

© 2025 г. Е.Л. КУЛИДА, канд. техн. наук (elena-kulida@yandex.ru),
В.Г. ЛЕБЕДЕВ, д-р техн. наук (lebedev-valentin@yandex.ru)
(Институт проблем управления им. В.А. Трапезникова РАН, Москва)

ОБЗОР МЕТОДОВ ПРЕДУПРЕЖДЕНИЯ КОНФЛИКТОВ ПРИ УПРАВЛЕНИИ ВОЗДУШНЫМ ДВИЖЕНИЕМ С ПОМОЩЬЮ ГЛУБОКОГО ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ

Представлен обзор развития современных подходов к предупреждению конфликтов между воздушными судами на основе глубокого обучения с подкреплением. Рассмотрена базовая концепция обучения с подкреплением и некоторые основные алгоритмы, используемые для предупреждения конфликтов воздушных судов. Представлены модели с дискретными и непрерывными действиями по предупреждению конфликтов в двумерном и трехмерном воздушном пространстве при движении по фиксированным траекториям или в свободном полете. Рассмотрены различные подходы к представлению информации о состоянии воздушного пространства (с помощью вектора состояния и в виде графа) и разные типы взаимодействия между воздушными судами (на основе информации о состоянии окружающих воздушных судов или при помощи обмена сообщениями).

Ключевые слова: управление воздушным движением, предупреждение конфликтов, маневр воздушного судна, глубокое обучение с подкреплением.

DOI: 10.31857/S0005231025090048, EDN: VMUGQH

1. Введение

Важнейшей функцией системы управления воздушным движением является поддержание безопасного расстояния разделения как по вертикали, так и по горизонтали между любыми двумя самолетами в любое время [1]. Нарушение безопасного разделения между воздушными судами (ВС) называется конфликтом. Растущая плотность воздушного движения приводит к увеличению числа потенциальных конфликтов, методы их предупреждения имеют важное значение для снижения рисков столкновений. Стратегическое предупреждение потенциальных конфликтов на основе глобальной модели оптимизации, которая заблаговременно генерирует бесконфликтные четырехмерные траектории для всех ВС, не позволяет справиться с неопределенностью, возникающей в динамике полетов в реальном времени [2]. Tактическое предупреждение потенциальных конфликтов в режиме реального времени критически важно для обеспечения безопасного управления воздушным движением, поскольку позволяет лучше справляться с неопределенностью, возникающей в динамике полетов.

В настоящее время движение ВС в основном осуществляется по фиксированным трассам, и ответственность за предупреждение конфликтов лежит на авиадиспетчерах. Тактические решения по-прежнему принимаются авиадиспетчерами с очень небольшими изменениями по сравнению с решениями, принимавшимися 50 лет назад [3]. С ростом интенсивности воздушного движения нагрузка авиадиспетчеров постоянно увеличивается и может презойти человеческие возможности. Развитие существующих методов оценки динамической воздушной обстановки авиадиспетчером с целью сокращения факторов опасности, включая информационную перегрузку и дефицит времени на функциональные операции, рассматривается в [4]. В [5–7] предложена концепция разработки средств автоматизации для увеличения пропускной способности и безопасности эксплуатации воздушного пространства с целью эффективной интеллектуальной поддержки принятия решений авиадиспетчерами для предупреждения конфликтов. Перспективная организация воздушного движения предполагает использование свободных полетов, когда ВС будут двигаться по произвольным траекториям, и предупреждение конфликтов будет обеспечиваться автономной системой управления воздушным движением. Теоретические исследования подтверждают, что свободный полет обладает потенциалом для повышения безопасности [8] и снижения расхода топлива [9]. Для реализации концепции свободного полета необходимо «обеспечивать контроль за эшелонированием ВС с помощью бортовых систем в дополнение к наземным системам. Более высокая надежность такой структуры позволит внести ясность в целый ряд сценариев моделирования риска столкновения ВС и будет способствовать созданию более безопасных, гибких и емких условий организации воздушного движения» [10]. основополагающее значение для реализации концепции свободного полета имеет сложнейшая задача разработки децентрализованных автономных средств предупреждения конфликтов [11]. Методы и алгоритмы обнаружения и предотвращения опасных сближений в воздухе в рамках перспективной системы организации воздушного движения с учетом требований по безопасности и эффективности выполнения полетов исследовались в [12, 13].

Разработаны методы, позволяющие предоставить авиадиспетчерам рекомендации по разрешению конфликтов, на основе оптимального управления [14], математического программирования [15–18], геометрической оптимизации [19–21], эволюционных алгоритмов [22], алгоритма поиска по дереву по методу Монте-Карло [23]. Эти методы работают при существующей плотности воздушного движения, но при большей плотности проблемой является их недостаточная вычислительная эффективность. Требуется десятки или даже сотни секунд для получения решения. Учет неопределенности, присущей воздушному движению, значительно увеличивает время вычислений, тем самым снижая способность методов быстро генерировать решения. Большинство традиционных подходов к обеспечению эшелонирования не справляются со стохастическими средами и высокой плотностью воздушного движения [24]. Необходимы новые подходы, которые смогут эффективно реагиро-

вать на динамику внешней среды в режиме реального времени, например, на основе нейронных сетей и машинного обучения [25–27].

В последнее время, благодаря способности решать проблемы принятия решений, которые раньше были недоступны из-за сочетания нелинейности и высокой размерности, широкое применение в различных областях авиации получило глубокое обучение с подкреплением [28]. Использование глубокого обучения с подкреплением позволит автоматически получать безопасные и эффективные варианты решений по предупреждению конфликтов для поддержки принятия решений авиадиспетчерами и снизит их рабочую нагрузку [29]. В будущем полностью автоматизированная система управления станет окончательным решением для обработки высокоплотного, сложного и динамичного воздушного движения [30].

Применение методов обучения с подкреплением состоит из двух этапов: этап обучения модели и этап применения обученной модели на практике. Обучение модели может потребовать значительного времени, зато потом с помощью обученной модели решения получаются очень быстро. Скорость принятия решений – показатель эффективности и преимущество глубокого обучения с подкреплением по сравнению с традиционными алгоритмами. Более быстрое принятие решений означает более раннее обнаружение конфликтов и выработку инструкций для снижения рабочей нагрузки авиадиспетчеров и пилотов. В [31] приводятся следующие цифры (с оговоркой, что данные получены в разных условиях расчета): алгоритму на основе смешанно-целочисленного линейного программирования для генерации решения требуется 49 с, среднее время решения с помощью генетического алгоритма составляет 37,6 с, а обученному агенту глубокого обучения с подкреплением требуется менее 0,2 с. Методы обучения с подкреплением по сравнению с традиционными методами имеют явное преимущество в скорости вычисления решений и способности адаптироваться к динамике внешней среды, что критически важно при разрешении конфликтов в воздушном движении.

Исследования по разрешению конфликтов в воздушном движении с использованием глубокого обучения с подкреплением постоянно проводятся с 2018 г., и за это время было предложено множество моделей и алгоритмов [31]. Рассматриваются модели разрешения конфликтов и для случая движения по трассам, и для свободного полета, модели как с дискретными, так и с непрерывными действиями. В настоящее время в большинстве случаев предлагаются двумерные модели для разрешения конфликтов с помощью горизонтальных маневров и значительно реже трехмерные – для горизонтальных и вертикальных маневров [32]. Двумерные модели игнорируют вертикальные маневры из-за потенциальной нестабильности, которую они могут вызвать в воздушном движении [33]. Различается количество конфликтующих ВС: рассматриваются конфликты между двумя ВС, в группах ВС с фиксированным или с переменным количеством ВС. В [34–36] разработан гибридный подход, объединяющий для разрешения конфликтов сильные стороны геометрических методов и методов обучения с подкреплением. Утверждает-

ся, что большой диапазон различных оптимальных решений, найденных методом обучения с подкреплением, показывает, что правила геометрического метода должны быть расширены, учитывая различные геометрии конфликта. В РФ задача управления движением ВС на основе обучения с подкреплением в настоящее время активно исследуется специалистами ГосНИИАС, и уже имеется значительный научно-технический задел [37].

Поскольку существует большое разнообразие постановок задачи разрешения конфликтов, во многих случаях построение модели обучения с подкреплением начинается с разработки интерактивной среды, в которой агент может обучаться различным стратегиям. Агент искусственного интеллекта учится при помощи алгоритмов глубокого обучения с подкреплением методом проб и ошибок, применяя различные возможные действия и получая обратную связь от среды в виде вознаграждения. Цель агента заключается в изучении такой стратегии выбора действий, которая будет максимизировать математическое ожидание суммарных дисконтированных вознаграждений за продолжительное время. Сходимость модели обучения с подкреплением к нужному для решения поставленной задачи результату определяется выбором функции вознаграждения, с помощью которой агент учится оптимизировать стратегию выбора действий в различных ситуациях. Функция вознаграждения оказывает влияние на скорость обучения, сходимость и производительность агентов. Главное, что должно учитываться в функции вознаграждения – это удача или неудача разрешения конфликта. Кроме этого, для повышения эффективности модели вознаграждение должно учитывать количество маневров и время, необходимые для разрешения конфликта. При помощи функции вознаграждения можно приблизить поведение агента к существующим правилам разрешения конфликтов авиадиспетчерами [38, 39].

Несмотря на успешность обучения с подкреплением в исследованиях по решению задач управления воздушным движением, для применения этого метода в реальных условиях остаются две существенные проблемы. Первая проблема заключается в уязвимости глубоких нейронных сетей к состязательным атакам, вторая – проблема объяснимости моделей «черного ящика» – в непонимании пилотами и авиадиспетчерами механизма принятия моделями тех или иных решений [40, 41]. В [42] предлагаются подходы к решению этих проблем при автономном разрешении конфликтов ВС.

2. Обучение с подкреплением

2.1. Базовая концепция [43]

Машинное обучение подразделяется на контролируемое обучение, неконтролируемое обучение и обучение с подкреплением. Контролируемое обучение позволяет аппроксимировать любую функцию, но требует наличия наборов размеченных данных, которые доступны далеко не всегда. Неконтролируемое обучение предполагает наличие наборов неразмеченных данных. Обучение с подкреплением позволяет реализовать последовательный процесс

принятия решений методом проб и ошибок, синтезируя данные для обучения в процессе взаимодействия агента со средой. При использовании обучения с подкреплением каждое ВС моделируется как интерактивный агент, действиями которого являются маневры по предупреждению конфликтов.

В основе обучения с подкреплением лежит модель марковского процесса принятия решений, при котором состояние системы и действия агента не зависят от того, каким образом система пришла в это состояние. Для марковского процесса принятия решений справедливы уравнения Беллмана. Основной модели марковского процесса принятия решений являются среда и действующий в ней агент. Среда характеризуется набором параметров, состояние среды s – это определенный набор значений этих параметров. Агент – это программа, которая может анализировать состояние среды и выполнять в каждом состоянии определенный набор действий $a \in A(s)$. В результате выполнения агентом действия среда переходит из состояния s в новое состояние s' и получает обратную связь от среды в виде вознаграждения $r = R(s, a, s')$. При многошаговом взаимодействии агента со средой, начиная с состояния s на шаге t до конца эпизода на шаге T , определяется суммарная дисконтированная выгода:

$$G_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots + \gamma^{T-1} r_T = r_{t+1} + \gamma G_{t+1},$$

$\gamma \in [0, 1]$ – коэффициент дисконтирования, который определяет, насколько убывает ценность награды, полученной на более поздних шагах.

Функция перехода $p(s'|s, a)$ определяет вероятность перехода в состояние s' на шаге t при условии, что на шаге $t - 1$ в состоянии s было выбрано действие a :

$$p(s'|s, a) = P(S_t = s' | S_{t-1} = s, A_{t-1} = a), \quad \sum_{s' \in S} p(s'|s, a) = 1, \forall s \in S, \forall a \in A(s).$$

Стратегией (или политикой) называется функция $\pi(a|s)$, сопоставляющая действие агента каждому нетерминальному состоянию среды.

Ожидаемая выгода при следовании агентом стратегии π в состоянии s называется функцией ценности состояния:

$$V_\pi(s) = E_\pi[r_{t+1} + \gamma G_{t+1} | S_t = s].$$

Однако функция ценности состояния не позволяет узнать ожидаемую выгоду от выполнения агентом действия a в состоянии s при следовании агентом стратегии π , она определяется функцией ценности действия:

$$Q_\pi(s, a) = E_\pi[r_{t+1} + \gamma G_{t+1} | S_t = s, A_t = a].$$

Базовая концепция обучения с подкреплением – обобщенная итерация по стратегиям – это итеративная процедура. Шаг этой процедуры включает два процесса: оценка текущей стратегии для уточнения текущего приближения функции ценности и затем улучшение стратегии в соответствии с изменившейся функцией ценности.

Функция ценности может быть представлена в виде

$$V_{\pi}(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r|s, a)[r + \gamma V_{\pi}(s')], \forall s \in S.$$

Шаг итерации для уточнения функции ценности заключается в ее вычислении для действия с наиболее высокой ценностью:

$$V_{k+1}(s) = \max_a \sum_{s',r} p(s', r|s, a)[r + \gamma V_k(s')].$$

Действие, доставляющее максимальное значение Q -функции, называется жадным. Стратегия может быть улучшена при помощи алгоритма оптимизации стратегий, который заключается в выборе жадного действия относительно Q -функции:

$$\pi'(s) = \operatorname{argmax}_a \sum_{s',r} p(s', r|s, a)[r + \gamma V_{\pi}(s')].$$

В процессе изучения функции ценности очень важно соблюдать баланс между выбором жадного действия и выбором случайного действия для исследования. Существует много различных подходов к решению этой проблемы: эпсилон-жадная стратегия, при которой случайное действие выбирается с вероятностью эпсилон; затухающая эпсилон-жадная стратегия, при которой эпсилон уменьшается по мере обучения агента; стратегии выбора действий, использующие полученные к текущему шагу знания о ценности и изученности действий, и др.

Рассмотренные два процесса стабилизируются, когда функция ценности будет соответствовать стратегии, а стратегия будет жадной по отношению к функции ценности. Стратегия и функция ценности оптимальны, если:

$$V^*(s) = \max_{\pi} V_{\pi}(s).$$

2.2. Алгоритмы глубокого обучения с подкреплением [44, 45]

Для простых примеров функция ценности действий представляется в виде таблицы. Для важных практических задач табличное представление функций ценности реализовать невозможно из-за большого числа состояний, непрерывных переменных или действий. В этом случае используется глубокое обучение с подкреплением, при котором для аппроксимации функций ценности используются алгоритмы глубокого обучения с учителем на основе выборок (s, a, r, s') , сформированных в процессе взаимодействия агента со средой.

В настоящее время в области предупреждения авиационных конфликтов этот подход бурно развивается, предлагаются и исследуются различные алгоритмы аппроксимации функции ценности и оптимизации стратегий. Широко

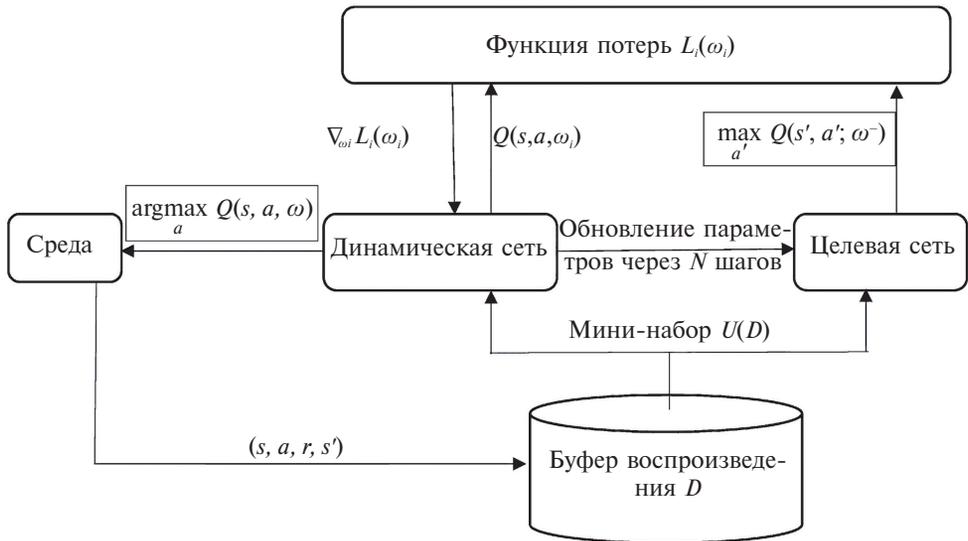


Рис. 1. Принцип работы алгоритма глубокого Q -обучения.

применяется алгоритм глубокого Q -обучения (deep Q -network – DQN), который использует две ключевые технологии: воспроизведение опыта и двойную сетевую структуру. На рис. 1 показан принцип работы алгоритма DQN [46].

Воспроизведение опыта заключается в создании буфера воспроизведения D , в котором накапливается большое количество выборок. Мини-наборы $U(D)$ для обучения сети выбираются из накопленного буфера воспроизведения равномерно случайно и, таким образом, соответствуют различным траекториям и политикам, что повышает стабильность обучения сети. Двойная сетевая структура основана на использовании одной и той же сети с разными наборами параметров. Динамическая сеть используется для аппроксимации текущего значения $Q(s, a; \omega_i)$, параметры этой сети ω_i обновляются на каждом временном шаге i . Целевая сеть используется для получения более устойчивого целевого значения Q , параметры целевой сети ω^- обновляются через N временных шагов. Функция потерь имеет вид

$$L_i(\omega_i) = E_{(s,a,r,s') \sim U(D)} \left[\left(r + \gamma \max_{a'} Q(s', a'; \omega^-) - Q(s, a; \omega_i) \right)^2 \right].$$

Функция потерь оптимизируется методом градиентного спуска:

$$\begin{aligned} \nabla_{\omega_i} L_i(\omega_i) &= \\ &= E_{(s,a,r,s') \sim U(D)} \left[\left(r + \gamma \max_{a'} Q(s', a'; \omega^-) - Q(s, a; \omega_i) \right) \nabla_{\omega_i} Q(s, a; \omega_i) \right]. \end{aligned}$$

Отличие метода двойной глубокой Q -сети (Double Deep Q -network – DDQN) [47] от метода DQN заключается в том, что для выбора действия в уравнении вместо параметров целевой сети используются параметры динамической

сети:

$$\nabla_{\omega_i} L_i(\omega_i) = E_{(s,a,r,s') \sim U(D)} \left[\left(r + \gamma Q(s', \underset{a'}{\operatorname{argmax}} Q(s', a'; \omega_i); \omega^-) - Q(s, a; \omega_i) \right) \nabla_{\omega_i} Q(s, a; \omega_i) \right].$$

Алгоритмы DQN и DDQN используются в моделях предупреждения конфликтов с дискретным пространством действий [48–52].

Предлагаются также методы, использующие для разрешения конфликтов между ВС непрерывное пространство действий [34, 53–55]. В этом случае стратегии представляются параметризованными стохастическими функциями $\pi_\theta(a, s)$, которые оптимизируются при помощи алгоритмов актер–критик. В этих алгоритмах, помимо нейронной сети для оценки стратегии (критик), используется вторая нейронная сеть для формирования стратегии на основе оптимизации функции ценности (актер).

Для траектории $\tau = S_0, A_0, R_1, S_1, \dots, S_{T-1}, A_{T-1}, R_T, S_T$ функция $G(\tau)$ – полная дисконтированная выгода, $\pi_\theta(A_t|S_t)$ – вероятность выбора действия A_t в состоянии S_t на шаге t . Сеть актера обновляется в соответствии с градиентом функции ценности:

$$\nabla_\theta E_{\tau \sim \pi_\theta} [G(\tau)] = E_{\tau \sim \pi_\theta} \left[\sum_{t=0}^T G_t(\tau) \nabla_\theta \log \pi_\theta(A_t|S_t) \right].$$

3. Модели предупреждения конфликтов между воздушными судами

3.1. Модели предупреждения конфликтов между двумя воздушными судами

Стратегия разрешения конфликтов может включать как двумерные (2D) маневры – изменение курса и скорости в плоском воздушном пространстве, так и трехмерные (3D) маневры с изменением еще и высоты полета. Хотя 2D-модель не так эффективна в разрешении конфликтов, как 3D-модель, маневры в двумерном воздушном пространстве вызывают меньший дискомфорт у пассажиров и не искажают вертикально стратифицированную структуру воздушного пространства [56]. Слишком большие нейронные сети могут быть пагубными для обучения агента из-за слишком большого числа параметров. Поэтому меньшие нейронные сети для 2D-моделей имеют больший потенциал для дальнейшего расширения модели, чтобы учитывать больше факторов реального мира [30].

Первые исследования по предотвращению конфликтов между ВС при помощи обучения с подкреплением рассматривали разрешение конфликтов между двумя ВС в двумерном воздушном пространстве. В одной из первых работ по предотвращению конфликтов между ВС рассматривались полеты по маршрутам и была предложена иерархическая структура глубокого обучения с подкреплением [48]. В качестве среды обучения использовалось программное обеспечение под названием NASA Sector 33, содержащее 35 задач по

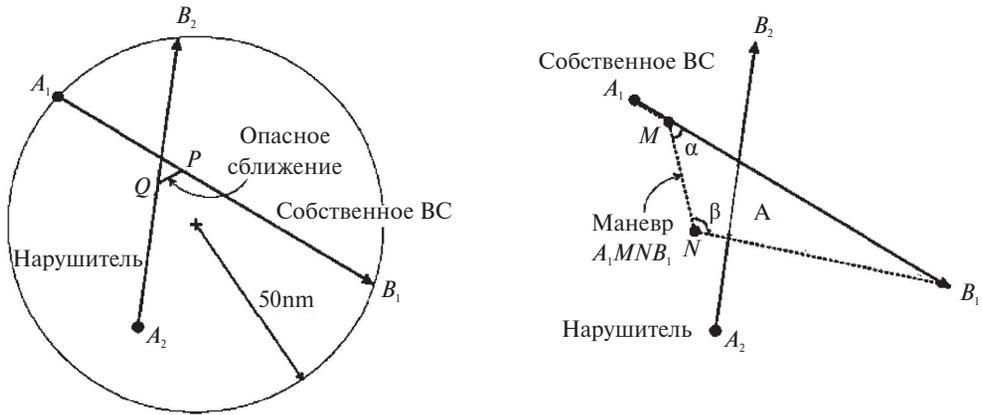


Рис. 2. Сценарий конфликта двух ВС и маневр по его предупреждению.

управлению воздушным движением с участием от двух до пяти самолетов. Иерархическая структура включает родительский агент, предназначенный для решения задачи выбора маршрутов самолетов в начале эпизода, затем дочерний агент управляет действиями по изменению скоростей на выбранных маршрутах. Иерархическая структура позволяет разделить действия по выбору маршрутов, выполняемых в начале эпизода, и действия по управлению скоростями в процессе эпизода. Для обучения агентов используется алгоритм обучения с подкреплением на основе двойной глубокой Q -сети. Первая нейронная сеть (целевая) используется для выбора действий, жадных по отношению к текущей Q -функции, вторая нейронная сеть (динамическая) используется для корректировки Q -функции на основе оценки успешности выполненных действий. В [48] показано, что иерархический агент глубокого обучения с подкреплением может выбрать оптимальные комбинации маршрутов и скоростей, чтобы избежать конфликта двух самолетов при полетах по маршрутам.

В одной из первых работ по обучению с подкреплением для разрешения конфликта при свободном полете рассматривается случай двух самолетов с учетом неопределенности [53]. Разработана среда для моделирования потенциальных конфликтов для обучения и тестирования агента. На рис. 2 представлен конфликт двух ВС в круговом воздушном пространстве с радиусом R , равным 50 морских миль, и маневр по его предотвращению. Траектория собственного ВС – A_1B_1 , траектория нарушителя – A_2B_2 , QP – ближайшее расстояние между ВС, на котором они теряют безопасное разделение, если ни одно из ВС не совершит маневра. В качестве действия по предупреждению конфликта используется однократный маневр по изменению курса в непрерывном двумерном пространстве. Маневр A_1MNB_1 на рис. 2 представляет ряд действий, выполняемых собственным ВС: отклонение от первоначального пути в точке M изменением курса на угол α , затем движение по вектору MN , в точке N поворот к точке B_1 .

Вознаграждение обучающегося агента рассчитывается в виде суммы:

$$R_{final} = 5 + R_{conflict} + R_{maneuver} + R_{deviation},$$

т.е. к вознаграждению 5 добавляются отрицательные вознаграждения: $R_{conflict} = -3$, если маневр не разрешает конфликт; $R_{maneuver} = -2$, в случае выхода за границы области или если угол $\beta > 120^\circ$; $R_{deviation} = -S_{\Delta MNB_1}$, отклонение от исходной траектории оценивается площадью между траекторией маневра и исходной траекторией.

Используется метод глубокого градиента по детерминированным политикам (Deep Deterministic Policy Gradient – DDPG) – один из передовых методов глубокого обучения с подкреплением для задач управления с непрерывным пространством действий [44]. Алгоритм использует две нейронные сети: сеть критика для изучения функции полезности пар состояние–действие $Q(s, a)$ и сеть актера для отображения состояния в детерминированное действие на основе градиента политики. Производительность алгоритма DDPG для предупреждения конфликтов в воздушном движении близка к производительности традиционных методов, но время расчетов при этом значительно сокращается [57].

В [54, 56, 58, 59] постановка задачи предупреждения конфликтов обобщается на случай, когда помимо собственного ВС и нарушителя в области находится еще несколько ВС. В этом случае при маневрировании для предупреждения конфликта могут возникнуть вторичные потенциальные конфликты (эффект домино) [60].

В [54] разработана двумерная среда для моделирования свободных полетов, которая может быть применена для нескольких ВС в секторе (не более 5) (рис. 3).

Предполагается, что между этими ВС нет конфликтов, конфликты могут быть вызваны входящим в сектор ВС, цель которого – пролететь от начальной точки до конечной за минимальное время без столкновений с другими ВС. Предлагается алгоритм «актер–критик» с фиксированным количеством K действий (циклов управления). Чтобы избежать конфликтов, генерируются действия по изменению курсового угла входящего ВС. На каждом временном шаге агент выбирает действие $A = \{\rho, \varphi | \rho \in [0, L], \varphi \in [-\pi, \pi]\}$, которое описывается двумерной полярной координатой, где ρ, φ – полярный радиус и угол.

Функция вознаграждения

$$R_t = \begin{cases} -1, & \text{если есть конфликт,} \\ 1 - \frac{1}{K} \times \frac{|\Delta\varphi_t|}{\pi} & \text{в противном случае.} \end{cases}$$

Функция ценности аппроксимируется с помощью нейронной сети $\hat{V}(S_t, \omega) \approx V_\pi(S)$, где ω – веса нейронов.

Для сети критика определяется δ :

$$\delta_t = R_t + \gamma \hat{V}(S_{t+1}, \omega) - \hat{V}(S_t, \omega),$$

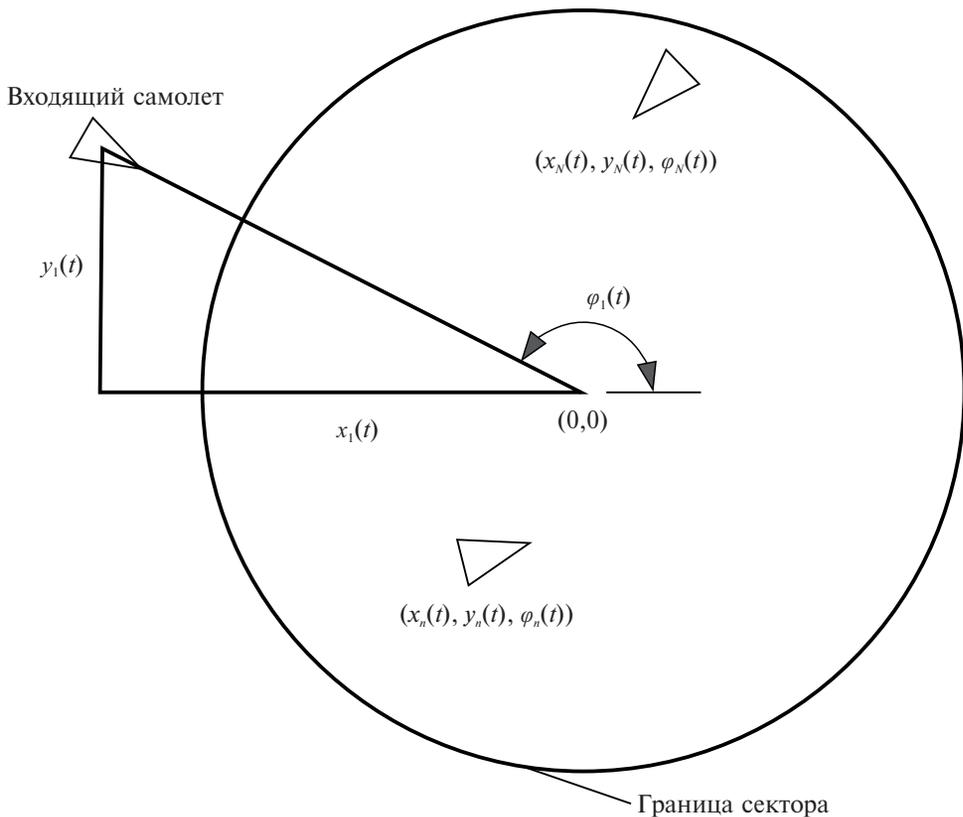


Рис. 3. Сценарий конфликта с несколькими ВС.

где R_t – немедленное вознаграждение, $\hat{V}(S_{t+1}, \omega)$ – значение ценности следующего состояния, $\hat{V}(S_t, \omega)$ – значение ценности текущего состояния. Для обновления параметров применяется метод наименьших квадратов: $\omega \leftarrow \omega + \alpha \nabla \delta^2$, α – скорость обучения.

Для сети актера используется метод градиента политики. Для действия (ρ, φ) применяется уравнение:

$$\ln \pi(\rho_t, \varphi_t | S_t, \theta) = \ln \pi(\rho_t | S_t, \theta) + \ln \pi(\varphi_t | S_t, \theta),$$

где $\pi(\rho_t, \varphi_t | S_t, \theta)$ – вероятность выбора ρ и φ в состоянии S_t с параметрами θ , $\pi(\rho_t | S_t, \theta)$ – вероятность выбора ρ в состоянии S_t с параметрами θ , $\pi(\varphi_t | S_t, \theta)$ – вероятность выбора φ в состоянии S_t с параметрами θ . Параметры θ обновляются по формуле:

$$\theta \leftarrow \theta + \alpha^\theta \delta_t \nabla \ln \pi(\rho_t, \varphi_t | S_t, \theta).$$

Переменное количество циклов управления не отражено в функции вознаграждения, и его нельзя корректировать динамически. Поскольку хорошо обученный агент обладает очень высокой вычислительной эффективностью, предлагается обучить несколько агентов с разными значениями K , например $K = 1, 2, 3$. В реальном процессе управления воздушным движением мож-

но быстро рассчитать несколько вариантов управления и выбрать лучшее решение из нескольких вариантов. Результаты моделирования подтвердили возможность применения глубокого обучения с подкреплением для разрешения конфликтов и его преимущество в вычислительной эффективности по сравнению с известными методами.

В [61] предложена и проанализирована модель для разрешения конфликтов между двумя ВС, учитывающая неопределенность, связанную с ветром. Предложенный метод предупреждения конфликтов применим в случае, когда неопределенность, связанная с направлением и скоростью ветра, непостоянна (стохастична) на протяжении всего моделирования.

В [46] рассматривается стратегия разрешения конфликтов двух самолетов в трехмерном пространстве на основе глубокого обучения с подкреплением. Из сценария воздушного движения, который может содержать несколько конфликтов, для разрешения выбирается один конфликт двух самолетов. Модель разрешения конфликтов моделируется как дискретный по времени марковский процесс принятия решений. Агент использует команды корректировки высоты, корректировки скорости или корректировки курса для разрешения конфликта. Предпочтения авиадиспетчера по выбору маневров разрешения конфликтов передаются агенту при помощи корректировки функции вознаграждения.

3.2. Модели предупреждения конфликтов между фиксированным количеством воздушных судов

Многоагентное обучение с подкреплением рассматривает набор агентов, которые взаимодействуют с одной и той же средой [62]. Каждый агент пытается достичь своих целей, которые неизвестны другим агентам. Одной из стратегий решения проблем в многоагентной среде является независимое Q -обучение, при котором между агентами нет связи и другие агенты считаются частью среды [63]. Однако, когда агент меняет свою политику, это влияет на политику других агентов, что приводит к нестабильности обучения [64].

В [3, 65, 66] для обеспечения коммуникации между агентами предлагается, чтобы состояние для каждого агента включало информацию о состоянии N -ближайших агентов. Пространство состояний для агента имеет постоянный размер, поскольку оно зависит только от N ближайших агентов и не масштабируется с увеличением числа агентов в среде. При этом отмечается, что определение того, какие N ближайших агентов следует учитывать, очень важно для получения хорошего результата, поскольку добавление нерелевантной информации в пространство состояний усложняет обучение [3].

В [67] для решения проблемы самоподдержания безопасного разделения ВС при высоких потоках движения в секторе представлен алгоритм обучения с подкреплением в сочетании с поиском по дереву Монте-Карло (MCTS-UCT). Все агенты (ВС) находятся на одном и том же эшелоне полета, стратегии действий включают изменения курса и крейсерской скорости.

Значение вознаграждения для агента в зависимости от состояния s определяется следующим образом:

$$r(s) = \begin{cases} 1, & \text{если } s \text{ – целевое состояние,} \\ 0, & \text{если } s \text{ – } LOS \text{ или вне границы сектора,} \\ 1 - \frac{d(o, g)}{\max d(o, g)} & \text{в противном случае,} \end{cases}$$

где состояние LOS – потеря безопасного разделения, $d(o, g)$ – расстояние между текущим положением агента и его целью, $\max d(o, g)$ – наибольшее расстояние между агентом и его целью.

Каждое состояние агента рассматривается как узел в дереве, развертывание по дереву выполняется на основе значений состояния, рассчитываемых по формуле

$$UCT(S_j) = \bar{r}_j + 2C \sqrt{\frac{2 \ln N}{n_j}},$$

где \bar{r}_j – среднее значение вознаграждения действия j для текущего агента, N – счетчик посещений узла, n_j – счетчик выбора действия j , $C = 1/\sqrt{2}$.

В процессе совместного принятия решений все n агентов $\{A_1, \dots, A_n\}$ должны делиться своим намерением при выборе каждого отдельного действия. Одна итерация алгоритма мультиагентного марковского процесса принятия решений заключается в следующем. Сначала n агентов $\{A_1, \dots, A_n\}$ инициализируются на уровне $L - 1$. Все агенты продолжают выполнять политику совместных действий по умолчанию $a_{-j} = \{a_i \text{ из политики действий по умолчанию } | i = 1, \dots, n, i \neq j\}$. Агент A_j с минимальным индексом на уровне $L - 1$ при помощи алгоритма MCTS-UCT выбирает свою оптимальную стратегию действий a_j^* в алгоритме, используя следующее уравнение:

$$a_j^* = \underset{a_j}{\operatorname{argmax}} r_j^*(s, a_j, a_{-j}), \quad j = 1, \dots, n,$$

где $r_j^*(s, a_j, a_{-j})$ – это значение вознаграждения агента A_j в состоянии s при выполнении действия a_j , в то время как стратегии действий других агентов представлены как a_{-j} . При вычислении a_j^* другие агенты продолжают следовать стратегии действия по умолчанию, установленной как $a_{-j} = \{a_i \text{ из политики действий по умолчанию } | i = 1, \dots, n, i \neq j\}$. Когда агент A_j получает свою оптимальную стратегию действий a_j^* , A_j повышается до уровня L и сохраняет стратегию действия a_j^* для обновления стратегии совместного принятия решений по умолчанию. Затем следующий агент, с минимальным индексом на уровне $L - 1$, выбирает свою оптимальную стратегию действий. Этот процесс будет продолжаться до тех пор, пока все агенты получают свою оптимальную стратегию действий $\{a_i^*, i = 1, \dots, n\}$. Полученные стратегии совместного принятия решений используются для следующего временного шага Δt для всех агентов. Итерации повторяются до тех пор, пока все агенты достигнут целевого состояния.

В [68] предложен метод глубокого ансамблевого многоагентного обучения с подкреплением для динамической корректировки скорости ВС в режиме реального времени. Авторы утверждают, что обширные эмпирические результаты, полученные с помощью модели управления воздушным движением с открытым исходным кодом, разработанной Eurocontrol и основанной на реальных данных, включающих тысячи ВС, демонстрируют, что предложенный метод значительно превосходит другие эталонные подходы.

В [69] предлагается подход к многоагентному обучению с подкреплением для трехмерного разрешения конфликтов в свободном маршрутном пространстве, в котором агенты используют общую нейронную сеть. Обученная сеть развертывается на каждом самолете для формирования распределенной системы принятия решений в реальном времени. При этом коммуникация между агентами сводится к информированию других агентов о выбранных действиях. Введение трехмерного пространства приводит к взрывному росту масштаба нейронной сети и, как следствие, к увеличению сложности обучения. Для преодоления этой проблемы предлагается для трехмерных маневров вместо трехмерного пространства рассматривать конфликты в трех плоскостях: плоскости текущего эшелона полета и двух смежных плоскостях на уровнях выше и ниже. Это значительно снижает сложность обучения агента. Предлагается модель буфера конфликта, в которой каждому ВС назначается защищенная зона и зона предупреждения. Рисунок 4 [69] иллюстрирует модель буфера конфликта, где R^W и R^P обозначают радиусы зоны предупреждения и защищенной зоны соответственно. Нарушители, обнаруженные внутри защищенной зоны, всегда получают большой штраф, нарушители, обнаруженные в зоне предупреждения, получают небольшой штраф.

Количество конфликтующих ВС, за которыми может наблюдать агент, фиксировано в структуре нейронной сети и не должно изменяться. Для решения этой проблемы разработана модель частичного наблюдения, при которой рассматривается только фиксированное количество ВС, представляющих наибольшую угрозу, например наиболее близких к агенту. Для реализации обучения с подкреплением используется алгоритм DQN с улучшениями, называемый Rainbow [70]. Для обучения и оценки предлагаемого подхода создана среда моделирования, учитывающая неопределенность полета (возникающую, например, в результате механических и навигационных ошибок и ветра). Экспериментальные результаты показывают [69], что предложенный метод может разрешать конфликты в сценариях с гораздо более высокой плотностью трафика, чем в сегодняшних реальных ситуациях. Сравниваются 2D и 3D модели. Время обучения 2D модели составляет менее 1% от времени обучения 3D модели, однако это не влияет на производительность модели во время решения из-за природы методов обучения с подкреплением. В некоторых сценариях 3D модель может разрешать конфликты легче, изменяя эшелон полета. Показано, что 3D-модель превосходит 2D-модель по показателю успешности и показателю снижения дополнительной дальности полета.

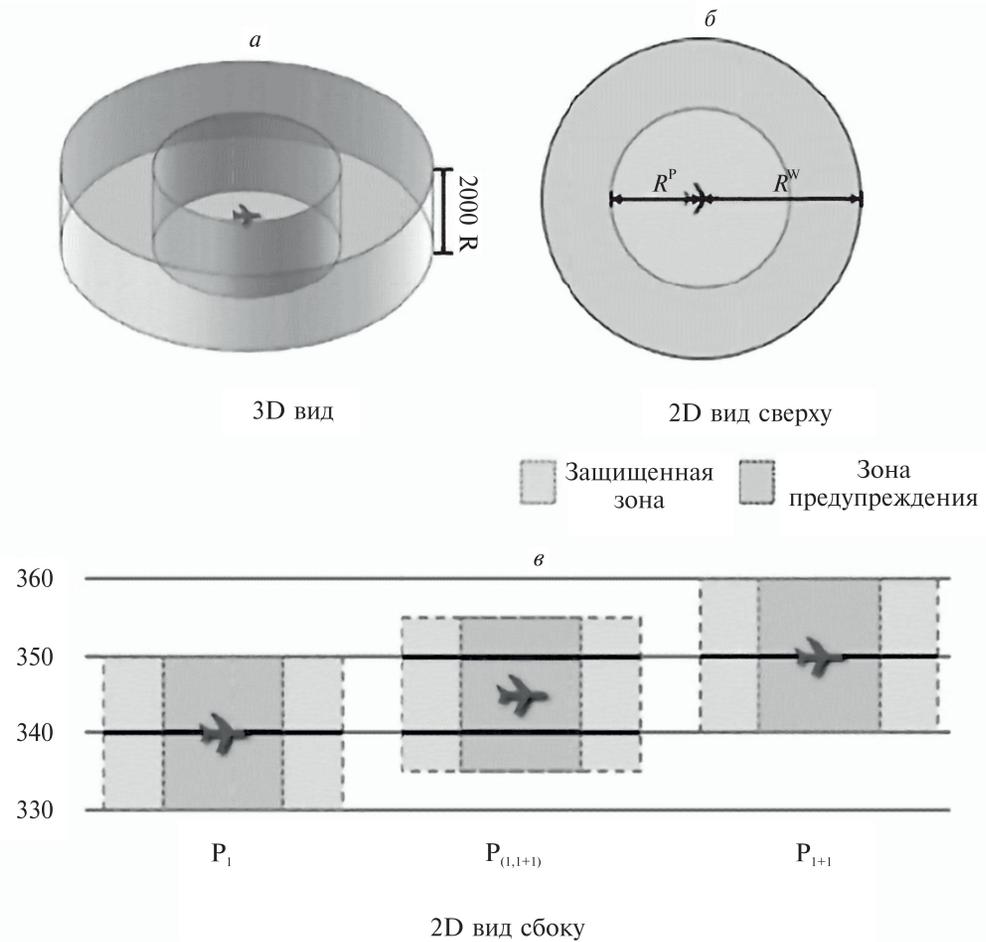


Рис. 4. Модель буфера конфликта.

При централизованном управлении воздушным движением, когда авиадиспетчер передает директивы пилотам, действия должны быть достаточно редкими. Однако в моделях с непрерывным пространством действий агенты могут вносить небольшие корректировки своих траекторий на каждом шаге. В [71] предлагается многоагентный метод глубокого обучения с подкреплением с непрерывным пространством действий, при котором количество действий значительно сокращается с помощью механизма приоритетов. На каждом временном шаге максимум один самолет с наивысшим приоритетом может выполнять действия. Такой подход значительно сокращает количество предпринимаемых действий, сохраняя при этом высокий уровень производительности предупреждения конфликтов. Получаемые решения подходят для централизованного управления воздушным движением, где количество директив, которые могут быть переданы пилоту, ограничено. В [72] механизм приоритетов на основе динамической оценки близости конфликтов между ВС используется в модели с дискретным пространством действий.

3.3. Модели предупреждения конфликтов между произвольным количеством воздушных судов

Выше рассмотрены модели, в которых агент имеет доступ к информации о состоянии N ближайших ВС, где N – гиперпараметр, выбираемый в процессе экспериментов, однако это ограничивает переносимость модели. Использование параметра N является недостатком, поскольку небольшое изменение местоположения самолета может привести к изменению множества N ближайших самолетов и, таким образом, к изменению входных данных для нейронной сети. Нейронная сеть должна понимать, что несмотря на перестановки ВС, это почти одно и то же состояние воздушного пространства, но это может быть сложной задачей [49]. Решение этой проблемы заключается в использовании более продвинутых архитектур нейронных сетей, поиске других способов представления данных, использовании алгоритмов улучшения данных, выборе наиболее релевантных соседних ВС [71]. Одним из способов решения проблемы переменного числа ВС является графическое кодирование информации в изображения фиксированного размера и использование сверточных нейронных сетей (CNN) для извлечения полезной информации аналогично экранам авиадиспетчеров [73]. Другая возможность заключается в использовании рекуррентных нейронных сетей (RNN) с ячейками долгой краткосрочной памяти (LSTM) [74] или управляемыми рекуррентными блоками (GRU) [75], которые работают со всем набором самолетов в среде, кодируя соответствующую информацию в скрытое состояние фиксированного размера.

В [30, 76] рассматривается многоагентное обучение с подкреплением для разрешения конфликтов на маршрутах и пересечениях в структурированном двумерном воздушном пространстве между переменным количеством ВС. Информация о состоянии кодируется при помощи нейронной сети LSTM в вектор фиксированной длины. Агент имеет доступ к закодированной информации обо всех ВС в секторе, в этом случае нет необходимости определять значение N для каждой новой среды. В качестве среды для обучения используется симулятор управления воздушным движением BlueSky [77]. Используется централизованное обучение и децентрализованная схема выполнения, при которой обучается одна нейронная сеть. Эта сеть используется всеми агентами для получения рекомендаций по скорости, при этом в зависимости от состояния действия агентов могут быть разными. Среда является стохастической, поскольку существует неопределенность в действиях других агентов, поэтому используется алгоритм «актер–критик» с названием «проксимальная оптимизация политики» [78].

Для обеспечения требований безопасного разделения вводится идентичная функция вознаграждения для всех агентов:

$$r_t = \begin{cases} -1, & \text{если } d_o^c < d^{LOS}, \\ -\alpha + \delta d_o^c, & \text{если } d_o^c < 10 \ \& \ d_o^c \geq d^{LOS}, \\ 0 & \text{в противном случае,} \end{cases}$$

где d^{LOS} – минимальное безопасное расстояние разделения в морских милях ($d^{LOS} = 3$), d_c^c – расстояние от собственного ВС до ближайшего самолета в морских милях, α и δ – маленькие положительные константы для штрафования агентов по мере приближения к потере безопасного расстояния разделения. На трех практических сценариях воздушного движения показана способность решать задачи принятия решений с переменным числом агентов и неопределенностью [30].

Рекуррентные нейронные сети обрабатывают входные данные последовательно, и выходные данные зависят от этой последовательности. Это может привести к нежелательному результату в ситуациях, когда входная последовательность значения не имеет. Трансформеры были введены как альтернатива рекуррентным нейронным сетям для последовательной обработки входных данных, чтобы обеспечить параллельное обучение [79]. Трансформеры вычисляют относительную важность так называемых токенов, содержащих информацию о состояниях ВС, используя механизм внимания. В [80] для наблюдения используются абсолютные состояния – координаты и скорости ВС в системе отсчета, связанной со средой. В [81] используются относительные состояния – координаты и скорости ВС в системе отсчета собственного ВС с положительным направлением оси абсцисс в направлении полета. Однако исследования пока не показали превосходства производительности архитектуры сетей трансформеров по сравнению с нейронными сетями с прямой связью и с рекуррентными нейронными сетями для обеспечения безопасной навигации ВС [81].

4. Кооперативные стратегии предупреждения конфликтов на основе нейронных сетей передачи сообщений

Коммуникация является ключевой способностью кооперативных многоагентных систем, в которых агенты могут получать значительную выгоду от обмена информацией перед выполнением совместных действий [82]. Модель, основанная на нейронных сетях передачи сообщений, позволяющая агентам обмениваться информацией через протокол связи, может позволить агентам разрабатывать кооперативные стратегии совместных действий по предотвращению конфликтов [83].

В [80] самолеты в воздушном пространстве моделируются как агенты кооперативной многоагентной системы. Состояние $s_i = [x_i, y_i, v_i, \chi_i]$ каждого агента $i \in N$ состоит из координат в евклидовом пространстве x_i, y_i , скорости v_i и курса χ_i . Состояние изменяется по формулам:

$$\begin{aligned} x_i(t+1) &= x_i(t) + v_i(t) \sin \chi_i(t) \Delta t, \\ y_i(t+1) &= y_i(t) + v_i(t) \cos \chi_i(t) \Delta t, \\ v_i(t+1) &= v_i(t) + \Delta v_i, \\ \chi_i(t+1) &= \chi_i(t) + \Delta \chi_i, \end{aligned}$$

где Δv_i и $\Delta \chi_i$ – приращения скорости и курса, Δt – шаг моделирования.

Взаимодействие между агентами представлено в виде графа $G = (V, E)$, каждый узел соответствует одному агенту $i \in N$, ребрами e_{ij} соединены агенты, которые могут обмениваться сообщениями. Вектор наблюдения о текущем состоянии i -го агента состоит из пяти элементов:

$$o_i = \begin{bmatrix} d_i/D \\ \cos(\chi_i - \psi_i) \\ \sin(\chi_i - \psi_i) \\ \bar{v}_i \\ \bar{v}_{ei} \end{bmatrix}^T,$$

где d_i – кратчайшее расстояние до точки выхода из зоны наблюдения, D – нормирующий коэффициент, ψ_i – угол пеленга к точке выхода, нормализованная скорость и отклонение скорости определяются следующим образом:

$$\bar{v}_i = \frac{v_i - v_{\min_i}}{v_{\max_i} - v_{\min_i}}; \quad \bar{v}_{ei} = \frac{v_i - v_{opt_i}}{v_{\max_i} - v_{\min_i}}.$$

На каждом шаге агент кодирует свое состояние o_i в скрытое состояние с помощью нейронной сети:

$$h_i^{(0)} = f_h(o_i).$$

Затем начинается фаза передачи сообщений, состоящая из C раундов связи. На каждом раунде $c = 0, 1, \dots, C - 1$ сообщение каждого агента вычисляется как взвешенная сумма ребер, которые соединяют его с соседями, на основе механизма внимания:

$$m_i^{(c+1)} = \sum_{j \in N_i} a_{ij}^{(c+1)} e_{ij}^{(c+1)},$$

где N_i – множество узлов, связанных ребрами с узлом i .

Значения ребер рассчитываются с помощью нейронной сети с учетом скрытых состояний агентов:

$$e_{ij}^{(c+1)} = f_e^{(c)} \left(\left[h_i^{(c)}, h_j^{(c)}, e_{ij}^{(c)} \right] \right).$$

Веса внимания вычисляются по формуле

$$a_{ij}^{(c+1)} = \frac{\exp \left(v_a^{(c)} f_a^{(c)} \left(\left[h_i^{(c)}, h_j^{(c)}, e_{ij}^{(c)} \right] \right) \right)}{\sum_{j \in N_i} \exp \left(v_a^{(c)} f_a^{(c)} \left(\left[h_i^{(c)}, h_j^{(c)}, e_{ij}^{(c)} \right] \right) \right)},$$

где $v_a^{(c)}$ – вектор параметров.

Затем каждый узел обновляет свое состояние с помощью функции обновления:

$$h_i^{(c+1)} = U^{(c)} \left(h_i^{(c)}, m_i^{(c+1)} \right).$$

После C раундов связи между узлами генерируется распределение вероятностей по всем возможным действиям для каждого агента:

$$a_i = f_a \left(\left[h_i^{(0)}, h_i^{(C)} \right] \right).$$

Ожидаемая награда, одинаковая для всех агентов, вычисляется при помощи функции считывания:

$$V^\pi = f_v \left(\sum_{i \in N} f_y \left(\left[h_i^{(0)}, h_i^{(C)} \right] \right) \right),$$

$f_h, f_e^{(c)}, f_a^{(c)}, f_a, f_v, f_y$ – нейронные сети прямого распространения.

На каждом временном шаге каждый агент выбирает действие a_i , после этого среда дает коллективное вознаграждение команде.

В результате экспериментов по обучению представленной модели было показано, что общая награда за эпизод увеличивается, а количество ожидаемых конфликтов уменьшается с ростом числа эпизодов, т.е. агенты могут улучшать свою политику на основе взаимодействия со средой.

5. Использование глубокого обучения с подкреплением на основе графов для предупреждения конфликтов

Во многих исследованиях в многоагентной формулировке векторы состояний нескольких ВС объединяются в многомерный вектор совокупного состояния с помощью операции конкатенации [30, 84]. Однако агрегирование состояний всех ближайших соседних ВС, независимо от наличия потенциальных конфликтов между ними, может привести к обработке избыточных данных и снижению эффективности модели. Такие векторы не могут инкапсулировать пространственно-временную динамику и различать разные уровни риска и срочности в сценариях конфликтов. Обучение с подкреплением на основе графов (Graph RL) предназначено для обработки данных, структурированных в виде графов [85]. Использование присущих графам свойств позволяет улучшить масштабируемость, эффективность и адаптивность при работе с многомерными и динамическими средами [86]. Обучение с подкреплением на основе графов позволяет использовать свойства графов для представления взаимосвязей между самолетами [87–89]. Методы глубокого обучения с подкреплением на основе графов инвариантны к порядку и количеству самолетов.

В [49] предлагается метод глубокого обучения с подкреплением на основе графов при управлении воздушным движением в трехмерном воздушном пространстве. Для предупреждения конфликтов подбираются высота, курс и скорость ВС. Самолеты представляются вершинами графа, узлы в этом графе соединены, если расстояние между парой самолетов ниже определенного порога. Сравняются два подхода: графовые нейронные сети со сверточными слоями (GCN) [90] и графовые нейронные сети с механизмом внимания (GAT – Graph Attention Network) [91], который используется для эффективного агрегирования информации из соседних узлов в графе. При нормальной плотности движения модель с механизмом внимания способна предотвратить 100% потенциальных столкновений и 89,8% потенциальных конфликтов. Однако производительность ухудшается при увеличении плотности движения.

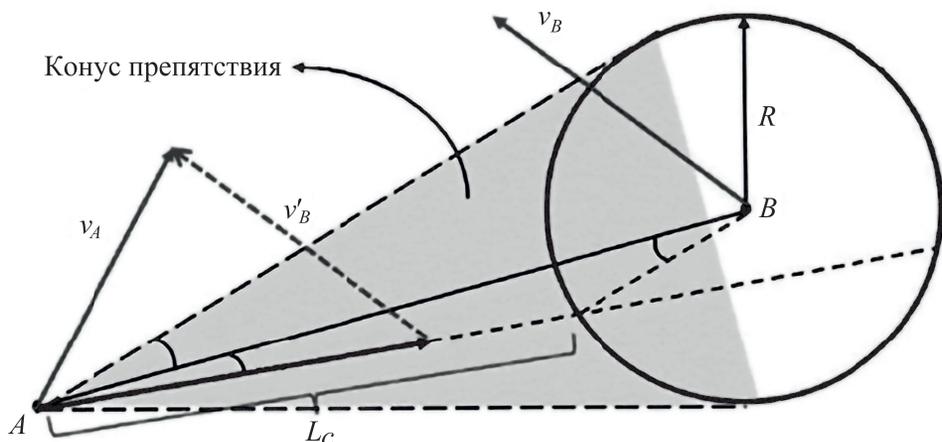


Рис. 5. Геометрия конфликта.

При увеличении плотности потока движения оба метода испытывают трудности с преодолением перегруженного воздушного пространства.

В [92] для сбора пространственно-временных зависимостей полетных данных используется графовая сверточная сеть с ячейками LSTM, а для концентрации на информационных характеристиках ключевых узлов применяется графовая нейронная сеть с повышенным вниманием.

В [93] предложено использовать развивающийся во времени граф конфликтов, в котором самолеты представляются узлами, а связи между ними обозначают срочность конфликта. Срочность конфликта определяется временем до конфликта при сохранении самолетами, находящимися в точках A и B (рис. 5), текущего курса и скорости. Наблюдение ведется из точки A . Точка B является центром окружности с радиусом R (радиус защитной зоны). Из точки A проведены касательные к этой окружности, образующие конус препятствия; v_A и v_B – скорости самолетов A и B . Потенциальный конфликт существует, если вектор относительной скорости $v_r = v_A - v_B$ находится внутри конуса препятствия. Время до конфликта определяется на основе геометрической модели относительного положения и скоростей пары самолетов A и B :

$$t_c = \frac{L_c}{|v_r|},$$

где L_c – отрезок прямой от точки A вдоль относительной скорости v_r до точки пересечения с защитной зоной самолета B . Вес ребра графа конфликта ω_{AB} нормализуется в диапазоне $[0, 1]$ следующим образом:

$$\omega_{AB} = e^{-t_c}.$$

Если потенциального конфликта не существует, то $\omega_{AB} = 0$, если самолеты столкнулись, то $\omega_{AB} = 1$.

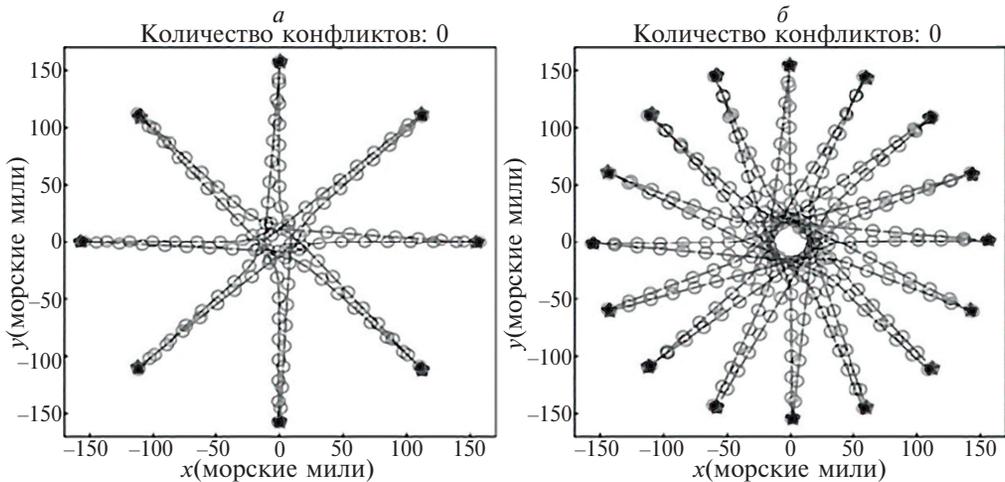


Рис. 6. Сценарий кругового конфликта.

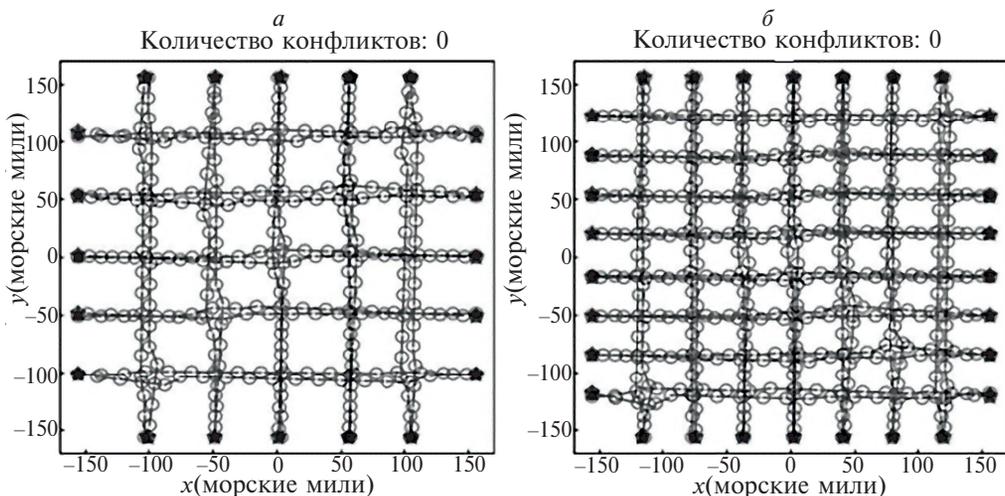


Рис. 7. Сценарий конфликта пересечения.

Затем на основе графа конфликта информация агрегируется при помощи многоголовой нейронной сети внимания. Для повышения стабильности обучения используется механизм временной регуляризации. Эффективность предложенного алгоритма демонстрируется в том числе на двух наглядных сценариях [93].

1) Сценарий кругового конфликта: в этой настройке самолеты начинают полет на точках окружности радиусом 160 морских миль и летят в противоположных направлениях. Такая конфигурация приводит к тому, что каждый самолет конфликтует со всеми остальными в центре окружности. Экспериментальные результаты для сценариев с участием 8 и 16 самолетов показаны на рис. 6, *a* и 6, *б* соответственно. Эти результаты демонстрируют способность

предложенного метода управлять характерными круговыми потенциальными конфликтами и предупреждать все потенциальные столкновения.

2) Сценарий конфликта пересечения: самолеты делятся на две группы, каждая из которых содержит равное количество самолетов. Эти группы летят по пересекающимся траекториям, вызывая конфликты на каждом пересечении. Проведено тестирование с 20 и 30 самолетами. Как видно на рис. 7,а и 7,б, для 20 и 30 самолетов соответственно, с помощью предложенного метода определяются бесконфликтные точки маршрутов на каждом временном шаге с минимальными отклонениями от исходных траекторий.

Несмотря на значительные успехи исследований, имеются серьезные препятствия для практического применения методов обучения с подкреплением в сфере воздушного транспорта из-за строгих стандартов сертификации в авиационной отрасли. Существующая нормативно-правовая база не обеспечивает адекватных и приемлемых средств соответствия требованиям для приложений обучения с подкреплением, и, таким образом, пока не существует правовой базы для их безопасного использования. Необходимо разработать рекомендации по сертификации моделей обучения с подкреплением, предназначенных для воздушного транспорта, чтобы эти многообещающие методы стало возможно использовать в реальных условиях [94].

6. Заключение

Из обзора литературы следует, что методы предупреждения конфликтов в воздушном движении на основе глубокого обучения с подкреплением направлены на решение двух основных задач: автоматическая генерация вариантов решений для поддержки авиадиспетчеров при централизованном управлении воздушным движением и поддержка систем автономного разрешения конфликтов в свободном полете. Для поддержки принятия решений авиадиспетчерами предлагаются, главным образом, модели с дискретными действиями. Модели с непрерывными действиями предназначены для автономного разрешения конфликтов при свободном полете и позволяют всем агентам выполнять действия по корректировке траекторий на каждом временном шаге.

Наиболее перспективным представляется подход обучения с подкреплением на основе графов, поскольку представление информации в виде развивающегося во времени графа конфликта позволяет сократить объем обрабатываемой избыточной информации и обеспечивает масштабируемость моделей на различное количество ВС. Механизм внимания позволяет выделить наиболее срочную информацию, содержащуюся в графе конфликта, обеспечивая улучшенные с точки зрения безопасности и эффективности стратегии предупреждения конфликтов.

По мере развития исследований потенциал практического применения методов предупреждения конфликтов между ВС на основе глубокого обучения с подкреплением становится все более очевидным. Обзор литературы свидетельствует, что исследуемые методы обучения с подкреплением показывают

многообещающие результаты за счет адаптивного принятия решений в режиме реального времени для предупреждения конфликтов в воздушном движении. Однако для их применения на практике управления воздушным движением, где критически важна безопасность, еще существуют серьезные нерешенные проблемы. Обучать модели в реальных условиях нельзя из-за потенциального ущерба, при этом идеально имитировать реальность невозможно. Сдвиг распределения между смоделированной средой и реальностью может ограничивать эффективность моделей обучения с подкреплением. Сертификация таких моделей должна стать одним из направлений исследований в этой области.

СПИСОК ЛИТЕРАТУРЫ

1. *International Civil Aviation Association*. Doc 4444: Air Traffic Management // Procedures for Air Navigation Services, 16th ed. ICAO: Montreal, QC, Canada. 2016.
2. *Кулида Е.Л., Лебедев В.Г.* Методы решения задач планирования и регулирования потоков воздушного движения. Ч. 1 // Проблемы управления. 2023. № 1. С. 3–14. <https://doi.org/10.25728/pu.2023.1.1>
Kulida E.L., Lebedev V.G. Methods for Solving Some Problems of Air Traffic Planning and Regulation. PART I: Strategic Planning of 4D Trajectories // Control Sciences. 2023. No. 1. P. 2–11. <https://doi.org/10.25728/cs.2023.1.1>
3. *Brittain M., Wei P.* Autonomous Separation Assurance in an High-Density En Route Sector: a Deep Multi-Agent Reinforcement Learning Approach // The 22nd IEEE Intelligent Transportation Systems Conference (ITSC). Auckland, New Zealand, 2019. <https://doi.org/10.109/ITSC.2019.8917217>
4. *Пономарев К.Ю.* Метод оценки динамической воздушной обстановки на конфликтность посредством полихромного отображения объектов в информационном обеспечении диспетчера управления воздушным движением // Автореф. дис. канд. техн. наук. СПб.: ФГБОУ ВО СПбГУ ГА им. А.А. Новикова, 2023. 24 с.
5. *Erzberger H.* Automated Conflict Resolution for Air Traffic Control // 25th International Congress of the Aeronautical Sciences (ICAS). Germany, Hamburg. 2006.
6. *Farley T., Field M., Erzberger H.* Fast-time Simulation Evaluation of a Conflict Resolution Algorithm Under High Air Traffic Demand. 2007. <https://www.researchgate.net/publication/255062615>
7. *Erzberger H., Heere K.* 2010. Algorithm and Operational Concept for Resolving Short-range Conflicts // Proc. Inst. Mechan. Engin., Part G: J. Aerospac. Engin. 2010. V. 224. No. 2. P. 225–243. <https://doi.org/10.1243/09544100JAERO546>
8. *Hoekstra J.M., van Gent R.N.H.W., Ruigrok R.C.J.* Designing for Safety: The ‘free flight’ Air Traffic Management Concept // Reliab. Engin. Syst. Safet. 2002. V. 75. No. 2. P. 215–232. [https://doi.org/10.1016/S0951-8320\(01\)00096-5](https://doi.org/10.1016/S0951-8320(01)00096-5)
9. *Clari M.V., Ruigrok R.C.J., Hoekstra J.M., Visser H.G.* Cost-benefit Study of Free Flight with Airborne Separation Assurance // Air Traffic Control Quarterly. 2001. V. 9. No. 4. P. 287–309. <https://doi.org/10.2514/atcq.9.4.287>
10. *Марьин Н.П.* Перспектива внедрения концепции «свободный полет» // Проблемы безопасности полетов. 2009. № 5. С. 42–55.

11. *Yang Y., Zhang J., Cai K., Prandini M.* Multi-aircraft Conflict Detection and Resolution Based on Probabilistic Reach Sets // *IEEE Transactions on Control Systems Technology*. 2016. V. 25. No. 1. P. 309–316. <https://doi.org/10.1109/TCST.2016.2542046>
12. *Орлов В.С.* Разработка и исследование алгоритмов обнаружения и предотвращения опасных сближений в воздухе в рамках перспективной системы ОрВД // *Дисс... канд. техн. наук. М. ФГБОУ ВПО «Московский авиационный институт (национальный исследовательский университет)»*. 2015. 116 с.
13. *Буржин В.С.* Синтез алгоритмов обнаружения и разрешения конфликтов-столкновений воздушных судов по информации системы автоматического зависимого наблюдения в условиях неопределенности // *Изв. РАН. Теория и системы управления*. 2017. № 3. С. 157–169. <https://doi.org/10.7868/S0002338817030088>
14. *Кумков С.И., Пятко С.Г.* Быстрые алгоритмы обнаружения конфликтных ситуаций между воздушными судами / «Теория оптимального управления и приложения (ОСТА 2022)». Материалы международной конференции. Ин-т мат. и механики им. Н.Н. Красовского (ИММ УрО РАН). Екатеринбург. 2022. С. 126–131.
15. *Pelegri M., D'Ambrosio C.* Aircraft Deconfliction via Mathematical Programming: Review and Insights // *Transportation Science*. 2022. V. 56. No. 1. P. 118–140. <https://doi.org/10.1287/trsc.2021.1056>
16. *Cafieri S., Conn A.R., Mongeau M.* Mixed-integer Nonlinear and Continuous Optimization Formulations for Aircraft Conflict Avoidance via Heading and Speed Deviations // *Eur. J. Oper. Res.* 2023. V. 310. No. 2. P. 670–679. <https://doi.org/10.1016/j.ejor.2023.03.002>
17. *Dias F., Rey D.* Aircraft Conflict Resolution with Trajectory Recovery Using Mixed-integer Programming // *J. Global Optim.* 2024. V. 90. P. 1031–1067. <https://doi.org/10.1007/s10898-024-01393-1>
18. *Cecen R.K., Cetek C.* Conflict-free En-route Operations with Horizontal Resolution Manoeuvres Using a Heuristic Algorithm // *Aeronaut. J.* 2020. V. 124. P. 767–785. <https://doi.org/10.1017/aer.2020.5>
19. *Eby M.* A Self-organizational Approach for Resolving Air Traffic Conflicts // *Lincoln Lab. J.* 1994. V. 7. No. 2. P. 239–254.
20. *Balasooriyan S.* Multi-aircraft Conflict Resolution Using Velocity Obstacles // *MSc thesis. Delft University of Technology*. 2017. 126 с.
21. *Durand N.* Constant Speed Optimal Reciprocal Collision Avoidance // *Transportation Research. Part C, Emerging Technologies*. 2018. P. 366–379. <https://doi.org/10.1016/j.trc.2018.10.004>
22. *Pan W., Qin L., He Q., Huang Y.* Three-Dimensional Flight Conflict Detection and Resolution Based on Particle Swarm Optimization // *Aerospace*. 2023. V. 10. No. 9. <https://doi.org/10.3390/aerospace10090740>
23. *Sui D., Zhang K.* A Tactical Conflict Detection and Resolution Method for En Route Conflicts in Trajectory-Based Operations // *J. Advanc. Transport.* 2022. No. 2. P. 1–16. <https://doi.org/10.1155/2022/9283143>
24. *Brittain M., Wei P.* Scalable Autonomous Separation Assurance with Heterogeneous Multi-agent Reinforcement Learning // *IEEE Transactions on Automation Science and Engineering*. 2022. V. 19. No. 4. P. 2837–2848. <https://doi.org/10.1109/TASE.2022.3151607>

25. *Самойлов В.А., Дюенко Д.В.* Возможность применения нейронных сетей для поиска и решения потенциальных конфликтных ситуаций между воздушными судами при полетах в верхнем воздушном пространстве // *Транспорт России: проблемы и перспективы – 2022. Материалы Международной научно-практической конференции. Институт проблем транспорта им. Н.С. Соломенко РАН. Санкт-Петербург. 2022. С. 180–184.*
26. *Wang Z., Liang M., Delahaye D.* Data-driven Conflict Detection Enhancement in 3D Airspace with Machine Learning // 2020 International Conference on Artificial Intelligence and Data Analytics for Air Transportation (AIDA-AT). Singapore. 2020. <https://doi.org/10.1109/AIDA-AT48540.2020.904.9180>
27. *Pinto Neto E.C., Baum D., Almeida J.R., et. al.* Deep Learning in Air Traffic Management (ATM): Applications, Opportunities, and Open Challenges // *Aerospace* 2023. V. 10. No. 4. <https://doi.org/10.3390/aerospace10040358>
28. *Razzaghi P., Tabrizian A., Guo W., et. al.* A Survey on Reinforcement Learning in Aviation Applications // *Engineering Applications of Artificial Intelligence*. 2024. V. 136. No. 3. <https://doi.org/10.1016/j.engappai.2024.108911>
29. *Кулида Е.Л., Лебедев В.Г.* Методы решения задач планирования и регулирования потоков воздушного движения Ч. 2. // *Проблемы управления*. 2023. № 2. С. 3–18. <https://doi.org/10.25728/ru.2023.2.2>
Kulida E.L., Lebedev V.G. Methods for Solving Some Problems of Air Traffic Planning and Regulation. PART II: Application of Deep Reinforcement Learning // *Control Sciences*. 2023. No. 2. P. 2–14. <https://doi.org/10.25728/cs.2023.2.2>
30. *Brittain M.W., Wei P.* One to Any: Distributed Conflict Resolution with Deep Multi-agent Reinforcement Learning and Long Short-term Memory // *AIAA Scitech 2021 Forum*. Nashville, Tennessee, USA. <https://doi.org/10.2514/6.2021-1952>
31. *Wang Z., Pan W., Li H., et. al.* Review of Deep Reinforcement Learning Approaches for Conflict Resolution in Air Traffic Control // *Aerospace*. 2022. V. 9. No. 6. <https://doi.org/10.3390/aerospace9060294>
32. *Groot J., Ribeiro M., Ellerbroek J., et. al.* Improving Safety of Vertical Manoeuvres in a Layered Airspace with Deep Reinforcement Learning // *International Conference on Research in Air Transportation (ICRAT)*. Tampa, Florida, USA. 2022. P. 19–23.
33. *Ribeiro M., Ellerbroek J., Hoekstra J.* Review of Conflict Resolution Methods for Manned and Unmanned Aviation // *Aerospace*. 2020. V. 7. No. 6. <https://doi.org/10.3390/aerospace7060079>
34. *Ribeiro M., Ellerbroek J., Hoekstra J.* Distributed Conflict Resolution at High Traffic Densities with Reinforcement Learning // *Aerospace*. 2022. V. 9. No. 9. <https://doi.org/10.3390/aerospace9090472>
35. *Ribeiro M.* Conflict Resolution at High Traffic Densities with Reinforcement Learning // *Thesis for PhD*. 2023. <https://doi.org/10.4233/uuid:a2979919-cb01-41d1-bbba-fefa9079463b>
36. *Ribeiro M., Ellerbroek J., Hoekstra J.* Improving Algorithm Conflict Resolution Manoeuvres with Reinforcement Learning // *Aerospace*. 2022. V. 9. No. 12. <https://doi.org/10.3390/aerospace9120847>
37. *Визильтер Ю.В., Вишняков Б.В., Желтов С.Ю.* Современные технологии искусственного интеллекта и их применение в авиационных комплексах // XVI Всероссийская мультиконференция по проблемам управления (МКПУ–2023). Волгоград. Материалы мультиконференции в 4 т. Т. 3. С. 13–16.

38. *Sui D., Ma C., Wei C.* Tactical Conflict Solver Assisting Air Traffic Controllers Using Deep Reinforcement Learning // *Aerospace*. 2023. V. 10. No. 2. <https://doi.org/10.3390/aerospace10020182>
39. *Bastas A., Vouros G.* Data-Driven Modeling of Air Traffic Controllers' Policy to Resolve Conflicts // *Aerospace*. 2023. V. 10. No. 6. <https://doi.org/10.3390/aerospace10060557>
40. *Кулида Е.Л., Лебедев В.Г.* Проблемы при применении методов машинного обучения в авиации // Труды 16-й Международной конференции «Управление развитием крупномасштабных систем» (MLSD'2023, Москва). М.: ИПУ РАН, 2023. С. 1315–1320. <https://doi.org/10.25728/mlsd.2023.1315>
41. *Degas A., Islam M.R., Hurter C., et. al.* A Survey on Artificial Intelligence (AI) and eXplainable AI in Air Traffic Management: Current Trends and Development with Future Research Trajectory // *Appl. Sci.* 2022. V. 12. Iss. 3. <https://doi.org/10.3390/app12031295>
42. *Wang L., Yang H., Lin Y., et. al.* Enhancing Air Traffic Control: A Transparent Deep Reinforcement Learning Framework for Autonomous Conflict Resolution // *Expert Systems with Applications*. 2024. V. 260(2). <https://doi.org/10.06/j.eswa.2024.125389>
43. *Саттон Р.С., Барто Э.Г.* Обучение с подкреплением. М.: ДМК-Пресс, 2020.
44. *Грессер Л., Кенг В.Л.* Глубокое обучение с подкреплением: теория и практика на языке Python. СПб.: Питер, 2022.
45. *Моралес М.* Грокаем глубокое обучение с подкреплением. СПб.: Питер, 2023.
46. *Sui D., Ma C., Dong J.* Conflict Resolution Strategy Based on Deep Reinforcement Learning for Air Traffic Management // *Aviation*. 2023. V. 27. Iss. 3. P. 177–186. <https://doi.org/10.3846/aviation.2023.19720>
47. *Hasselt H.V.* Double q-learning // 24th Annual Conference on Neural Information Processing Systems. Vancouver, Canada. 2010. P. 2613–2621.
48. *Brittain M., Wei P.* Autonomous Aircraft Sequencing and Separation with Hierarchical Deep Reinforcement Learning // International Conference for Research in Air Transportation. Casteldeifeil, Spain. 2018.
49. *Mollinga J., Hoof H.* An Autonomous Free Airspace En-route Controller Using Deep Reinforcement Learning Techniques // 9th International Conference on Research in Air Transportation (ICRAT). Tampa, Florida, USA. 2020.
50. *Sui D., Xu W., Zhang K.* Study on the Resolution of Multi-aircraft Flight Conflicts Based on an IDQN // *Chin. J. Aeronaut.* 2021. V. 35. No. 11. P. 195–213. <https://doi.org/10.06/j.cja.2021.03.015>
51. *Li S., Egorov M., Kochenderfer M.J.* Optimizing Collision Avoidance in Dense Airspace Using Deep Reinforcement Learning // Proceedings of the 13th USA/Europe Air Traffic Management Research and Development Seminar. Vienna, Austria. 17–21 June 2019. <https://doi.org/10.48550/arXiv.1912.10146>
52. *Hermans M.C.* Towards Explainable Automation for Air Traffic Control Using Deep Q-Learning from Demonstrations and Reward Decomposition. Master's Thesis, Delft University of Technology, Delft, The Netherlands, May 2020.
53. *Pham D.-T., Tran N.P., Goh S.K., et. al.* Reinforcement Learning for Two-aircraft Conflict Resolution in the Presence of Uncertainty // IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF). Singapore. 2019. <https://doi.org/10.1109/RIVF.2019.8713624>

54. Wang Z., Li H., Wang J., et. al. Deep Reinforcement Learning Based Conflict Detection and Resolution in Air Traffic Control // IET Intelligent Transport Systems. 2019. V. 13. Iss. 6. P. 1041–1047. <https://doi.org/10.1049/iet-its.2018.5357>
55. Badea C.A., Groot J., Morfin Veytia A., et. al. Lateral and Vertical Air Traffic Control Under Uncertainty Using Reinforcement Learning // Proceedings of the 12th SESAR Innovation Days. Budapest, Hungary. 2022.
56. Pham D., Tran N., Alam S., et. al. A Machine Learning Approach for Conflict Resolution in Dense Traffic Scenarios with Uncertainties // 13th USA/Europe Air Traffic Management Research and Development Seminar. Vienne, Austria. Jun. 2019.
57. Wen H., Li H., Wang Z. Application of DDPG-based Collision Avoidance Algorithm in Air Traffic Control // 12nd International Symposium on Computational Intelligence and Design. Hangzhou, China. 2019. P. 130–133. <https://doi.org/10.1109/ISCID.2019.00036>
58. Pham D.-T., Tran P.N., Alam S., et. al. Deep Reinforcement Learning Based Path Stretch Vector Resolution in Dense Traffic with Uncertainties // Transportation Research, Part C. 2022. V. 135. No. 3. <https://doi.org/10.1016/j.trc.2021.103463>
59. Mukherjee A., Guleria Y., Alam S. Deep Reinforcement Learning for Air Traffic Conflict Resolution Under Traffic Uncertainties // 41st Digital Avionics Systems Conference (DASC). Portsmouth, USA. 2022. <https://doi.org/10.1109/DASC55683.2022.9925772>
60. Sunil E., Ellerbroek J., Hoekstra J.M. Camda: Capacity Assessment Method for Decentralized Air Traffic Control // International Conference on Research in Air Transportation (ICRAT). Barcelona, Spain. 2018. P. 26–29.
61. Dudoit A., Rimsa V., Bogdevicius M. Investigation of Aircraft Conflict Resolution Trajectories under Uncertainties // Sensors. 2024. V. 24. No. 18. <https://doi.org/10.3390/s24185877>
62. Busoniu L., Babuska R., De Schutter B. A Comprehensive Survey of Multi-agent Reinforcement Learning // IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews). 2008. V. 38. No. 2. P. 156–172. <https://doi.org/10.1109/TSMCC.2007.913919>
63. Tan M. Multi-agent Reinforcement Learning: Independent vs. Cooperative Agents // 10th International Conference on Machine Learning (ICML). 1993. P. 330–337. <https://doi.org/10.1016/B978-1-55860-307-3.50049-6>
64. Matignon L., Laurent G.J., Le Fort-Piat N. Independent Reinforcement Learners in Cooperative Markov Games: a Survey Regarding Coordination Problems // Knowledge Engineering Review. 2012. V. 27. No. 1. <https://doi.org/10.1017/S0269888912000057>
65. Everett M., Chen Y.F., How J.P. Motion Planning Among Dynamic, Decision-making Agents with Deep Reinforcement Learning // IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2018. P. 3052–3059. <https://doi.org/10.48550/arXiv.1805.01956>
66. Chen. Y., Hu M., Yang L., et. al. General Multi-agent Reinforcement Learning Integrating Adaptive Manoeuvre Strategy for Real-time Multi-aircraft Conflict Resolution // Transportation Research. Part C. Emerging Technologies. 2023. V. 151. <https://doi.org/10.1016/j.trc.2023.104125>

67. *Xu Q., Chen Z., Li F., et. al.* An Efficient Aircraft Conflict Detection and Resolution Method Based on an Improved Reinforcement Learning Framework // Int. J. Aerospac. Engin. 2023. V. 1. P. 1–16. <https://doi.org/10.1155/2023/6643903>
68. *Ghosh S., Laguna S., Lim S.H., et. al.* A Deep Ensemble Method for Multi-agent Reinforcement Learning: A Case Study on Air Traffic Control // 31st International Conference on Automated Planning and Scheduling. SuiGuangzhou, China. 2021. P. 468–476. <https://doi.org/10.1609/icaps.v31i1.15993>
69. *Chen Y., Xu Y., Yang L., et. al.* General real-time three-dimensional multi-aircraft conflict resolution method using multi-agent reinforcement learning // Transportation Research. Part C. Emerging Technologies. V. 157. <https://doi.org/10.1016/j.trc.2023.104367>
70. *Hessel M., Modayil J., Van Hasselt H., et. al.* Rainbow: Combining Improvements in Deep Reinforcement Learning // Thirty-Second AAAI Conference on Artificial Intelligence. 2018. V. 32. No. 3. <https://doi.org/10.1609/aaai.v32i1.11796>
71. *Nilsson J., Unger J., Eilertsen G.* Self-Prioritizing Multi-Agent Reinforcement Learning for Conflict Resolution in Air Traffic Control with Limited Instructions // Aerospace. 2025. V. 12. No. 2. <https://doi.org/10.3390/aerospace12020088>
72. *Sui D., Zhou Z., Cui X.* Priority-based Intelligent Resolution Method of Multi-aircraft Flight Conflicts // Aeronaut. J. 2024. V. 129. P. 326–350. <https://doi.org/10.1017/aer.2024.75>
73. *Zhao P., Liu Y.* Physics Informed Deep Reinforcement Learning for Aircraft Conflict Resolution // IEEE Transactions on Intelligent Transportation Systems. 2021. V. 23. No. 7. P. 8288–8301.
74. *Hochreiter S., Schmidhuber J.* Long Short-Term Memory // Neural Computation. 1997. V. 9. No. 8. P. 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
75. *Cho K., Merriënboer B., Gulcehre C., et. al.* Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation // Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar. 2014. P. 1724–1734. <https://doi.org/10.3115/v1/D14-1179>
76. *Brittain M.W., Yang X., Wei P.* Autonomous Separation Assurance with Deep Multi-agent Reinforcement Learning // J. Aerospac. Inform. Syst. 2021. V. 18. No. 12. P. 890–905. <https://doi.org/10.254/1.1010973>
77. *Hoekstra J.M., Ellerbroek J.* BlueSky ATC Simulator Project: an Open Data and Open Source Approach // 7th International Conference on Research in Air Transportation. Philadelphia, USA. 2016. V. 131. P. 132.
78. *Schulman J., Wolski F., Dhariwal P., et. al.* Proximal Policy Optimization Algorithms // ArXiv. 2017. <https://doi.org/10.48550/arXiv.1707.06347>
79. *Vaswani A., Shazeer N., Parmar N., et. al.* Attention is All You Need // 31st Conference on Neural Information Processing Systems (NIPS). Long Beach, USA. 2017. <https://doi.org/10.48550/arXiv.1706.03762>
80. *Dalmau R., Allard E.* Air Traffic Control Using Message Passing Neural Networks and Multi-agent Reinforcement Learning // Conference: SESAR Innovation Days (SID), Virtual Event. 2020. P. 7–10.
81. *Groot J., Ellerbroek J., Hoekstra J.* Using Relative State Transformer Models for Multi-Agent Reinforcement Learning in Air Traffic Control // Conference: SESAR Innovation days (SID). Seville, Spain. 2023.

82. *Wollkind S., Valasek J., Ioerger T.* Automated Conflict Resolution for Air Traffic Management Using Cooperative Multi-agent Negotiation // AIAA Guidance, Navigation, and Control Conference and Exhibit. 2004. <https://doi.org/10.2514/6.2004-4992>
83. *Pritchett R., Genton A.* Negotiated Decentralized Aircraft Conflict Resolution // IEEE Transactions on Intelligent Transportation Systems. 2017. V. 19. No. 1. P. 81–91. <https://doi.org/10.1109/TITS.2017.2693820>
84. *Lai J., Cai K., Liu Z., et. al.* A Multi-agent Reinforcement Learning Approach for Conflict Resolution in Dense Traffic Scenarios // IEEE/AIAA 40th Digital Avionics Systems Conference (DASC). San Antonio, USA. 2021. <https://doi.org/10.1109/DASC52595.2021.9594437>
85. *Wu Z., Pan S., Chen F., et. al.* A comprehensive Survey on Graph Neural Networks // 2019. <https://doi.org/10.48550/arXiv.1901.00596>
86. *Mendonca M., Ziviani A., Barreto A.* Graph-Based Skill Acquisition for Reinforcement Learning // ACM Computing Surveys (CSUR). 2019. V. 52. No. 1. <https://doi.org/10.1145/3291045>
87. *Papadopoulos G., Bastas A., Vouros G.A., et. al.* Deep Reinforcement Learning in Service of Air Traffic Controllers to Resolve Tactical Conflicts // Expert Systems with Applications. 2024. V. 236. No. 1. <https://doi.org/10.1016/j.eswa.2023.121234>
88. *Isufaj R., Aranega Sebastia D., Piera M.A.* Toward Conflict Resolution with Deep Multi-agent Reinforcement Learning // J. Air Transport. 2022. V. 30. No. 3. P. 71–80. <https://doi.org/10.2514/1.DO296>
89. *Vouros G., Papadopoulos G., Bastas A., et. al.* Automating the Resolution of Flight Conflicts: Deep Reinforcement Learning in Service of Air Traffic Controllers // PAIS 2022. P. 72–85. <https://doi.org/10.48550/arXiv.2206.07403>
90. *Kipf T.N., Welling M.* Semi-supervised Classification with Graph Convolutional Networks // 2017. <https://doi.org/10.48550/arXiv.1609.02907>
91. *Velickovic P., Cucurull G., Casanova A., et. al.* Graph Attention Networks // 2018. <https://doi.org/10.48.550/arXiv.170.0903>
92. *Zhang Y., Xu S., Zhang L., et. al.* Short-term Multi-step-ahead Sector-Based Traffic Flow Prediction Based on the Attention-enhanced Graph Convolutional LSTM Network (AGC-LSTM) // Neural Computing and Applications. 2024. <https://doi.org/10.1007/s00521-024-09827-3>
93. *Li Y., Zhang Y., Guo T., et. al.* Graph Reinforcement Learning for Multi-Aircraft Conflict Resolution // IEEE Transactions on Intelligent Vehicles. 2024. <https://doi.org/10.1109/TIV.2024.3364652>
94. *Ribeiro M., Tseremoglou I., Santos B.* Certification of Reinforcement Learning Applications for Air Transport Operations Based on Criticality and Autonomy // AIAA Science and Technology Forum and Exposition. Orlando, Florida, USA. 2024. <https://doi.org/10.2514/6.2024-1463>

Статья представлена к публикации членом редколлегии Н.И. Сельвесюком.

Поступила в редакцию 04.03.2025

После доработки 17.03.2025

Принята к публикации 25.03.2025