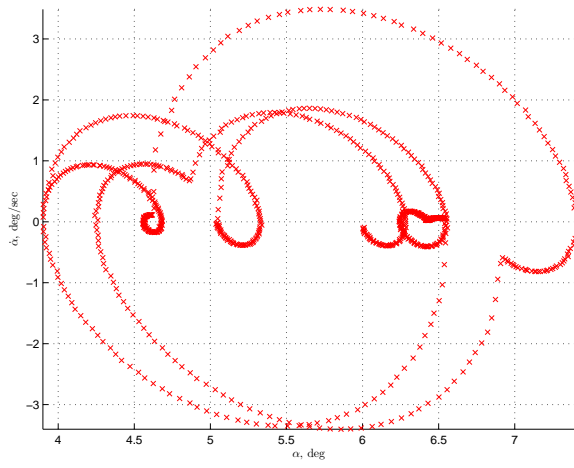
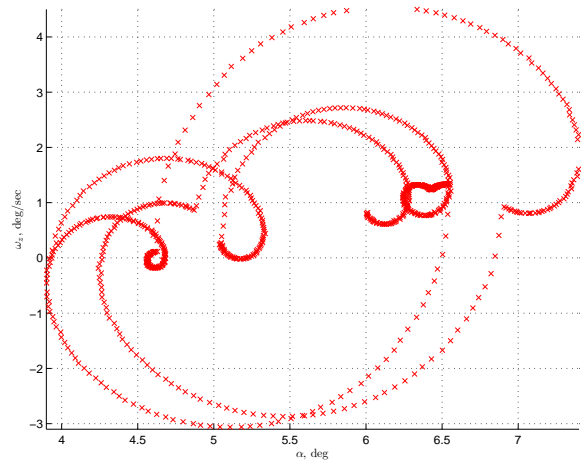


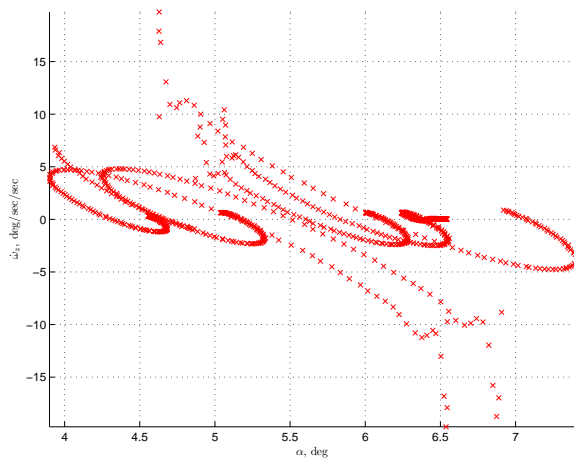
Рис. Б.35. Фазовые траектории, порождающие обучающий набор при тестовом воздействии типа 3-2-1-1 (монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



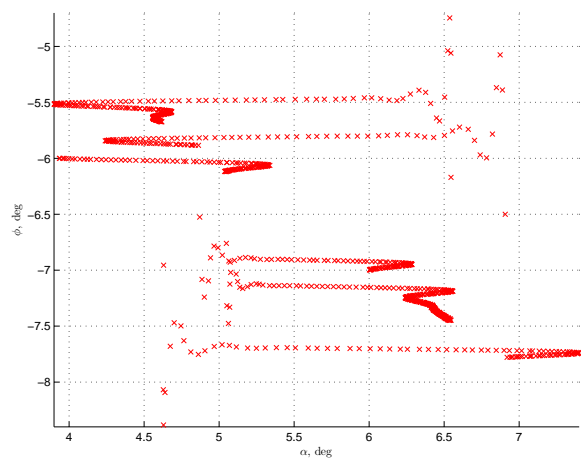
(a)



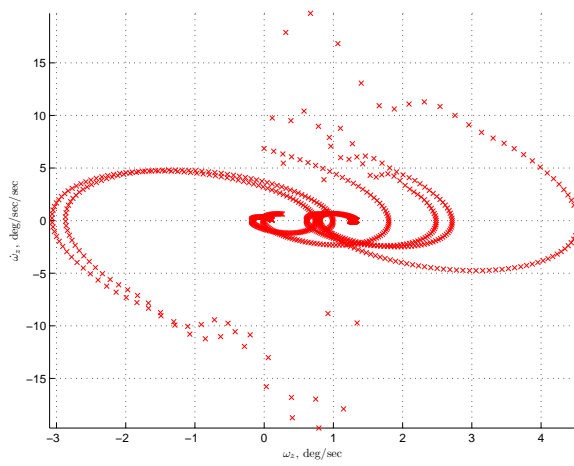
(b)



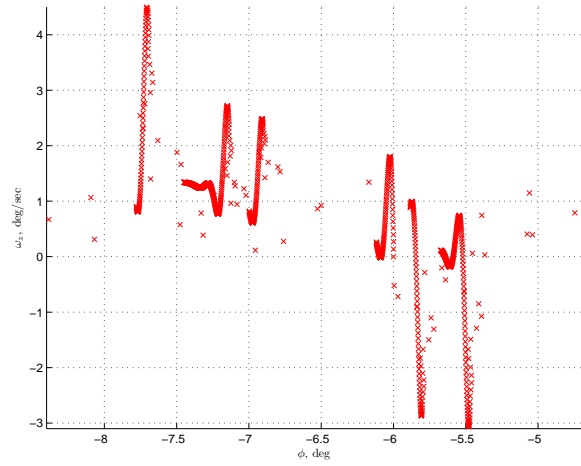
(c)



(d)



(e)



(f)

Рис. Б.36. Диаграммы информативности обучающего набора для тестового воздействия типа 3-2-1-1 (монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

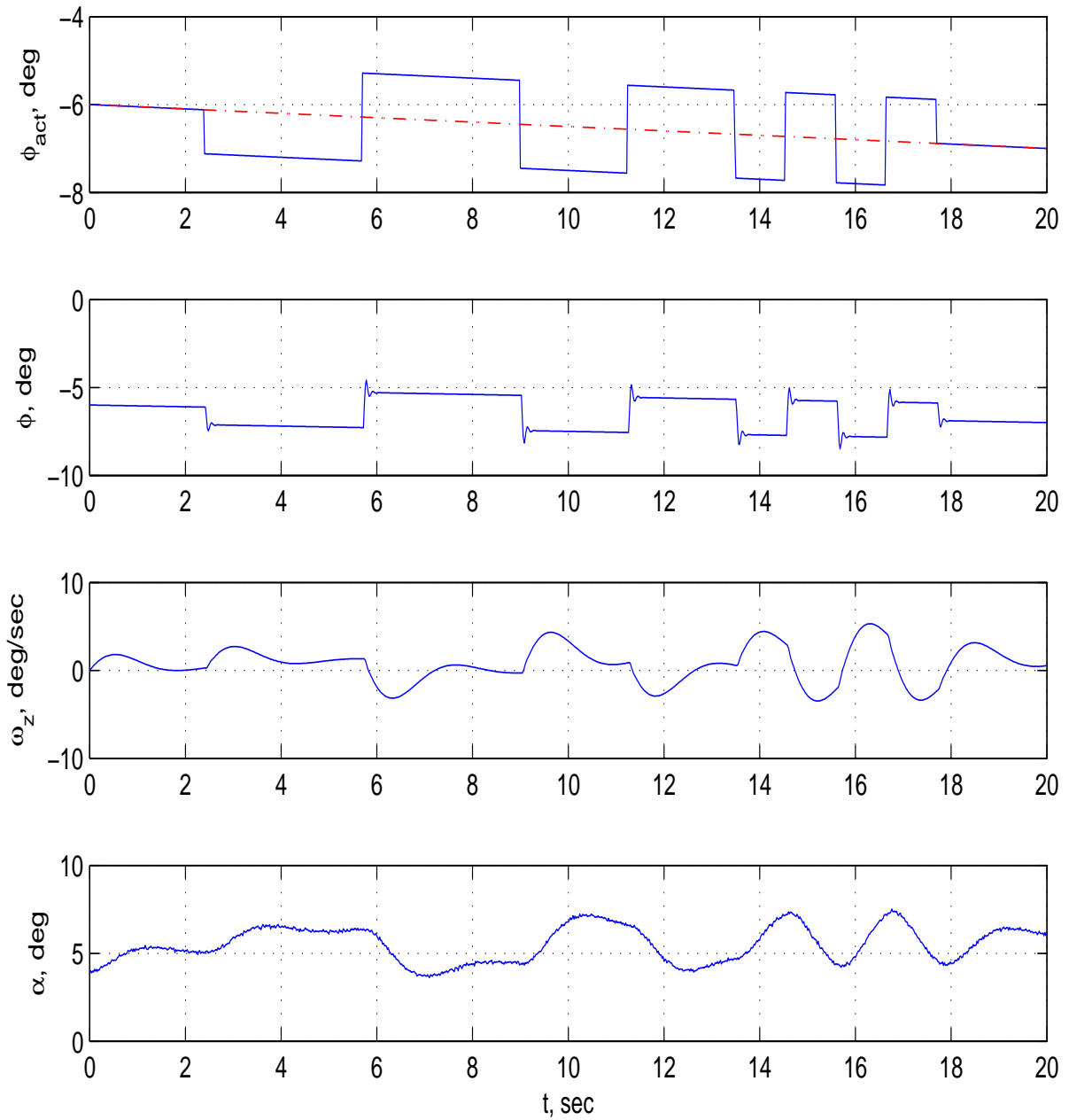


Рис. Б.37. Фазовые траектории, порождающие обучающий набор при тестовом воздействии типа 3-3-2-2-1-1-1-1 (монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО

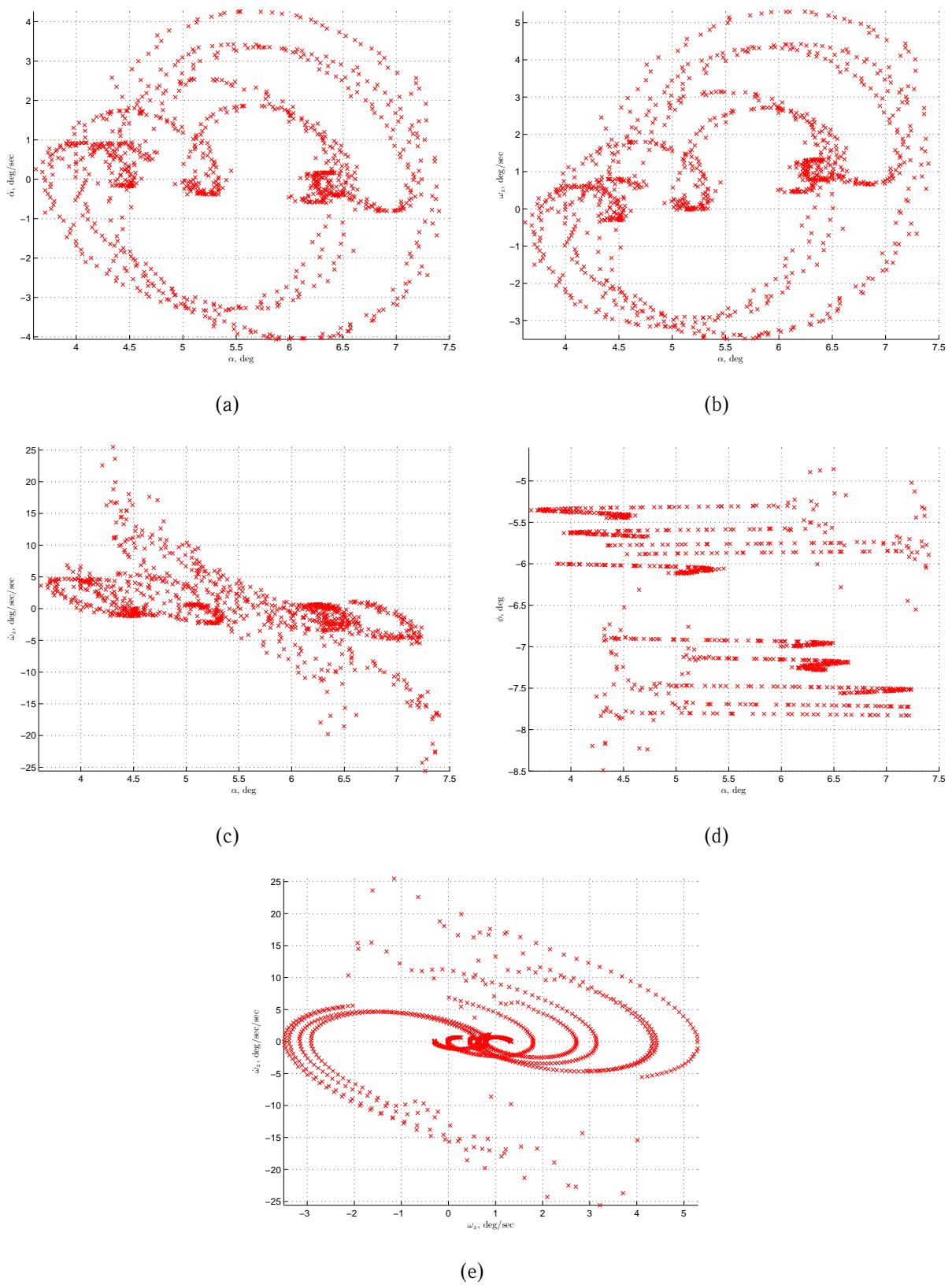


Рис. Б.38. Диаграммы информативности обучающего набора для тестового воздействия типа 3-3-2-2-1-1-1-1 (монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$

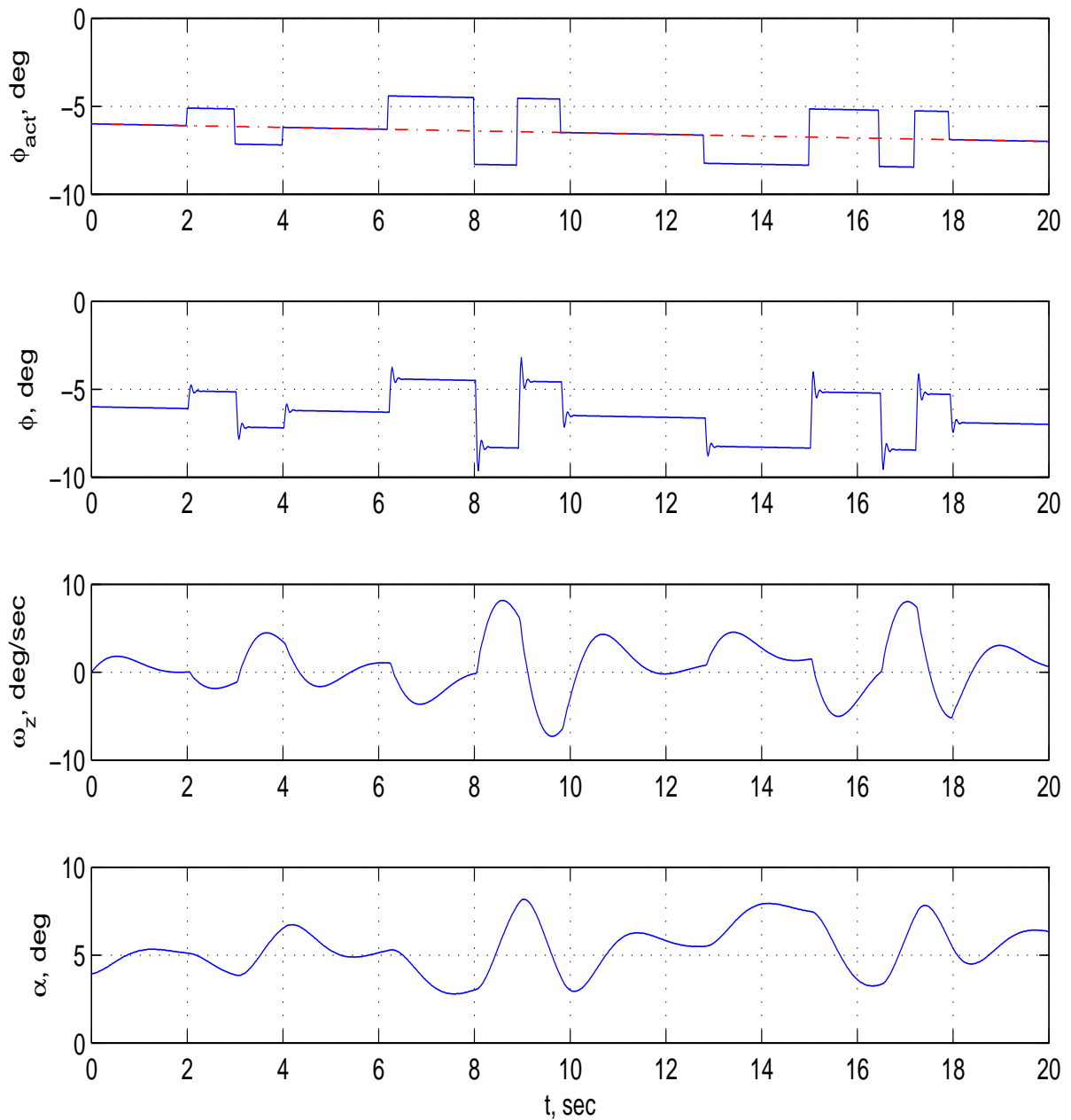
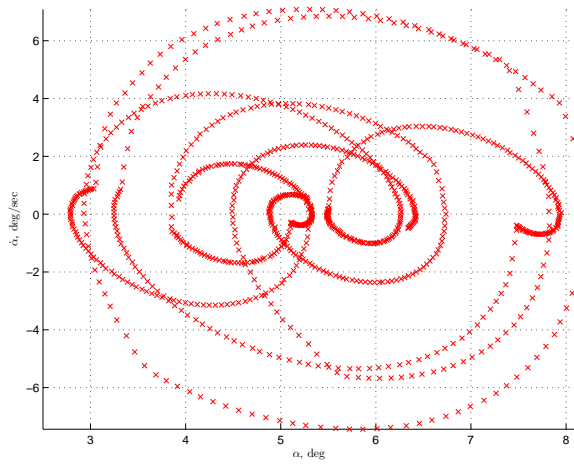
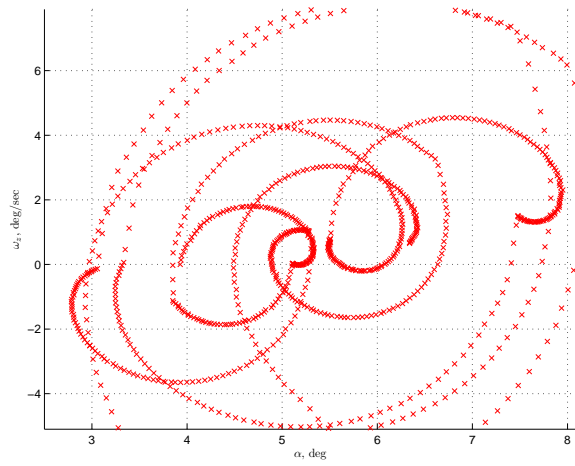


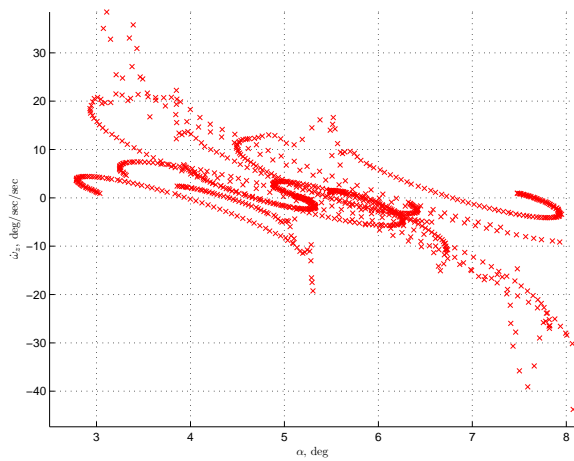
Рис. Б.39. Фазовые траектории, порождающие обучающий набор для комбинированного тестового воздействия типа дублет+(2-1-1)+(3-2-1-1) (монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



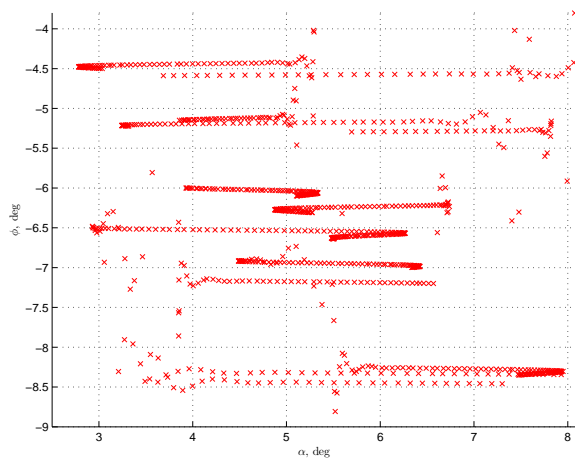
(a)



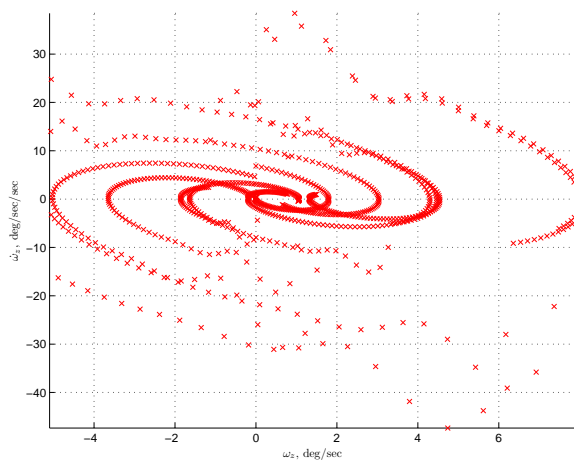
(b)



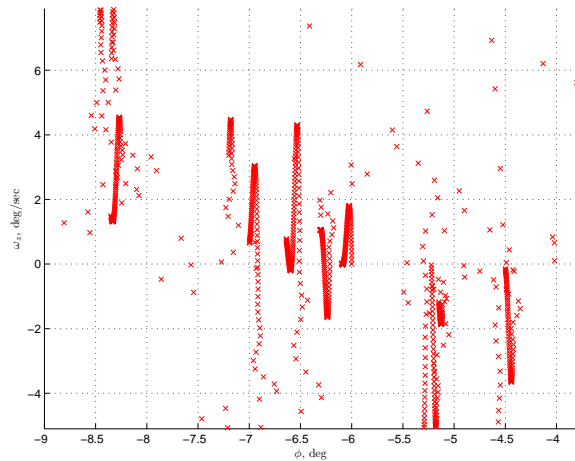
(c)



(d)



(e)



(f)

Рис. Б.40. Диаграммы информативности обучающего набора для комбинированного тестового воздействия типа дублет+(2-1-1)+(3-2-1-1) (монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — $(\alpha, \ddot{\omega}_z)$; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — $(\phi, \ddot{\omega}_z)$

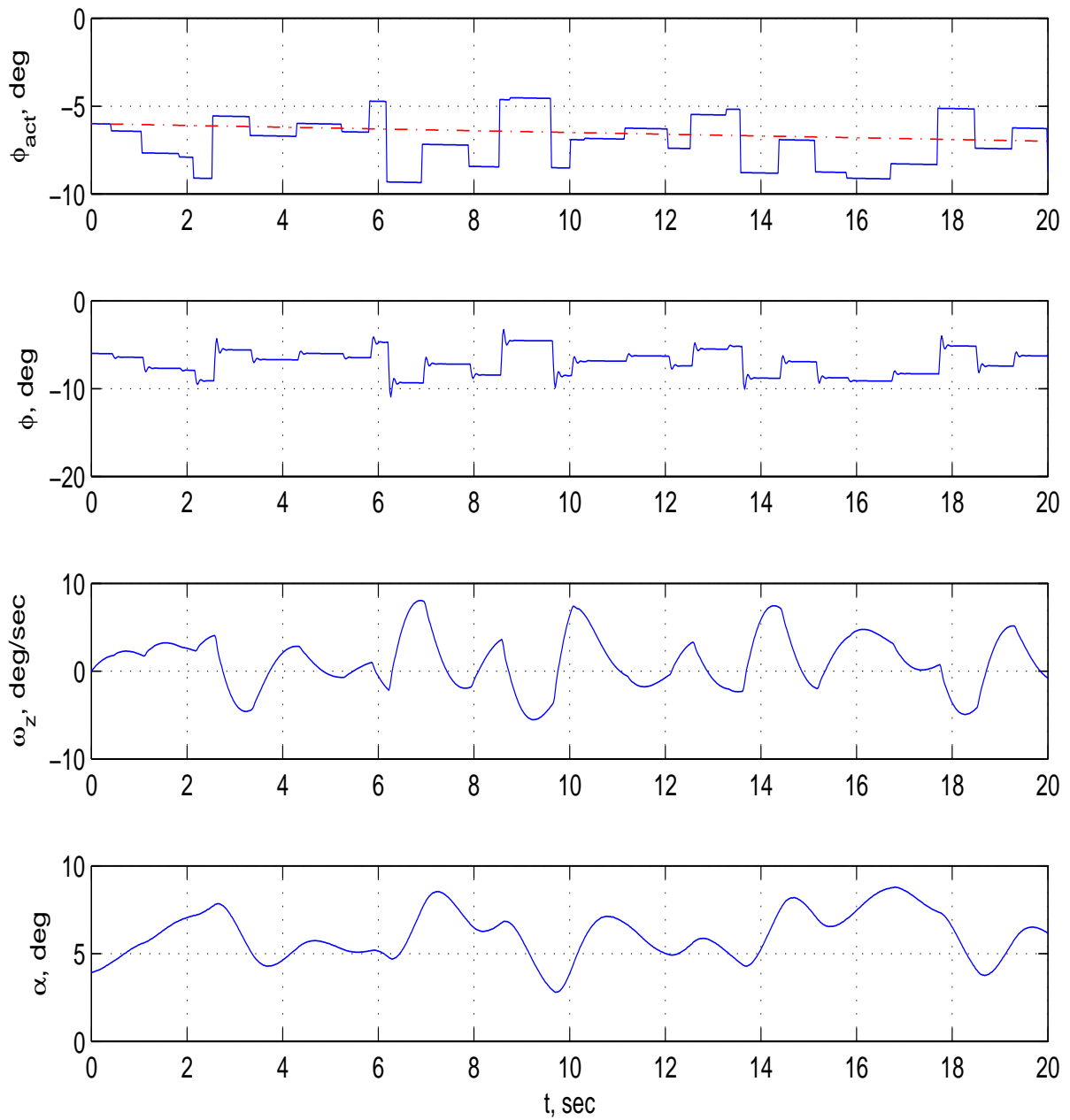
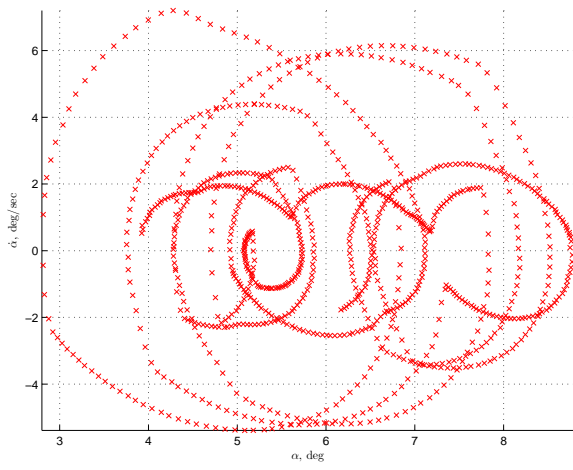
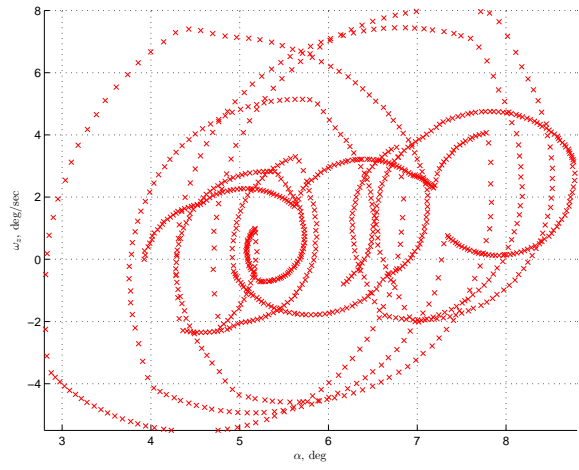


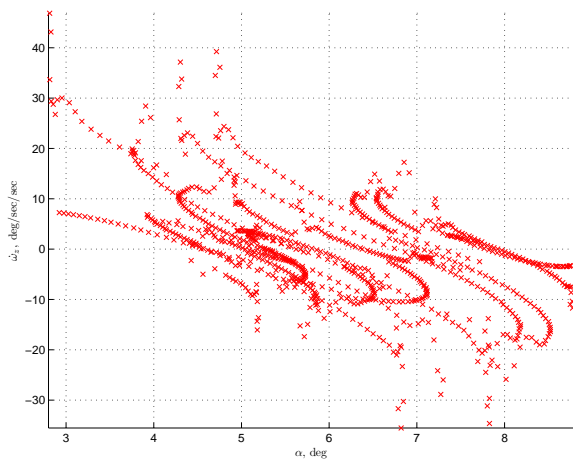
Рис. Б.41. Фазовые траектории, порождающие обучающий набор при случайном тестовом воздействии типа (монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



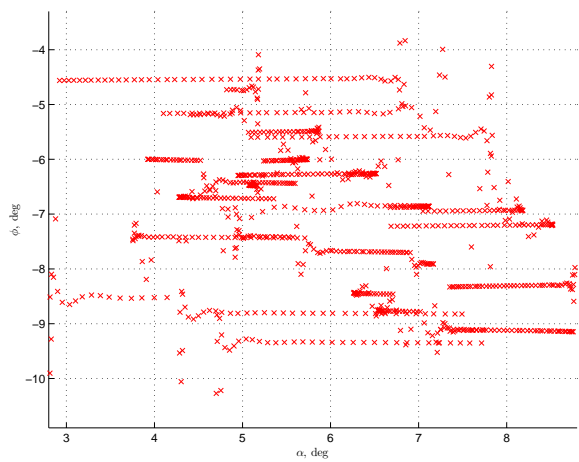
(a)



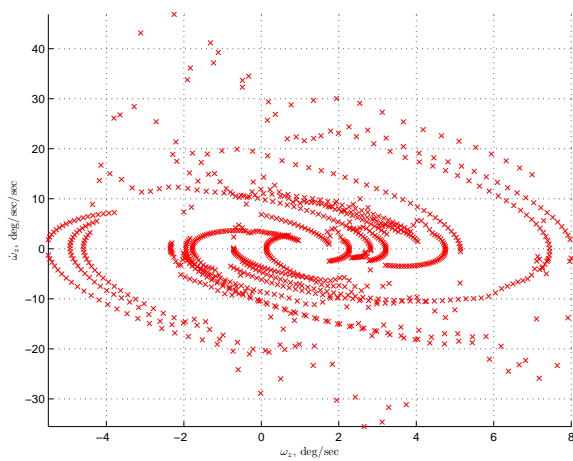
(b)



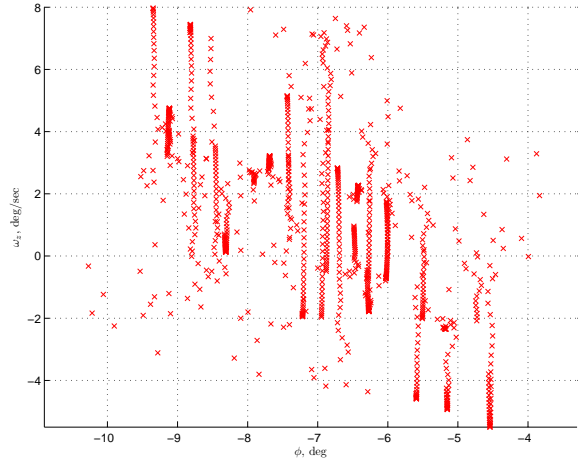
(c)



(d)



(e)



(f)

Рис. Б.42. Диаграммы информативности обучающего набора для случайного тестового воздействия (монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)

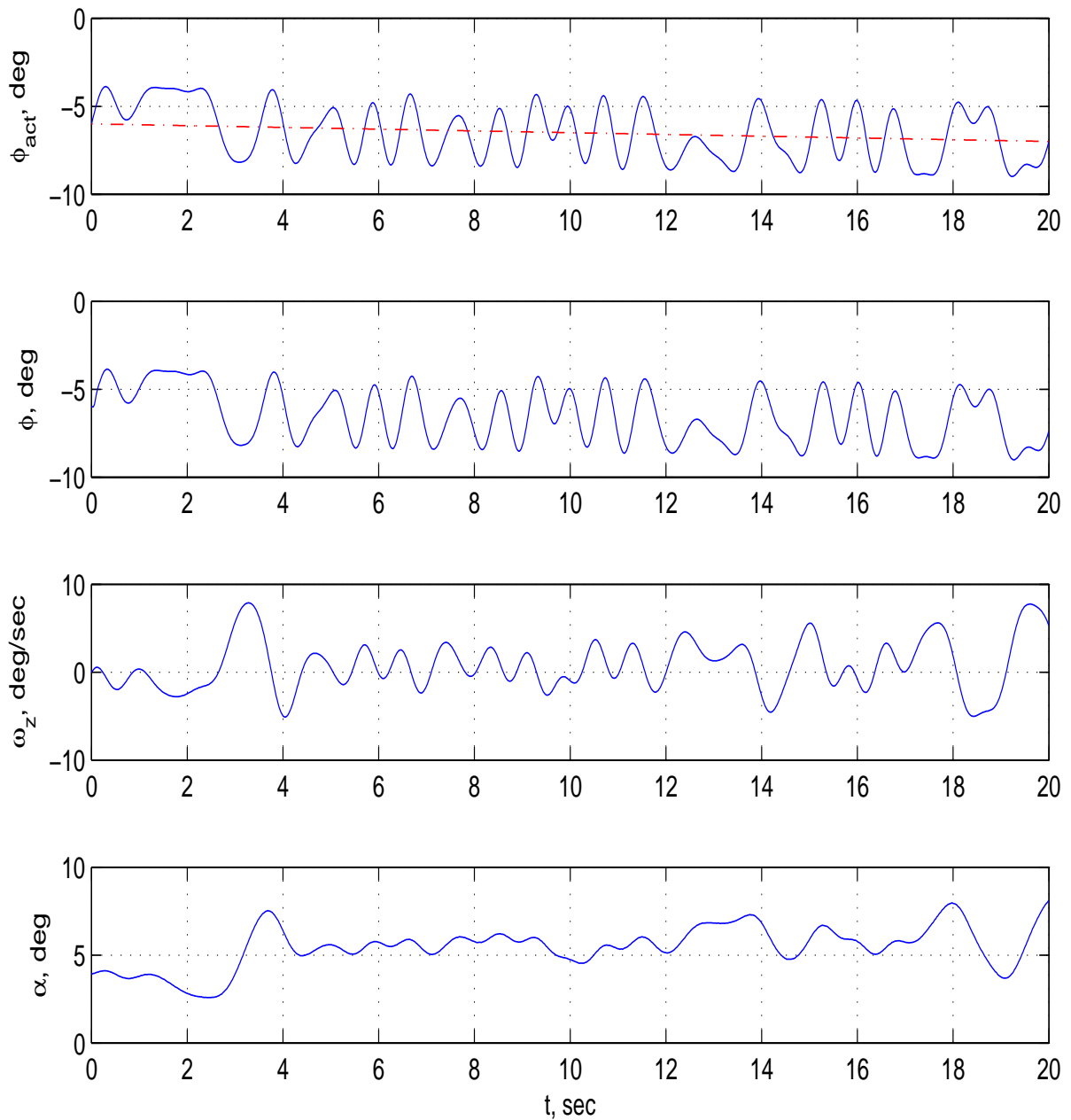
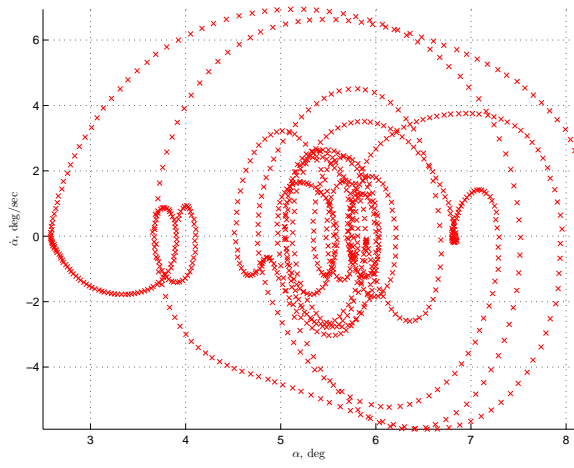
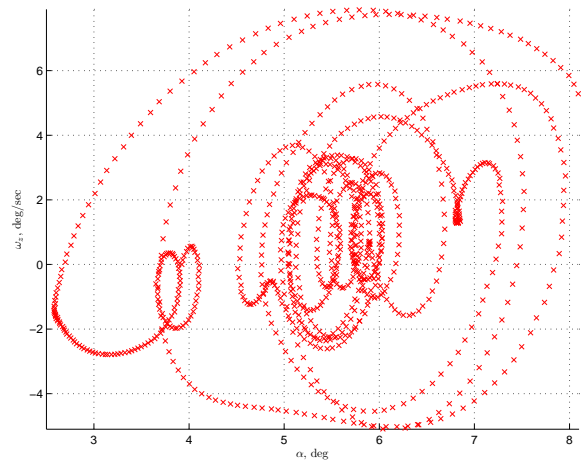


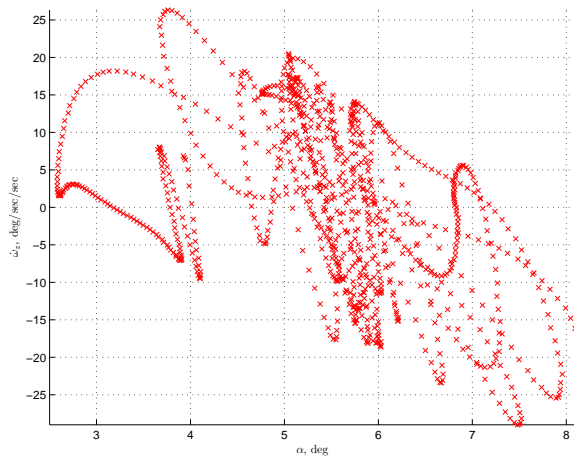
Рис. Б.43. Фазовые траектории, порождающие обучающий набор при полигармоническом тестовом воздействии типа (монотонный режим). Здесь: α — угол атаки; ω_z — угловая скорость тангажа; ϕ — угол отклонения цельноповоротного горизонтального оперения; ϕ_{act} — командный сигнал привода цельноповоротного ГО



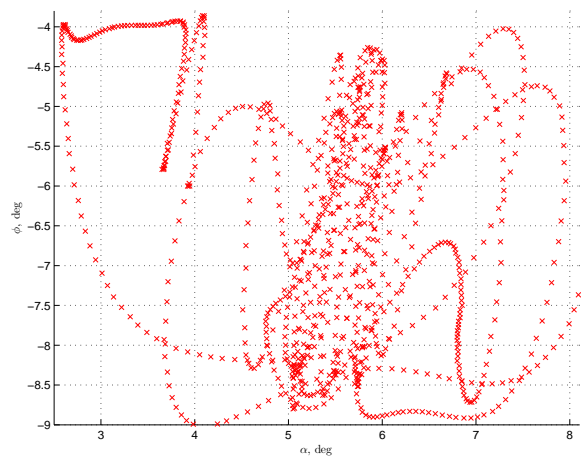
(a)



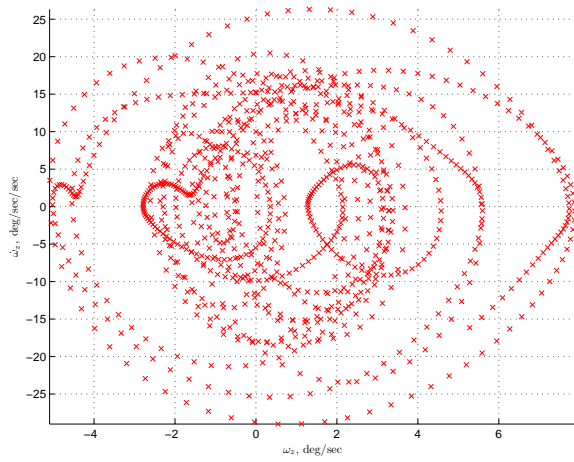
(b)



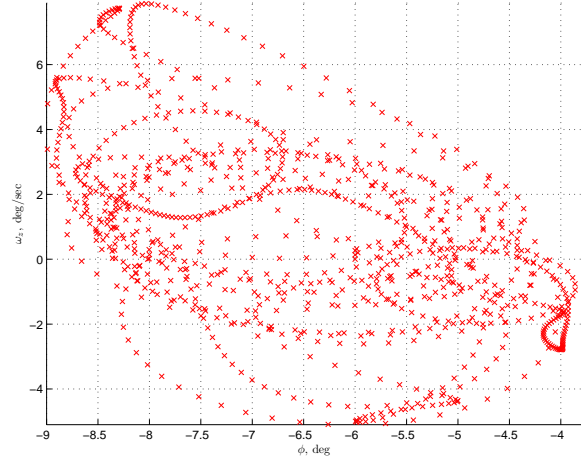
(c)



(d)



(e)



(f)

Рис. Б.44. Диаграммы информативности обучающего набора для полигармонического тестового воздействия (монотонный режим): (a) — $(\alpha, \dot{\alpha})$; (b) — (α, ω_z) ; (c) — $(\alpha, \dot{\omega}_z)$; (d) — (α, ϕ) ; (e) — $(\omega_z, \dot{\omega}_z)$; (f) — (ϕ, ω_z)