Труды МАИ. 2025. № 145.

Trudy MAI. 2025. No. 145. (In Russ.)

Научная статья
УДК 517.977.5
URL: https://trudymai.ru/published.php?ID=186897
EDN: https://www.elibrary.ru/AFJZPG

ОПТИМИЗАЦИЯ АЛГОРИТМА ПРЕДОТВРАЩЕНИЯ СТОЛКНОВЕНИЙ В ВОЗДУХЕ НА ОСНОВЕ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ С РЕСУРСНЫМИ ОГРАНИЧЕНИЯМИ

Е.С. НЕРЕТИН[1]✉, Ц. ЛЮ[2]

[1]Филиал ПАО "Яковлев" - Центр комплексирования, Москва, Россия [2]Северо-Западныи политехническии университет, Сиань, Китаи

✉evgeny.neretin@ic.yakovlev.ru

Цитирование: Неретин Е.С., Лю Ц. Оптимизация алгоритма предотвращения столкновении в воздухе на основе обучения с подкреплением с ресурсными ограничениями // Труды МАИ. 2025. № 145. URL: https://trudymai.ru/published.php?ID=186897

Аннотация. С увеличением плотности воздушного трафика возрастает необходимость в эффективных системах предотвращения столкновений в воздухе. Традиционные системы, такие как TCAS, хотя и эффективно поддерживают безопасность, сталкиваются с трудностями в адаптации и оптимизации в современных сложных условиях. Чтобы преодолеть эти ограничения, мы применяем обучение с подкреплением (RL) в рамках марковского процесса принятия решений с ограничениями по ресурсам (RC-MDP), вводя управление виртуальными ресурсами для сокращения числа ложных тревог. Мы предлагаем бонус за время и ресурсы (TRB) для модификации алгоритмов DQN и SAC в DQNTRB и SACTRB, которые поощряют эффективное использование ресурсов при сохранении эффективности предотвращения столкновений. Результаты экспериментов показывают, что эти

модифицированные алгоритмы значительно сокращают количество ложных тревог, достигая почти аналогичной эффективности по сравнению с алгоритмами без ограничений.

Ключевые слова: реакция пилота, глубокое обучение с подкреплением, воздушное столкновение, марковский процесс принятия решений, динамическое программирование

OPTIMIZATION OF AIRBORNE COLLISION AVOIDANCE ALGORITHM BASED ON RESOURCE-CONSTRAINED REINFORCEMENT LEARNING

E.S. NERETIN[1]✉, **L. ZUOCHENG**[2]

[1]Branch of PJSC Yakovlev – integration center, Moscow, Russia

[2]Northwestern Polytechnical University, Xi'An, P. R. China

✉evgeny.neretin@ic.yakovlev.ru

Citation: Neretin E.S., Zuocheng L. Optimization of airborne collision avoidance algorithm based on resource-constrained reinforcement learning // Trudy MAI. 2025. No. 145. (In Russ.). URL: https://trudymai.ru/published.php?ID=186897

Abstract. As air traffic density continues to rise with the advancement of aviation technology, the demand for efficient and reliable airborne collision avoidance systems becomes increasingly urgent. Traditional systems, such as the Traffic Collision Avoidance System (TCAS), mainly rely on heuristic rules and parameter settings, which, although effective in maintaining safety, struggle to adapt and optimize under the complexities of modern aviation environments. To address these limitations, we explore the application of reinforcement learning (RL) to optimize the performance of collision avoidance systems. We define the problem within a resource-constrained Markov decision process (RC-MDP) framework, incorporating virtual resource management to control the

frequency of nuisance alerts, which are frequent alarms that do not require actual evasive action. We propose a novel time-resource bonus (TRB) mechanism to modify and enhance two standard RL algorithms, DQN and SAC, into DQNTRB and SACTRB. This approach encourages resource-efficient actions while maintaining collision avoidance performance. Our experimental results demonstrate that these modified algorithms significantly reduce nuisance alerts while achieving near-equivalent collision avoidance performance compared to algorithms without resource constraints.

Keywords: pilot respond, deep reinforcement learning, airborne collision avoidance, markov decision-making process, dynamic programming.

Введение

С быстрым развитием авиационных технологии плотность воздушного трафика неуклонно увеличивается, что ведет к необходимости более эффективных и надежных систем предотвращения столкновении в воздухе. Традиционные системы, такие как система предотвращения столкновении в воздухе (TCAS), в основном полагаются на эвристические правила и настрои ки параметров, которые, хотя и эффективны для обеспечения безопасности, сталкиваются с проблемами при оптимизации и развитии в условиях все более сложных авиационных сред [1, 2]. Недавние исследования сосредоточены на повышении автоматизации и надежности алгоритмов предотвращения столкновении [3, 4, 5], улучшении использования аппаратных средств в авиационных системах [6, 7], решении проблем предотвращения столкновении между несколькими воздушными судами [8], а также использовании технологии автоматического зависимого наблюдения – вещания (ADS-B) для повышения эффективности предотвращения столкновении [9].

Обучение с подкреплением (RL), метод безнадзорного обучения, широко применяется для решения различных задач по уклонению от препятствии и принятия решении [8, 10, 11]. Основная идея RL заключается в постоянноі оптимизации политик путем взаимодеи ствия с окружающеи средои для

максимизации накопленных вознаграждении . При применении традиционных алгоритмов RL к задаче предотвращения столкновении воздушного движения мы обнаружили, что, хотя RL эффективно решает задачи предотвращения столкновении и планирования трафика, его производительность в уменьшении количества ложных тревог оказалась неудовлетворительнои [12]. Для решения этои проблемы мы ввели концепцию виртуальных ресурсов — когда каждое предупреждение о столкновении потребляет определенное количество виртуальных ресурсов, и система запрещает выдачу дальнеи ших предупреждении при их исчерпании. Эта задача может рассматриваться как проблема RL с ограниченными ресурсами, где ключом к успешному принятию решении является эффективное управление ресурсами при выполнении основнои задачи [13, 14].

В даннои статье сначала излагаются основные принципы алгоритмов RL, которые мы применили к задаче предотвращения столкновении , включая алгоритмы Deep Q-Network (DQN) и Soft Actor-Critic (SAC). Затем мы устанавливаем основную структуру задачи конфликта между двумя воздушными судами, охватывая пространства состоянии и деи ствии , функции вознаграждения для основнои и вспомогательнои задач, а также показатели оценки производительности алгоритмов. Мы анализируем различия в производительности этих двух алгоритмов при решении задач предотвращения столкновении как с ресурсными ограничениями, так и без них. Мы обнаружили, что в условиях ограничении по ресурсам оба алгоритма демонстрируют слабые результаты и низкую эффективность использования образцов. Дополнительныи экспериментальныи анализ показал, что исследование агентом среды часто ограничивается количеством ресурсов, и доступные деи ствия зависят от оставшихся ресурсов. Оба алгоритма, DQN и SAC, склонны быстро расходовать ресурсы, ограничивая последующую разведку.

Для решения этои проблемы мы расширили марковскии процесс принятия решении (MDP) до MDP с ограниченными ресурсами, введя управление ресурсами [14, 15], и предложили новыи механизм бонуса за время и ресурсы (TRB). Мы

модифицировали стандартные алгоритмы SAC и DQN в SACTRB и DQNTRB соответственно. Эти алгоритмы побуждают агента экономить ресурсы, выполняя задачи по предотвращению столкновении и сокращая количество ложных тревог, тем самым исследуя состояния с более широким набором возможных опции . Наконец, мы провели эксперименты для оценки производительности оптимизированных алгоритмов. Результаты показывают, что в условиях ограниченных ресурсов оптимизированные алгоритмы значительно превосходят традиционные алгоритмы как в эффективности предотвращения столкновении , так и в снижении количества ложных тревог. По сравнению с алгоритмами без ограничении по ресурсам оптимизированные алгоритмы достигли почти аналогичнои производительности в планировании столкновении , значительно сократив частоту ложных тревог.

Предварительные сведения

В этом разделе мы представляем модельныи каркас и основные алгоритмы, используемые в нашем исследовании.

1） Марковский процесс принятия решений

Марковскии процесс принятия решении — это математическая структура, используемая для моделирования последовательных задач принятия решении , которая определяется кортежем (S,A,p,r,γ), где $s_t \in S$ — состояние в момент времени t, $a_t \in A$ — деи ствие, совершаемое агентом в момент времени t в результате процесса принятия решении , $r_t = R(s_t, a_t, s_{t+1})$— вознаграждение, получаемое агентом в результате выполнения деи ствия a_t из состояния s_t и перехода в состояние s_{t+1}, а $p(s_t, a, s_{t+1})$ — функция переходов, которая отображает вероятность $p\ s\square_{t\square 1} | s\ a_t, {}_t\square$ перехода в состояние s_{t+1} при выполнении деи ствия a_t из состояния s_t. $\gamma \in [0,1]$ — это коэффициент дисконтирования, используемыи для взвешивания мгновенных вознаграждении по отношению к будущим. Цель MDP заключается в нахождении оптимальнои стратеги $\square\square a\ s\square$, которая максимизирует ожидаемую накопленную дисконтированную награду:

$$V(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \mid s_0 = s\right] \quad (1)$$

2) Глубокая Q-сеть (DQN)

Глубокая Q-сеть (DQN) [16, 17] — это алгоритм обучения с подкреплением, основанный на Q-обучении, который использует нейронные сети для аппроксимации функции Q-значений $Q(s,a)$, решая проблему многомерных пространств состояний. Функция Q-значений представляет ожидаемую накопленную награду за выполнение действия a в состоянии s. DQN обновляет Qзначения по следующей формуле:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r + \gamma \max_{a} Q(s_{t+1}, a) - Q(s_t, a_t) \right] \quad (2)$$

Где α—это скорость обучения, γ—это коэффициент дисконтирования, r—это немедленное вознаграждение, а s'—это следующее состояние после выполнения действия a. DQN также использует повторное проигрывание опыта для уменьшения корреляции между выборками и целевую сеть для улучшения стабильности обучения.

3) Мягкий актор-критик (SAC)

Мягкий актор-критик (SAC) [18] — это алгоритм обучения с подкреплением (off-policy), который объединяет методы градиента политики с регуляризацией энтропии для балансировки исследования и эксплуатации. Цель SAC — максимизировать как ожидаемое вознаграждение, так и энтропию политики, поощряя более исследовательское поведение. Оптимизационная цель:

$$J(\pi) = \mathbb{E}_{(s_t, a_t) \sim \rho_\pi}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) + \alpha H(\pi(\cdot \mid s_t))\right] \quad (3)$$

Где α — это параметр температуры, а $H(\pi(\cdot|s_t))$ — это энтропия политики в состоянии s_t. SAC особенно подходит для пространств с непрерывными деи ствиями, улучшая исследование с помощью регуляризации энтропии. В данном исследовании, для применения алгоритма SAC к задаче предотвращения столкновении , мы внесли соответствующие корректировки как в пространство состоянии , так и в пространство деи ствии , чтобы лучше соответствовать операционным требованиям задачи.

Постановка задачи

Мы формулируем задачу предотвращения столкновений в воздухе между двумя самолетами как MDP и решаем ее с помощью алгоритмов обучения с подкреплением. В этом разделе мы представляем динамическую модель самолетов в конфликтной ситуации и даем подробное описание пространств состояний и действий. Кроме того, мы строим функцию вознаграждения, основанную на целях предотвращения столкновений и уменьшения количества ложных тревог. Также мы устанавливаем метрики оценки для анализа эффективности алгоритмов.

1) Определение пространства состоянии и деи ствии

Для поддержания согласованности с TCAS, в даннои статье акцент делается на вертикальном избегании столкновении [4]. Пространство состоянии для задачи предотвращения столкновении состоит из пяти переменных. Таблица 1 описывает эти переменные и их диапазоны, а на Рисунке 1 представлено их визуальное отображение. Первые три переменные описывают относительные положения и вертикальные скорости собственного и вторгающегося самолета. Четвертая переменная, τ, обобщает горизонтальную геометрию, указывая время до того, как горизонтальное расстояние между двумя самолетами станет менее 500 футов, и это также определяется временем до конфликта (Time to Conflict, TTC). Включение предыдущего совета в пространство состоянии позволяет нам штрафовать за изменения или усиления советов, сохраняя марковское свои ство.

Таблица 1

Переменные пространства состояний

Переменная	Описание	Значения	Единицы измерения
h	Относительная высота нарушителя	[-2500, 2500]	ft
$\dot{h}_o$	Вертикальная скорость собственного самолета	[-70, 70]	ft/s
$\dot{h}_i$	Вертикальная скорость нарушителя	[-50, 50]	ft/s
τ	Время до потери горизонтального разделения (TTC)	[0, 40]	s
a_{prev}	Предыдущий совет	См. Таблица 2	-

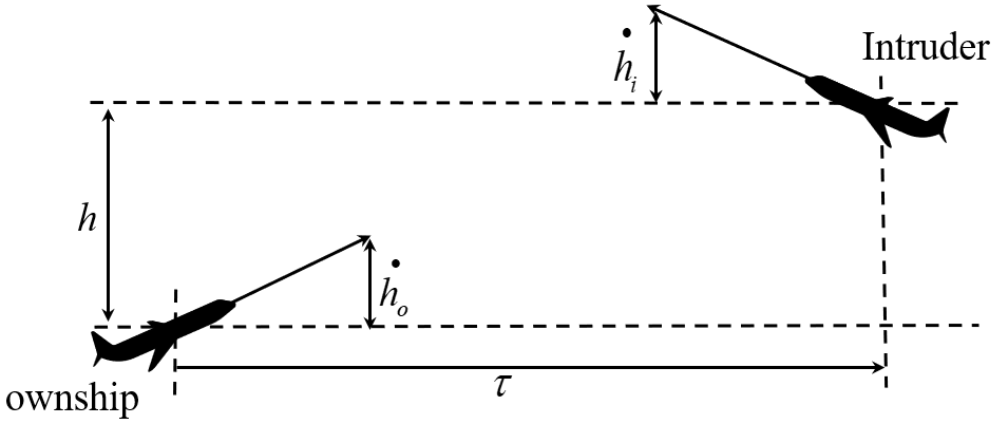


Рисунок 1 - Визуальное представление переменных состояния.

Пространство деи ствии включает советы, которые система предотвращения столкновении может предоставить во время полета, всего 7 возможных советов, как показано в Таблице 2. Все советы, кроме COC, вызывают предупреждение и направляют самолет в определенныи диапазон вертикальных

скоростеи с соответствующим ускорением. Совет COC указывает на отсутствие немедленноu угрозы столкновения с вторгающимся самолетом.

Таблица 2

Набор советов

Действие	**Описание**	**Ускорение**
COC	Нет конфликта	0
DES1500	Спуск<=-25ft/s	-g/3
CL1500	Взлет>=25ft/s	g/3
SDES1500	Спуск<=-25ft/s	-g/2.5
SCL1500	Взлет>=25ft/s	g/2.5
SDES2500	Спуск<=-42ft/s	-g/2.5
SCL2500	Взлет>=42ft/s	g/2.5

Таблица 3 описывает доступность каждого совета в зависимости от текущего отображаемого совета. Например, COC может быть выдан в любое время. Однако DES1500 и CL1500, будучи начальными советами, могут быть выданы только если в данныи момент пилоту отображается COC. SDES1500 может быть выдан после CL1500, SCL1500 и SCL2500, выступая как разворот, или после SDES2500, выступая как ослабление. Важно отметить, что SDES1500 не может следовать за COC и также не может следовать за DES1500 из-за их сходства по своеи природе. Таким образом, исходя из доступности каждого совета под текущим советом, агент на самом деле может выбирать только из трех деи ствии в каждом состоянии.

Таблица 3

Доступность советов

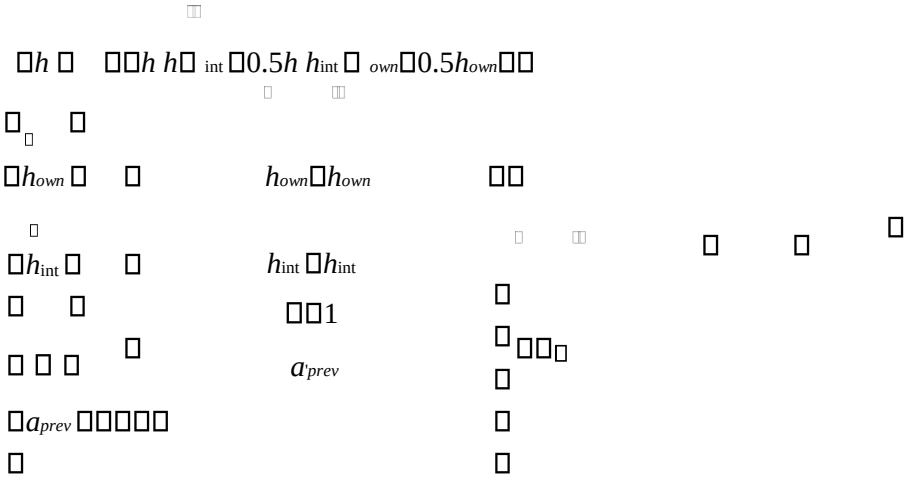
Действие	**Доступно от**
DES1500	COC
CL1500	COC
SDES1500	CL1500, SCL1500, SCL2500, SDE2500

SCL1500	DES1500, SDE1500, SDES2500, SCL1500	COC	В любое время
SDES2500	DES1500, SDES1500		
SCL2500	CL1500, SCL1500		

2) Динамическая модель самолета

Динамическая модель может быть записана как уравнение (4). Мы предполагаем временнои шаг в одну секунду, что означает обновление системы предотвращения столкновении с частотои 1 Гц. Для усложнения симуляционнои среды мы ограничиваем диапазон ускорении вторгающегося самолета значениями от [-a_{int}, a_{int}], и предполагаем, что скорость вторгающегося самолета h_{int} изменяется в направлении состояния столкновения на каждом шаге.

(4)



3) Формирование награды

В задачах предотвращения столкновении функция награды должна балансировать между безопасностью и эффективностью. Поэтому наша цель – предотвратить столкновения, минимизируя при этом выдачу отвлекающих предупреждении и неожиданных деи ствии системы (таких как усиление

рекомендации , изменение рекомендации или создание пересечения высот между самолетами). Кроме того, чтобы минимизировать влияние на другое воздушное пространство во время разрешения конфликта, мы поощряем весь процесс разрешения оставаться в определенном диапазоне высот. Таким образом, дизаи н функции награды в первую очередь разделен на две части.

Во-первых, в конце TTC нам необходимо, чтобы относительная высота самолетов поддерживалась на определеннои высоте h_{rel}, и штрафы налагаются за другие состояния, такие как столкновение, чрезмерно высокая или низкая относительная высота. Первая часть функции награды может быть сформулирована следующим образом:

$$R_1 = w_{NMAC} \cdot 1(|h_{rel}| < |h_{col}|) \cdot (|h_{col}| - |h_{rel}|) / |h_{col}| \quad (5)$$

$$= \max(w_{leav} \cdot (|h_{rel}| - |h_{rel\_s}|), 25) \cdot (|h_{rel}| - |h_{rel\_s}|)$$

Где w_{NMAC} штрафует за столкновение между собственным самолетом и нарушителем, h_{rel} представляет относительную высоту между двумя самолетами, h_{col} указывает высоту, на которои происходит конфликт между двумя самолетами, h_{rel_s} обозначает желаемую относительную высоту, которую мы хотим, чтобы два самолета достигли, а w_{leav} представляет штраф, накладываемыи , когда относительная высота между двумя самолетами чрезмерно велика.

В течение периода TTC нам необходимо обращать внимание на высоту собственного самолета, изменения в статусе предупреждения и ситуацию пересечения высот между двумя самолетами. Функция награды может быть сформулирована следующим образом:

$$R_2 = w_{leav} \cdot 1(|h_{own}| > |h_{upper}|) \cdot 1(\text{alert}) \cdot 1(\text{reversal}) \cdot \text{strength} \cdot \text{crossing} \cdot \exp(-t / tau) \cdot w_{COC} \quad (6)$$

w_{leav} представляет штраф, накладываемыи , когда высота собственного

самолета (h_{own}) превышает заданныи верхнии предел диапазона высот (h_{upper}). w_{Alert}, w_{stren}, $w_{reversal}$, w_{cros} штрафуют систему за выдачу предупреждении и ложных тревог. Среди них, τ представляет Время до Столкновения (TTC), и по мере приближения времени к TTC, штрафы за предупреждения и другие ложные тревоги усиливаются. w_{COC} предоставляет небольшую награду, когда предупреждение снято.

4) Метрики оценки производительности алгоритма

Для количественнои оценки эффективности алгоритма в решении проблемы предотвращения столкновении мы использовали следующие метрики оценки:

- **Результаты сходимости средней награды:** средняя награда за последние 100 эпизодов.
- **Результаты сходимости частоты столкновений:** средняя частота столкновении за последние 100 эпизодов.
- **Результаты сходимости коэффициента успеха координации высоты (ACSR):** средняя доля эпизодов за последние 100 эпизодов, в которых относительная высота самолетов была больше 900 футов, но меньше 1500 футов.
- **Результаты сходимости среднего количества предупреждений:** среднее количество предупреждении , выданных за эпизод, за последние 100 эпизодов.
- **Результаты сходимости среднего количества усилений советов:** среднее количество усилении советов за эпизод за последние 100 эпизодов.
- **Результаты сходимости среднего количества отмен советов:** среднее количество отмен советов за эпизод за последние 100 эпизодов.
- **Результаты сходимости среднего количества пересечений высот:** среднее количество раз, когда высоты обоих самолетов пересекались за эпизод, за последние 100 эпизодов.

Метод

В этом разделе мы представляем метод оптимизации алгоритма предотвращения воздушных столкновений с использованием ограниченного ресурсами обучения с подкреплением. Мы подробно рассматриваем формулировку RL с ограничением ресурсов. Наши эксперименты показывают, что информация, связанная с ресурсами, имеет ключевое значение для эффективного исследования и реализации основных функций системы предотвращения столкновений. Однако традиционные алгоритмы RL, как правило, игнорируют информацию о ресурсах и предполагают, что среда полностью известна. Чтобы решить эту проблему, мы моделируем задачу предотвращения столкновений с использованием RL с ограничением ресурсов, включая виртуальные ресурсы, и предлагаем подробное объяснение предлагаемого метода Time Resource Bonus (TRB).

1) Ресурсоограниченный MDP

В обучении с подкреплением (RL) определённые действия требуют ресурсов, таких как энергия в робототехнике или потребляемые предметы в видеоиграх. Пусть имеется d типов ресурсов. Обозначим вектор ресурсов как $\square\square^{d}\square$, а множество всех возможных векторов ресурсов — как $S_r \square^{d}\square$. Пространство состояний в проблемах с ограничением ресурсов расширяется до $S_R \square\square s\ s\square\square s\ s_o, {}_r\square,\square s_o\ \square S\ s,\square\ {}_r\ \square S_r\square$, что позволяет алгоритмам изучать информацию, связанную с ресурсами.

Для эффективного использования этой информации мы определяем функцию, учитывающую ресурсы, $I\ S: \square[\ ^{d}\ \square$, которая отображает состояния в ресурсы. Также мы вводим детерминированную функцию перехода ресурсов $f\ S: {}_R \square\ \square A\ S_r$, которая считается неизвестной, так как динамика ресурсов часто недоступна в реальных задачах. Распределение вероятности перехода задаётся следующим образом:

$p_R(s' \mid s, a), \quad p(s' \mid s, a) = p_o(o \mid s', a) \sum_{r \in \mathcal{R}} p(s', r \mid s, a), \quad (7)$

где $p_o(o \mid s', a) = \sum_{r \in \mathcal{R}} p(s', r \mid s, a)$. Мы обозначаем распределение вероятностей перехода в задачах с ограниченными ресурсами как p.

Проблемы с ограничением ресурсов формулируются в виде кортежа (S_R, A, p, r, I, f). Набор доступных действий для состояния s обозначается как $A(s)$, и $A(s) = \bigcup_{s \in S} A(s)$, причём $A(s)$ зависит от оставшихся ресурсов.

Допустимая политика $\pi(a \mid s)$ является функцией плотности вероятности над $A(s)$, и мы определяем множество допустимых политик как П. Подобно традиционным RL, ресурсоограниченное обучение с подкреплением нацелено на решение задачи оптимизации (1) для нахождения оптимальной допустимой политики.

Состояние s_2 доступно из s_1, если существует стационарная политика $\pi \in \Pi$ и временной шаг $t \in \mathbb{N}$, такой что $p(s_2 \mid s_1, t, \pi) > 0$. Если ресурсы невосполнимы и $\exists i \in [1, d], I(s_i) < I(s_i)$, то состояние s_2 недоступно из s_1.

Количество доступных ресурсов в состоянии s определяется как $I(s) = (I(s_1), \ldots, I(s_d))$, где $I(s_i) \geq 0$ для всех $i \in [1, d]$. Ресурсы i-го типа являются невосполнимыми, если их количество монотонно убывает, то есть $I(s_i)_t \geq I(s_i)_{t+1}$. Мы предполагаем, что I известно заранее, так как информация о текущих ресурсах обычно доступна в реальных задачах.

2) Проблема с ресурсными ограничениями в задаче предотвращения столкновений

При применении обучения с подкреплением для решения проблемы предотвращения столкновений в воздухе, несмотря на то что мы штрафовали за ненужные предупреждения, такие как эскалация и отмена рекомендаций через функцию вознаграждения, мы фактически позволяли этим предупреждениям

происходить бесконечно. В результате, хотя система могла завершить задачу по предотвращению столкновений, пилот подвергался множеству ненужных уведомлений. Чтобы решить эту проблему, мы ввели невосстанавливаемый виртуальный ресурс, при этом разные типы действий системы потребляют различные объемы ресурсов. Как только ресурсы исчерпаны, система больше не будет выдавать рекомендации. В рамках этой настройки агент должен не только завершить задачу для достижения высоких вознаграждений, но и минимизировать потребление ресурсов.

На основе условий задачи с ограничением ресурсов мы сначала продемонстрировали, что оба алгоритма RL, DQN и SAC, показали низкую эффективность использования образцов. Как показано на рисунке 2, оба алгоритма сходимости оказались менее успешными при наличии ограничений ресурсов по сравнению с ситуацией, когда таких ограничений не было. Мы также сравнили производительность обоих алгоритмов при различных уровнях ресурсов. Рисунки 3 и 4 иллюстрируют, что по мере увеличения доступности ресурсов производительность обоих алгоритмов улучшалась как в отношении предотвращения столкновений, так и в эффективности планирования. Таблицы 4, 5 и 6 соответствуют дополнительным показателям выполнения задач в ходе экспериментов, представленным на рисунках 2, 3 и 4 соответственно.

Результаты показывают, что после применения ограничений ресурсов количество предупреждений, выданных системой, значительно сократилось, но задачи по предотвращению столкновений и планированию были выполнены не на должном уровне. Рисунок 5 показывает траектории предотвращения столкновений во время обучения при ресурсных ограничениях для обоих алгоритмов. Можно увидеть, что оба алгоритма склонны выдавать непрерывные предупреждения на ранних стадиях сценария столкновения, тем самым быстро исчерпывая ресурсы. Таким образом, хотя эти методы частично решают проблему предотвращения столкновений, они не находят оптимальное решение.

В целом, эти методы страдают от неэффективного исследования в задаче предотвращения столкновений с ограничением ресурсов.

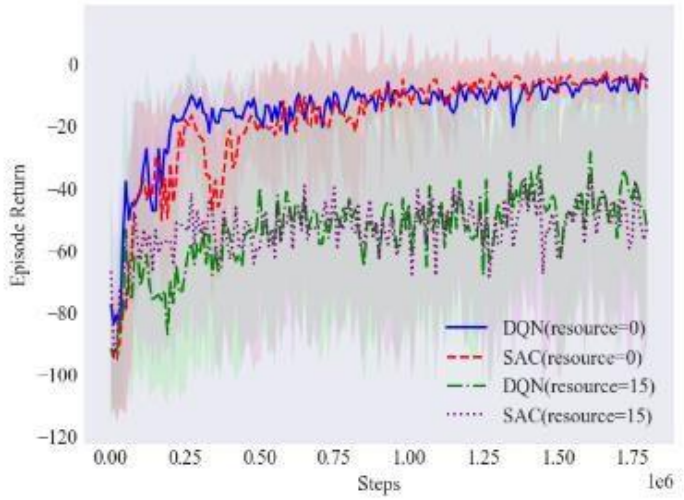


Рисунок 2 - Производительность обучения DQN и SAC в задачах предотвращения столкновений с ограничениями ресурсов и без них.

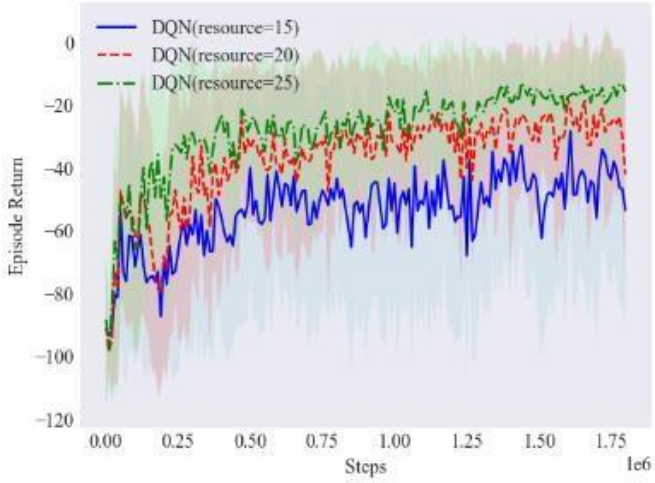


Рисунок 3 - Производительность обучения DQN в задачах предотвращения столкновений при различных ограничениях ресурсов.

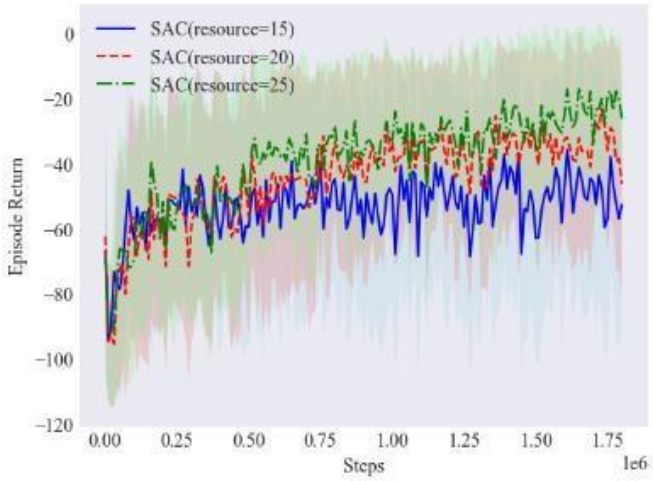


Рисунок 4 - Производительность обучения SAC в задачах предотвращения столкновений при различных ограничениях ресурсов.

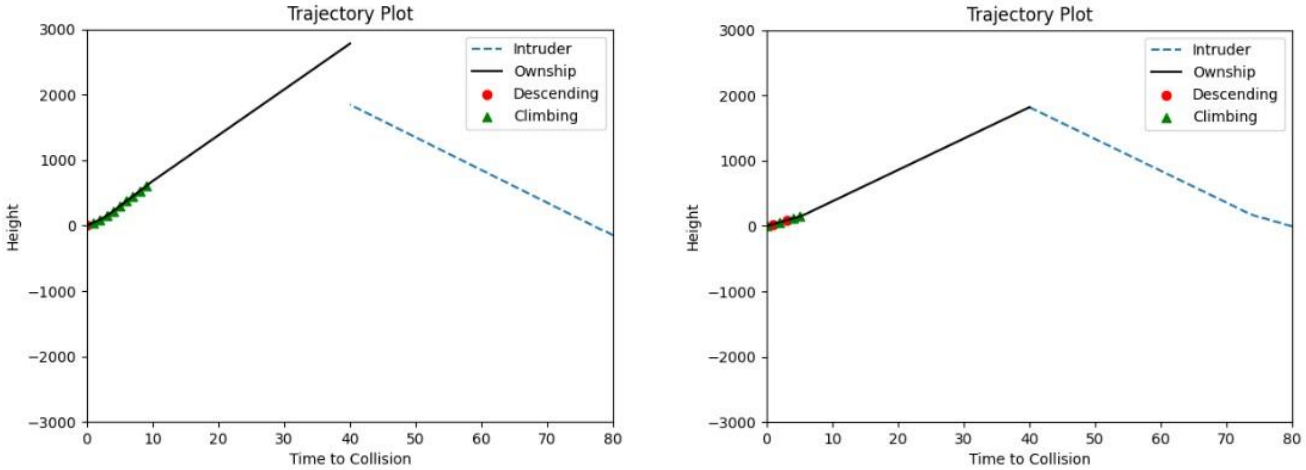


Рисунок 5 - Примеры траекторий предотвращения столкновений во время обучения для DQN (слева) и SAC (справа) при ограничениях ресурсов.

Таблица 4

Производительность dqn и sac в задачах предотвращения столкновений с ограничениями ресурсов и без них

Алгоритм	Частота столкновений	Уровень успеха координации	Количество оповещений	Количество укреплений оповещений	Количество изменений оповещений	Количество пересечений высоты
DQN(ресурс=0)	0.021	0.900	6.140	2.011	7.190	0.082
SAC(ресурс=0)	0.045	0.830	4.578	2.734	7.581	0.111
DQN(ресурс=15)	0.292	0.380	0.939	0.474	2.406	0.138

SAC(ресурс=15)	0.262	0.372	0.796	0.427	2.514	0.175

Таблица 5

Производительность dqn в задачах предотвращения столкновений с различными ограничениями ресурсов

Алгоритм	Частота столкновений	Уровень успеха координации	Количество оповещений	Количество укреплений оповещений	Количество изменений оповещений	Количество пересечений высоты
DQN(ресурс=15)	0.292	0.380	0.939	0.474	2.406	0.138
DQN(ресурс=20)	0.120	0.573	1.104	0.629	3.421	0.118
DQN(ресурс=25)	0.056	0.692	1.372	0.608	3.989	0.108

Таблица 6

Производительность обучения sac в задачах предотвращения столкновений с различными ограничениями ресурсов

Алгоритм	Частота столкновений	Уровень успеха координации	Количество оповещений	Количество укреплений оповещений	Количество изменений оповещений	Количество пересечений высоты
SAC (ресурс=15)	0.262	0.372	0.796	0.427	2.514	0.175
SAC(ресурс=20)	0.175	0.472	0.980	0.622	4.396	0.163
SAC(ресурс=25)	0.126	0.551	1.060	0.675	5.023	0.177

3) Time Resource Bonus

В задачах обучения с подкреплением с ограничениями ресурсов ресурсы играют решающую роль в эффективном исследовании. Мы наблюдали, что в данном состоянии размер доступного множества состоянии часто положительно коррелирует с оставшимися доступными ресурсами, так как состояния с высокими ресурсами не могут быть достигнуты из состоянии с низкими ресурсами. Ранее

использованные методы исследования на основе расширения [19] показали, что исследование состоянии с большими доступными наборами состоянии является важным для эффективного исследования. Другими словами, движение к состояниям с высокими ресурсами позволяет агенту достигать большого числа будущих состоянии , тем самым улучшая его исследование окружающеи среды. Однако при решении проблемы предотвращения столкновении с ограничениями виртуальных ресурсов оба алгоритма, DQN и SAC, не учитывают информацию, связанную с ресурсами, и склонны быстро исчерпывать ресурсы.

Чтобы обеспечить агенту по предотвращению столкновении поддержание высокои эффективности исследования при снижении количества ненужных предупреждении , мы предлагаем Time Resource Bonus (TRB). Этот алгоритм побуждает агента экономить ресурсы при выполнении задач по предотвращению столкновении и снижении количества ненужных предупреждении , таким образом исследуя состояния с более широким набором приемлемых состоянии .

Этот метод вводит временнои коэффициент дисконтирования в процессе потребления ресурсов. Мы определяем количество ресурсов, потребляемых агентом каждыи раз, как функцию g, связанную с коэффициентом дисконтирования. Мы требуем, чтобы функция g была возрастающеи функциеи времени. Поэтому в даннои статье мы определяем функцию g в следующеи форме:

$$\frac{t \cdot g(C(s,a)), \quad C(s,a) \neq 0}{T} \tag{8}$$

В этои формуле α — это гиперпараметр, представляющии чувствительность потребления ресурсов к времени. Чем больше значение α, тем ниже чувствительность потребления ресурсов к времени. В даннои статье мы произвольно устанавливаем α=T, где T — это максимальное значение интервала времени до конфликта (TTC) в процессе предотвращения столкновении . t обозначает текущее значение TTC, а $C(s,a)$ представляет собои количество

потребляемых ресурсов при выполнении деи ствия a в текущем состоянии s. Следовательно, мы определяем функцию, учитывающую ресурсы, как

$I\ s\square_{i\square 1}\square\square I\ s\square_{i}\square\square g\ C\ s\ a\square\square_{i,\ i}\square\square$. Мы используем внутреннее вознаграждение r_{int}, чтобы представить долю оставшегося виртуального ресурса, что влияет на исследование агента через форму внутренних вознаграждении . r_{int} принимает следующую форму:

$$r_{\text{int}}\square s\ a,\ \square\square \quad \frac{I_{\max}\square E_{s\ a\ g,\ \ldots,\ P,}\square\square_{T}\ g}{I_{\max}} \quad s\ \square \qquad (9)$$

Эксперименты и результаты

Наши эксперименты преследуют две основные цели:

1 проверить, превосходит ли алгоритм TRB алгоритмы DQN и SAC в одинаковых условиях ресурсов;

2 проанализировать сравнительную производительность алгоритма TRB с ограничениями виртуальных ресурсов и алгоритмов DQN и SAC без ограничений ресурсов.

1) Настрои ки симуляции

Параметры алгоритма установлены следующим образом. В соответствии с определением пространства состоянии и деи ствии алгоритм имеет 5 входов и 3 выхода. Архитектура неи роннои сети как для актера, так и для критика состоит из двух скрытых слоев с 64 и 32 узлами, с коэффициентом обучения 6e-5 и порогом отсечения градиента 3.0.

Размер пакета составляет 64, длина горизонта для исследования — 512, размер буфера для воспроизведения — 1e6, и сети повторно обновляются с использованием буфера воспроизведения, чтобы поддерживать малую потерю критика. Что касается формирования вознаграждении , коэффициент дисконтирования для будущих вознаграждении (гамма) установлен на уровне

0.99, а масштаб вознаграждения равен 1. Процесс обучения прекращается, если общее количество шагов превышает 1 миллион. Операционная система — Windows 11, а GPU — GeForce RTX 3050.

2) Экспериментальные результаты и анализ

Сначала мы сравнили производительность TRB с алгоритмами DQN и SAC в одинаковых условиях ресурсов. Мы эмпирически выбрали подходящее количество ресурсов (Ресурс = 20), которое отражает производительность алгоритмов. На рисунке 6 показано, что после включения внутреннего вознаграждения как

DQNTRB, так и SACTRB значительно превосходят стандартные алгоритмы DQN и SAC. Более того, хотя SAC в настоящее время считается современным алгоритмом (state-of-the-art), DQN и DQNTRB достигают лучших результатов сходимости по сравнению с SAC и SACTRB в контексте предупреждении о предотвращении столкновении в воздухе. Конкретные параметры предотвращения столкновении представлены в таблице 7.

Можно наблюдать, что DQNTRB и SACTRB снижают частоту столкновении и повышают уровень успешного согласования высоты по сравнению с DQN и SAC, но также влекут за собои более высокие затраты в отношении предупреждении (с увеличением ненужных предупреждении).

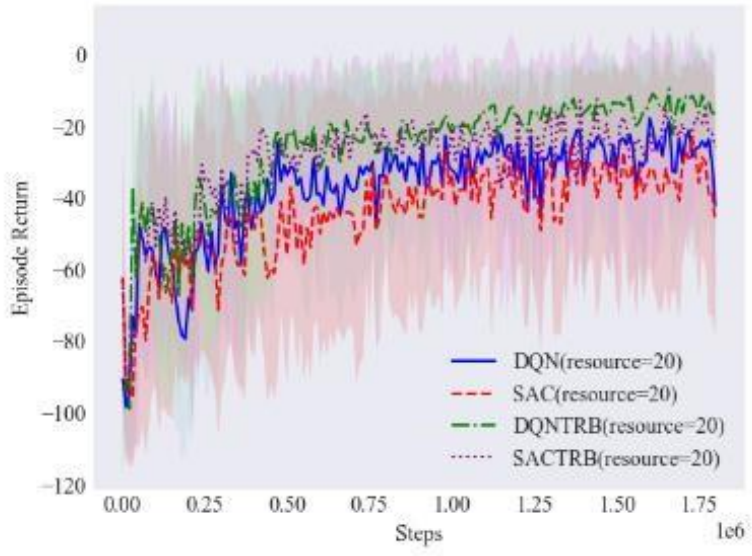


Рисунок 6 - Производительность обучения четырех алгоритмов — DQN, DQNTRB, SAC и SACTRB — в задачах предотвращения столкновений с одинаковыми ограничениями ресурсов.

Таблица 7

Производительность алгоритмов dqn, dqntrb, sac и sactrb в задачах предотвращения столкновений с одинаковыми ограничениями ресурсов

Алгоритм	Уровень успеха	Количество укреплений	Количество изменений	Количество пересечений	Количество столкновений	Частота оповещений
	Количество координации	оповещений	оповещений	высоты		
DQN(ресурс=20)	0.120	0.573	1.104	0.628	3.421	0.118
SAC(ресурс=20)	0.175	0.472	0.979	0.622	4.396	0.164
DQNTRB (ресурс=20)	0.056	0.717	1.856	0.729	3.436	0.123
SACTRB (ресурс=20)	0.078	0.657	1.565	0.889	5.352	0.114

Далее мы сравнили производительность обучения DQNTRB и SACTRB при различных ограничениях ресурсов с производительностью предотвращения столкновении DQN и SAC без ограничении ресурсов.

На рисунке 7 показана производительность обучения DQNTRB при различных ограничениях ресурсов, а также производительность обучения неограниченного алгоритма DQN в среде предупреждении о предотвращении столкновении .

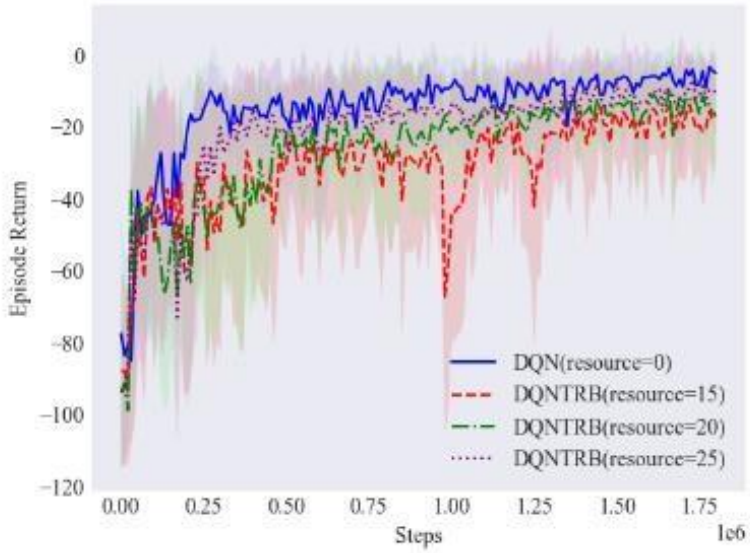


Рисунок 7 - Производительность обучения DQNTRB в задачах предотвращения столкновений при различных ограничениях ресурсов, вместе с производительностью DQN в задачах предотвращения столкновений без ограничений ресурсов.

Явно видно, что без ограничении ресурсов алгоритм DQN сходится быстрее всего и достигает лучших результатов. DQNTRB более чувствителен к изменениям уровня ресурсов. Конкретные параметры предотвращения столкновении в таблице 8 также отражают тот же результат, где алгоритм DQN демонстрирует оптимальную эффективность предотвращения столкновении и уровень успешного согласования высоты. Однако в отношении частоты предупреждении DQNTRB значительно уступает DQN.

Таблица 8 Производительность dqntrb в задачах предотвращения столкновений с различными ограничениями ресурсов и dqn в задачах предотвращения столкновений без ограничений ресурсов

Алгоритм	Частота столкновений	Уровень успеха координации	Количество оповещений	Количество укреплений оповещений	Количество изменений оповещений	Количество пересечений высоты
DQN(ресурс=0)	0.021	0.900	6.140	2.011	7.190	0.082
DQNTRB (ресурс=15)	0.082	0.662	1.146	0.682	4.307	0.134
DQNTRB (ресурс=20)	0.056	0.717	1.856	0.729	3.436	0.123
DQNTRB (ресурс=25)	0.041	0.799	2.247	0.889	4.332	0.102

На рисунке 8 показана производительность обучения SACTRB при различных ограничениях ресурсов и производительность алгоритма SAC без ограничении ресурсов в среде предотвращения столкновении . Аналогично рисунку 7, алгоритм SAC демонстрирует хорошие результаты сходимости; однако скорости сходимости различных алгоритмов схожи, и кривые обучения указывают на то, что SACTRB менее чувствителен к изменениям уровня ресурсов.

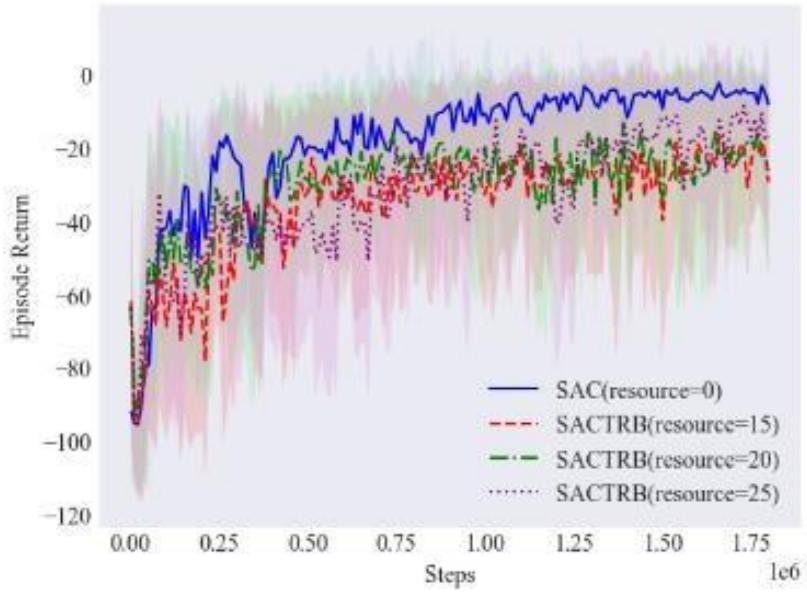


Рисунок 8 - Производительность обучения SACTRB в задачах предотвращения столкновений при различных ограничениях ресурсов, вместе с производительностью SAC в задачах предотвращения столкновений без ограничений ресурсов.

Конкретные параметры предотвращения столкновении в таблице 9 показывают, что при ограничениях ресурсов, в отличие от DQNTRB, алгоритм достигает наилучшеи частоты предотвращения столкновении и уровня успешного согласования высоты при количестве ресурсов 20, при этом поддерживая аналогичную эффективность предотвращения столкновении и значительно более низкую частоту предупреждении по сравнению с SAC.

Таблица 9

Производительность sactrb в задачах предотвращения столкновений с различными ограничениями ресурсов и sac в задачах предотвращения столкновений без ограничений ресурсов

Алгоритм	Частота столкновений	Уровень успеха координации	Количество оповещений	Количество укреплений оповещений	Количество изменений оповещений	Количество пересечений высоты
SAC(ресурс=0)	0.045	0.831	4.578	2.734	7.581	0.111
SACTRB (ресурс=15)	0.111	0.603	0.968	0.766	4.248	0.128
SACTRB (ресурс=20)	0.078	0.657	1.565	0.889	5.352	0.114
SACTRB (ресурс=25)	0.098	0.649	1.621	1.063	6.929	0.181

Заключение

В данной статье демонстрируется эффективность включения ограничений ресурсов в алгоритмы обучения с подкреплением для предотвращения столкновений в воздухе. Формулируя задачу как задачу МДП с ограничениями ресурсов и вводя механизм бонуса за время ресурсов (TRB), мы успешно оптимизировали как алгоритм DQN, так и SAC, в результате чего были получены DQNTRB и SACTRB.

Эти модифицированные алгоритмы достигли значительного сокращения ненужных предупреждений без ущерба для производительности предотвращения столкновений. Результаты подчеркивают потенциал обучения с подкреплением с учетом ресурсов для повышения эффективности и надежности систем предотвращения столкновений в условиях все более сложной воздушной среды.

Конфликт интересов

Авторы заявляют об отсутствии конфликта интересов.

Conflict of interest The authors declare no conflict of interest.

Список источников

1. Holland J.E., Kochenderfer M.J., Olson W.A. Optimizing the next generation collision avoidance system for safe, suitable, and acceptable operational performance // Air Traffic Control Quarterly. 2013. Vol. 21, no. 3. P. 275–297.

2. De D., Sahu P.K. A survey on current and next generation aircraft collision avoidance system // International Journal of Systems, Control and Communications. 2018. Vol. 9, iss. 4. P. 306–337.

3. Kochenderfer M.J., Holland J.E., Chryssanthacopoulos J.P. Next generation airborne collision avoidance system // Lincoln Laboratory Journal. 2012. Vol. 19, iss. 1. P. 17–33.

4. Kochenderfer M.J., Chryssanthacopoulos J.P. Robust airborne collision avoidance through dynamic programming : technical report / Massachusetts Institute of Technology, Lincoln Laboratory. 2011, 130 p. Project report ATC-371.

5. Optimized airborne collision avoidance / M.J. Kochenderfer, C. Amato, G. Chowdhary et al. // Decision making under uncertainty: theory and application. MIT Press, 2015. P. 249–276.

6. Julian K.D., Kochenderfer M.J., Owen M.P. Deep neural network compression for aircraft collision avoidance systems // Journal of Guidance, Control, and Dynamics. 2019. Vol. 42, iss. 3. P. 598–608.

7. Julian K.D., Kochenderfer M. Guaranteeing safety for neural network-based aircraft collision avoidance systems // 2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC). DOI 10.1109/DASC43569.2019.9081748.

8. Li S, Egorov M., Kochenderfer M. Optimizing collision avoidance in dense airspace using deep reinforcement learning // Thirteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2019). 2019. DOI 10.48550/arxiv.1912.10146.

9. Online multiple-aircraft collision avoidance method / P. Zhao, W. Wang, L.Ying et al. // Journal of Guidance, Control, and Dynamics. 2020. Vol. 43, iss. 2. P. 1–17.

DOI 10.2514/1.G005161.

10. A partially observable multi-ship collision avoidance decision-making model based on deep reinforcement learning // K. Zheng, X. Zhang, C. Wang et al. // Ocean & Coastal Management. 2023. Vol. 242. Art. 106689.

11. Kormushev P., Calinon S., Caldwell D.G. Reinforcement learning in robotics: Applications and real-world challenges // Robotics. 2013. Vol. 2, iss. 3. P. 122–148.

12. An aircraft collision avoidance method based on deep reinforcement learning / Z. Liu, E. Neretin, X. Gao, et al. // 9th International Conference on Control and Robotics Engineering (ICCRE), IEEE. 2024. P. 241–246.

13. Bhatia A., Varakantham P., Kumar A. Resource constrained deep reinforcement learning // Proceedings of the international conference on automated planning and scheduling. 2019. Vol. 29. P. 610–620.

14. Efficient exploration in resource-restricted reinforcement learning / Z. Wang, T. Pan, Q. Zhou et al. // Proceedings of the AAAI Conference on Artificial Intelligence. 2023 Vol. 37, no. 8. P. 10279–10287.

15. Sutton R.S., Barto A.G. Reinforcement learning: An introduction. Cambridge : MIT press, 2018.

16. Playing atari with deep reinforcement learning / V. Mnih, K. Kavukcuoglu, D. Silver, et al. // ArXiv.org : website / arXiv preprint arXiv:1312.5602: 2013. 9 p.

17. Human-level control through deep reinforcement learning / V. Mnih, K. Kavukcuoglu, D. Silver et al. // Nature. 2015. Vol. 518(7540). P. 529–533.

18. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor / T. Haarnoja, A. Zhou, P. Abbeel et al. // International conference on machine learning. PMLR, 2018. P. 1861–1870.

19. Mohamed S., Jimenez Rezende D. Variational information maximisation for intrinsically motivated reinforcement learning // Advances in neural information processing systems. 2015. 28 p.

References

1. Holland J.E., Kochenderfer M.J., Olson W.A. Optimizing the next generation collision avoidance system for safe, suitable, and acceptable operational performance // Air Traffic Control Quarterly. 2013. Vol. 21, no. 3. P. 275–297.

2. De D., Sahu P.K. A survey on current and next generation aircraft collision avoidance system // International Journal of Systems, Control and Communications. 2018. Vol. 9, iss. 4. P. 306–337.

3. Kochenderfer M.J., Holland J.E., Chryssanthacopoulos J.P. Next generation airborne collision avoidance system // Lincoln Laboratory Journal. 2012. Vol. 19, iss. 1. P. 17–33.

4. Kochenderfer M.J., Chryssanthacopoulos J.P. Robust airborne collision avoidance through dynamic programming : technical report / Massachusetts Institute of Technology, Lincoln Laboratory. 2011, 130 p. Project report ATC-371.

5. Optimized airborne collision avoidance / M.J. Kochenderfer, C. Amato, G. Chowdhary et al. // Decision making under uncertainty: theory and application. MIT Press, 2015. P. 249–276.

6. Julian K.D., Kochenderfer M.J., Owen M.P. Deep neural network compression for aircraft collision avoidance systems // Journal of Guidance, Control, and Dynamics. 2019. Vol. 42, iss. 3. P. 598–608.

7. Julian K.D., Kochenderfer M. Guaranteeing safety for neural network-based aircraft collision avoidance systems // 2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC). DOI 10.1109/DASC43569.2019.9081748.

8. Li S, Egorov M., Kochenderfer M. Optimizing collision avoidance in dense airspace using deep reinforcement learning // Thirteenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2019). 2019. DOI 10.48550/arxiv.1912.10146.

9. Online multiple-aircraft collision avoidance method / P. Zhao, W. Wang, L.Ying et al. // Journal of Guidance, Control, and Dynamics. 2020. Vol. 43, iss. 2. P. 1–17. DOI 10.2514/1.G005161.

10. A partially observable multi-ship collision avoidance decision-making model based on deep reinforcement learning // K. Zheng, X. Zhang, C. Wang et al. // Ocean & Coastal Management. 2023. Vol. 242. Art. 106689.

11. Kormushev P., Calinon S., Caldwell D.G. Reinforcement learning in robotics: Applications and real-world challenges // Robotics. 2013. Vol. 2, iss. 3. P. 122–148.

12. An aircraft collision avoidance method based on deep reinforcement learning / Z. Liu, E. Neretin, X. Gao, et al. // 9th International Conference on Control and Robotics Engineering (ICCRE), IEEE. 2024. P. 241–246.

13. Bhatia A., Varakantham P., Kumar A. Resource constrained deep reinforcement learning // Proceedings of the international conference on automated planning and scheduling. 2019. Vol. 29. P. 610–620.

14. Efficient exploration in resource-restricted reinforcement learning / Z. Wang, T. Pan, Q. Zhou et al. // Proceedings of the AAAI Conference on Artificial Intelligence. 2023 Vol. 37, no. 8. P. 10279–10287.

15. Sutton R.S., Barto A.G. Reinforcement learning: An introduction. Cambridge : MIT press, 2018.

16. Playing atari with deep reinforcement learning / V. Mnih, K. Kavukcuoglu, D. Silver, et al. // ArXiv.org : website / arXiv preprint arXiv:1312.5602: 2013. 9 p.

17. Human-level control through deep reinforcement learning / V. Mnih, K. Kavukcuoglu, D. Silver et al. // Nature. 2015. Vol. 518(7540). P. 529–533.

18. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor / T. Haarnoja, A. Zhou, P. Abbeel et al. // International conference on machine learning. PMLR, 2018. P. 1861–1870.

19. Mohamed S., Jimenez Rezende D. Variational information maximisation for intrinsically motivated reinforcement learning // Advances in neural information processing systems. 2015. 28 p.

Получено 30 сентября 2025 ● Принято к публикации 12 декабря 2025 ● Опубликовано 30 декабря 2025
Received 30 September 2025 ● Accepted 12 December 2025 ● Published 30 December 2025