РЕШЕНИЕ ЗАДАЧИ ОПТИМИЗАЦИИ ПОСЛЕДОВАТЕЛЬНОСТИ И ВРЕМЕН ПОСАДОК ВОЗДУШНЫХ СУДОВ ПРИ ПОМОЩИ ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ

Соседов В.А., Макаревич М.В., Кулида Е.Л.
Институт проблем управления им. В.А. Трапезникова РАН, Москва, Россия
vladyslav.sosedov@gmail.com, makarevich.matv@mail.ru, elena-kulida@yandex.ru

Аннотация. В докладе рассмотрена задача оптимизации последовательности и времен посадок воздушных судов и предложен новый подход к решению на основе обучения с подкреплением. Предложенный подход позволит получать решения для системы организации прилетов воздушных судов в аэропорты назначения в режиме реального времени.

Ключевые слова: организация воздушного движения, последовательность посадок воздушных судов, обучение с подкреплением, глубокие Q-сети.

Введение

Задача оптимизации последовательности и времен посадок воздушных судов (Aircraft Landing Problem, ALP) в течении многих лет изучается в странах с интенсивным воздушным движением, т.к. является важной составной частью автоматизированных систем организации потоков прибывающих воздушных судов (ВС) AMAN (Arrival Management), предназначенных для эффективной и безопасной организации воздушного движения в районах наиболее перегруженных аэроузлов. К сожалению, как показывает обзор литературы, в РФ этой задаче уделяется недостаточное внимание, и в настоящее время управление ВС в районах аэропортов в значительной степени базируется на принимаемых авиадиспетчерами решениях вопреки тому, что автоматизация этого процесса может повысить безопасность и эффективность организации воздушного движения вблизи крупных аэропортов [1, 2].

Несмотря на то, что было предложено большое число различных подходов к решению задачи, в том числе точных и приближенных на основе эвристических [3] и генетических алгоритмов [4], поиск новых решений актуален по сегодняшний день, поскольку применение предложенных подходов затруднительно вследствие недостаточно быстрых, для организации воздушного движения в районах аэропортов в режиме реального времени, решений. В частности, в работе [5] предлагается новая методология планирования посадки самолетов на основе машинного обучения.

В связи с большими достижениями, демонстрируемыми в последние несколько лет алгоритмами обучения с подкреплением в широком спектре задач принятия решений, в том числе в решении частных задач управления воздушным движением, которым посвящен обзор [6], в настоящей работе предлагается использование алгоритма обучения с подкреплением на основе глубоких Q-сетей (Deep Q-Networks, DQN) для решения задачи оптимизации последовательности и времен посадок воздушных судов, что позволит, после предварительного обучения нейронных сетей, получать эффективные решения задачи в режиме реального времени с учетом жестких ограничений по обеспечению безопасности полетов.

1. Математическая постановка задачи

Задача оптимизации последовательности и времен посадок ВС [7] заключается в определении последовательности $Y = \{y_i,\ i = \overline{1,P}\}$ и времен посадок $X = \{x_i,\ i = \overline{1,P}\}$ группы ВС в течение заданного интервала времени $[T_0, T_k]$, доставляющих минимум глобальной целевой функции $F(X)$ при выполнении двух ограничений, диктуемых требованиями обеспечения безопасности полета.

Ограничение 1: для каждого ВС в последовательности на посадку определено временно́е окно, в течении которого оно может совершить безопасную посадку в соответствии с летно-техническими характеристиками и параметрами зоны приземления:

$$E_i \le x_i \le L_i,\ i = \overline{1,P}, \tag{1}$$

где E_i – самое раннее возможное время посадки i-го ВС; L_i – самое позднее возможное время посадки i-го ВС.

Для каждого ВС в последовательности на посадку задан один из типов $C = \{c_1, \dots, c_n\}$, определяющий минимальный временной интервал $S_{C_iC_j}$ между посадкой ВС типа c_j после ВС типа c_i и известна матрица безопасных интервалов между посадками $S = \{S_{C_iC_j},\ c_i, c_j = \overline{c_1, c_n}\}$.

Ограничение 2: для обеспечения безопасной посадки должны выполнятся условия разделения последовательных посадок ВС:

$$x_j \geq x_i + S_{C_i C_j},\ i \neq j,\ x_i > x_j,\ i,j = \overline{1,P}. \tag{2}$$

В настоящей работе в качестве глобальной целевой функции используется нелинейная функция суммы квадратов отклонений фактических времен посадок от известных оптимальных времен посадок $T_i,\ i = \overline{1,P}$ при условии свободной взлетно-посадочной полосы:

$$F(X) = \sum_{i=1}^{P}(T_i - x_i)^2 \tag{3}$$

2. Постановка задачи обучения с подкреплением

Обучение с подкреплением на сегодняшний день является одной из наиболее активно развивающихся областей машинного обучения, представляющая собой вычислительный подход к пониманию и автоматизации обучения и принятия решений, направляемых стремлением к достижению некоторой поставленной цели. Оно отличается от других подходов машинного обучения упором на обучение агента методом проб и ошибок в процессе прямого взаимодействия с окружающей средой, без посредничества учителя и без полной модели среды.

В обучении с подкреплением используется формализм конечных марковских процессов принятия решений (МППР) для описания взаимодействия обучающегося агента со средой в терминах сигналов состояний, действий и вознаграждений. Конечные МППР являются математически идеализированной формой задачи обучения с подкреплением и представляют собой классическую формализацию последовательного принятия решений, когда от действий зависит не только немедленное вознаграждение, но и последующие состояния, а через них и будущие вознаграждения. Так, агент и окружающая его среда взаимодействуют на каждом шаге дискретной последовательности принятия решений (Рис. 1). На каждом шаге t агент получает некоторое представление состояния среды $S_t \in \mathcal{S}$ и, исходя из него, выбирает действие $A_t \in \mathcal{A}(S)$. На следующем шаге агент, отчасти как последствие своего действия, получает числовое вознаграждение $R_{t+1} \in \mathcal{R} \subset \mathbb{R}$ и оказывается в новом состоянии S_{t+1}. Целью агента является максимизация долгосрочных суммарных вознаграждений. Каждый новый эпизод обучения начинается с некоторого начального состояния $S_0 \in \mathcal{S}$ [8].

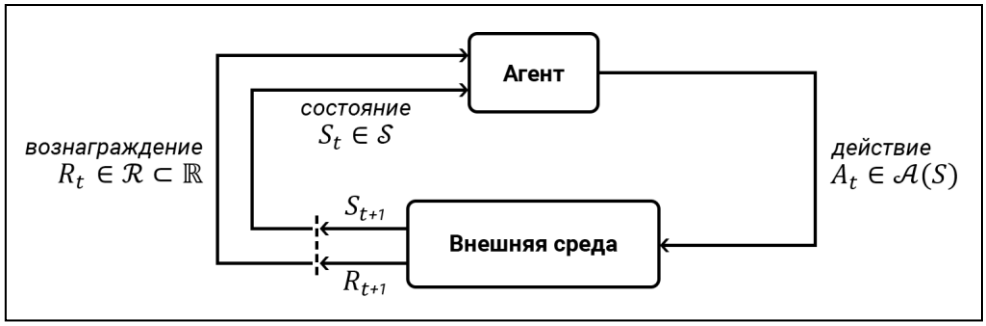

Рис. 1. Процесс взаимодействия агента обучения с подкреплением с окружающей его неопределенной внешней средой

Поставленную задачу оптимизации последовательности и времен посадок ВС предлагается моделировать как МППР с одним обучающимся агентом со следующими соотношениями, определяющими процессы взаимодействия агента и среды:

1. Состояние среды S_t определяется следующим образом:

$$S_t = \{\hat{D}(t),\ \hat{C}(t)\}, \tag{4}$$

где $\hat{D}(t)$ – нормированный вектор отклонений текущих времен посадки ВС x_i от их оптимальных значений T_i; $\hat{C}(t)$ – вектор категориально закодированных типов ВС в последовательности на посадку.

Вектор текущих времен посадок $X(t) = \{x_i,\ i = \overline{1,P}\}$ в подлежащей оптимизации последовательности ВС $Y(t) = \{y_i,\ i = \overline{1,P}\}$, формируется в возрастающем порядке с использованием приближенной формулы:

$$\begin{gathered} x_1 = \max(T_0, E_{y_1}) \\ x_i = \max\left(x_{i-1} + S_{C_{y_i} C_{y_{i-1}}}, E_{y_i}\right),\ i = \overline{2,P} \end{gathered} \tag{5}$$

При расчете $X(t)$ по формуле (5) ограничение (2) всегда будет выполняться, но в ограничении (1) может быть нарушено неравенство справа, связанное с самыми поздними возможными временами посадки ВС. Такое решение будет полагаться недопустимым, величина нарушения определяется по формуле:

$$W(X) = \sum_{i=1}^{P} \max(0, x_i - L_i) \tag{6}$$

Допустимые состояния $S_t \in \mathcal{S}$ характеризуются значением целевой функции (1).

2. Действия агента $A_t \in \mathcal{A}(S)$ представляют собой набор транспозиций соседних ВС в последовательности на посадку $A_k, k = \overline{1, P-1}$ и одного действия завершения текущего эпизода A_P:

$$A_t = \begin{Bmatrix} A_k: & Y_{t+1} = (y_i, y_{i+1})Y_t, \ \ k = \overline{1, P-1}, \\ A_P: & \text{завершение эпизода} \end{Bmatrix} \tag{7}$$

Выбор агентом действия A_P возможен только для допустимых состояний $S_t \in \mathcal{S}$.

3. Немедленное вознаграждение агента $R_t \in \mathcal{R} \subset \mathbb{R}$, генерируемое средой в ответ на действия агента, на каждом шаге последовательности принятия решений определяется следующим образом:

$$R_t = \begin{cases} \frac{F(X_t) - F(X_{t+1})}{F(X_t)}, & \text{если} \ \ W(X) = 0 \\ -K \cdot W(X), & \text{в противном случае} \end{cases} \tag{8}$$

где K – настраиваемый коэффициент пропорциональности штрафа.

4. Максимальное число шагов принятия решений в каждом эпизоде обучения ограничено числом $N = P \cdot (P-1)/2$ шагов, поскольку любой возможный вектор $X(t)$ может быть получен из произвольного вектора $X(0)$, удовлетворяющего условиям (2) и (3), не более чем за N шагов.

3. Алгоритм глубоких *Q*-сетей для решения поставленной задачи

В состоянии S_t агент выбирает действие $A_t \in \mathcal{A}(S)$ в соответствии со стратегией π, целью которой является максимизация долгосрочного вознаграждения. Формально стратегией называется отображение множества воспринимаемых агентом состояний среды на вероятности выбора им каждого действия: если агент следует стратегии π на шаге t, то $\pi(A_t|S_t)$ – вероятность выбора агентом действия A_t в состоянии S_t. Решение задачи обучения с подкреплением означает отыскание оптимальной стратегии, которая дает наибольшее вознаграждение за длительный период времени. Для конечных МППР возможно точно определить оптимальную стратегию.

Функция ценности действия $Q_\pi(S_t, A_t)$ представляет собой ожидаемое конечное суммарное вознаграждение, когда агент начинает работу в состоянии S_t, предпринимает действие A_t и в дальнейшем следует стратегии π:

$$Q_\pi(S_t, A_t) \doteq M_\pi\left[\sum_{k=0}^{\infty} \gamma^k \cdot R_{t+k+1} \,\middle|S_t, \ A_t\right], \ s \in \mathcal{S}, a \in \mathcal{A} \tag{9}$$

где $M_\pi[\cdot]$ – математическое ожидание случайной величины, при условии, что агент следует стратегии π, а t – произвольный временной шаг.

Хотя оптимальных стратегий может быть несколько, все они описываются одной функцией ценности действия, которая называется оптимальной функцией ценности действия:

$$Q_*(S_t, A_t) \doteq \max_\pi Q_\pi(S_t, A_t) \tag{10}$$

Фундаментальное свойство функций ценности действия, состоит в том, что они удовлетворяют уравнению оптимальности Беллмана вида:

$$Q_*(S_t, A_t) = M\left[R_{t+1} + \gamma \cdot \max_{a \in \mathcal{A}(s)} Q_*(S_{t+1}, a) \,\middle|S_t, \ A_t\right] \tag{11}$$

Имея $Q_*(S_t, A_t)$ возможно определить оптимальную стратегию агента: для любого состояния системы агенту достаточно найти действие, доставляющее максимум функции ценности действия. Такой подход позволяет агенту выбирать оптимальные действия, ничего не зная о возможных последующих состояниях и их ценностях, то есть не нуждаясь в полной модели среды. Для конечных МППР уравнение оптимальности Беллмана (11) представляет собой систему из n уравнений, где n – число возможных состояний, и имеет единственное решение.

Хотя явное решение уравнений оптимальности Беллмана и является одним из способов решения задачи обучения с подкреплением, он редко бывает полезен в решении прикладных задач ввиду его вычислительной дороговизны. В задачах обучения с подкреплением, где важна скорость нахождения искомого решения не менее, чем его точность, в том числе и в поставленной задаче оптимизации последовательности и времен посадок ВС, приходится прибегать к приближенным алгоритмам решения уравнений оптимальности Беллмана с использованием фактически обученных переходов вместо точного знания ожидаемых переходов.

В классическом алгоритме Q-обучения методом временных разниц агент динамически формирует табличное представление функции ценности действия на основе получаемого вознаграждения в процессе постепенного изучения среды, что впоследствии дает ему возможность учитывать опыт предыдущего взаимодействия со средой при выборе стратегии поведения. На каждом шаге последовательности принятия решений значения функции ценности действий обновляются как взвешенное среднее между предыдущим и текущим значениями:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \cdot \left[R_{t+1} + \gamma \cdot \max_{a \in \mathcal{A}(s)} Q(S_t, a) - Q(S_t, A_t) \right], \tag{12}$$

где α – скорость обучения; γ – коэффициент дисконтирования.

После обновления функции ценности действия обновляется стратегия агента π. Агент совершает эпсилон-жадный выбор действия: действие A_t, максимизирующее значение функции $Q(S_t, A_t)$ в состоянии S_t, выбирается с вероятностью $1 - \epsilon$, с вероятностью ϵ выбирается случайное действие. Сначала устанавливается максимальное значение $\epsilon = \epsilon_{max}$, затем в процессе обучения ϵ уменьшается до тех пор, пока не достигает минимального значения $\epsilon = \epsilon_{min}$. Таким образом реализуется итерационный процесс, в ходе которого последовательно уточняется функция $Q(S_t, A_t)$ и стратегия π постоянно совершенствуется относительно этой функции [9].

При небольших размерностях пространств состояний $\mathcal{S}$ и действий $\mathcal{A}(S)$ такой подход имеет право на существование, однако в поставленной задаче оптимизации последовательности и времен посадок ВС пространство допустимых состояний оказывается слишком велико для эффективного использования табличного представления функции ценности действий. Кроме того, в классическом алгоритме Q-обучения агент не имеет возможности оценивать значение функции ценности для неизученных состояний, что также оказывается критичным с точки зрения потребного времени на обучение агента и быстродействия его работы и, как следствие, обеспечения требований безопасности полетов. Для решения поставленной задачи был предложен алгоритм глубоких Q-сетей (Deep Q-Networks) [10], представляющий собой метод обучения с подкреплением, направленный на работу в сложных средах больших размерностей и сочетающий в себе классический подход Q-обучения с глубокими нейронными сетями, использующимися для аппроксимации значений функции ценности, в том числе и для неизученных состояний среды. Схема работы алгоритма представлена на (Рис. 2).

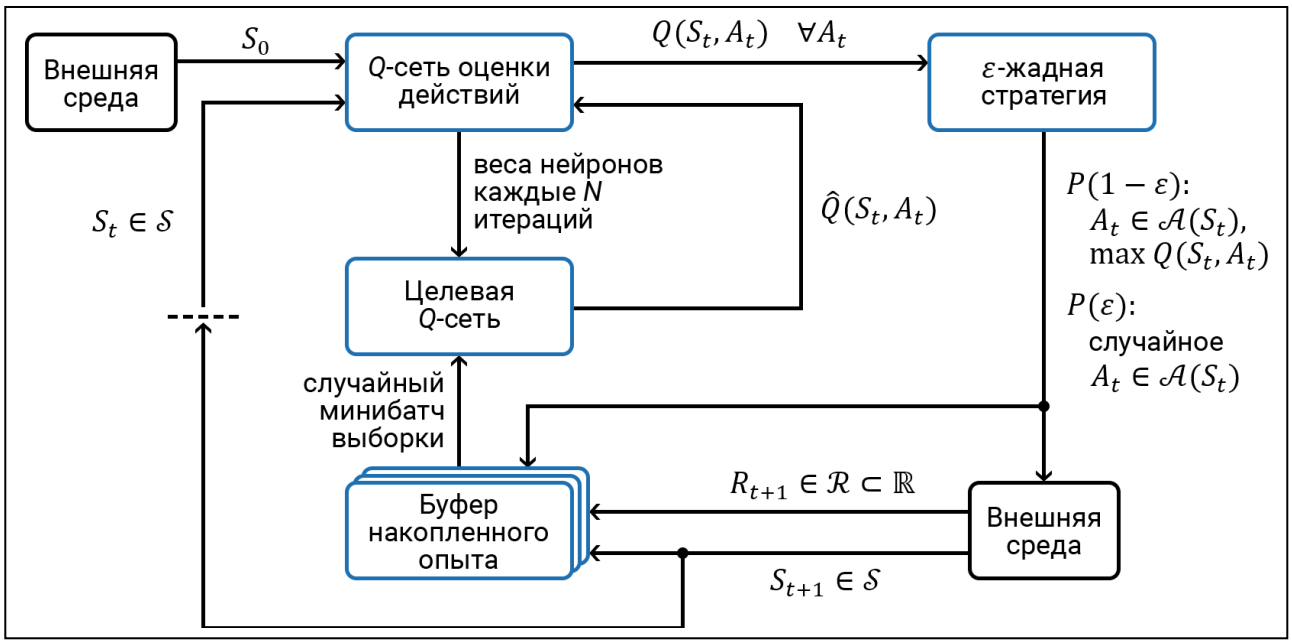

Рис. 2. Схема работы алгоритма глубоких Q-сетей

Для повышения эффективности и стабилизации процесса обучения агента, в алгоритме глубоких Q-сетей используются две отдельные нейронные сети с идентичными архитектурами: сеть оценки действий непосредственно участвует в процессе выбора действий и обучается постоянно, в то время как целевая сеть используется для оценки ожидаемого вознаграждения и обновляется с задержкой.

Для эффективного разрешения главной проблемы использования нейронных сетей – отсутствия гарантий сходимости – используется буфер накопленного опыта, позволяющий уменьшить кросс-корреляцию данных о последовательных взаимодействиях агента и среды и значительно уменьшить вероятность переобучения.

Для обеспечения баланса между исследованием среды на старте обучения и ее последующей эксплуатацией в алгоритме глубоких Q-сетей, аналогично классическому алгоритму Q-обучения используется эпсилон-жадная стратегия выбора действий агентом. Оптимальные значения гиперпараметров алгоритма настраивались методом поиска по сетке.

4. Результаты исследований

Целью настоящей работы является исследование применимости и эффективности предложенного подхода к решению задачи оптимизации последовательности и времен посадок ВС. Для оценки эффективности предлагаемого подхода полученные результаты сравнивались с точными решениями, получаемыми симплекс-методом стандартного пакета CPLEX и с приближенными решениями, получаемыми генетическим алгоритмом [4].

Были проведены эксперименты для последовательностей на посадку малого размера (17 ВС) для верификации работы алгоритма на точных решениях и подбора гиперпараметров реализованного алгоритма глубоких Q-сетей, а также для последовательностей на посадку большого размера (50 ВС) для проведения сравнительного анализа эффективности предлагаемого алгоритма в сравнении с существующим генетическим алгоритмом решения поставленной задачи. На (Рис. 3) представлен пример эволюции конечных суммарных вознаграждений в процессе обучения агента на случайно генерируемых малых последовательностях из 17 ВС.

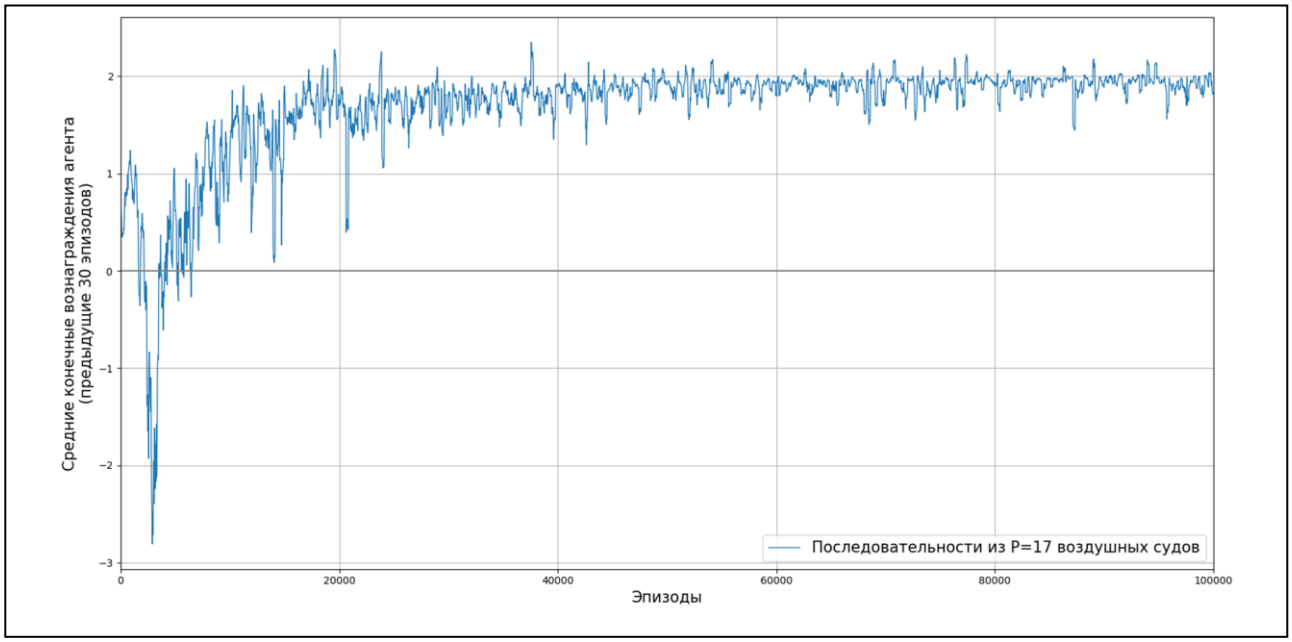

Рис. 3. Пример эволюции конечных суммарных вознаграждений в процессе обучения агента

По итогам исследования была сформирована сравнительная таблица эффективности работы предлагаемого алгоритма глубоких Q-сетей для 15 тестовых последовательностях на посадку (Табл. 1). В качестве метрики точности алгоритмов применялось относительное улучшение (Relative Change) последовательности на посадку:

$$RC = \left(\frac{F(X)-F(\bar{X})}{F(X)}\right) \cdot 100\%, \quad (13)$$

где $F(X)$ – значение глобальной целевой функции для начальной последовательности на посадку; $F(\bar{X})$ – значение глобальной целевой функции для оптимизированной последовательности на посадку.

Таблица 1. Сравнительный анализ эффективности предложенного алгоритма

	Генетический алгоритм		Алгоритм глубокого *Q*-обучения		Симплекс-метод (точное решение)	
Количество воздушных судов	17	50	17	50	17	50
Относительное улучшение	≈49.8%	≈70.1%	≈48.5%	≈51.6%	≈54.0%	–
Время на получение решения, с	0.667	1.189	0.016	0.018	28.417	–

5. Заключение

В настоящей работе был предложен новый подход к решению задачи оптимизации последовательности и времен посадок воздушных судов. Одним из основных преимуществ предлагаемого подхода является скорость получения решения с помощью заранее обученных нейронных сетей, к недостаткам можно отнести необходимость обучения нейронных сетей отдельно для каждой размерности последовательности ВС на посадку.

Обученный на последовательностях малой размерности алгоритм глубоких *Q*-сетей показал сравнимые с существующим генетическим алгоритмом результаты по точности решения и лучшие результаты по скорости его получения. В свою очередь, для последовательностей большой размерности, предложенный алгоритм показал худшие результаты по точности решения, однако хорошие результаты по скорости его получения. Этот факт позволяет использовать предложенный алгоритм для получения начальных приближений решений задач большой размерности с целью значительного ускорения работы генетического алгоритма.

Предполагается продолжение исследований для других постановок задачи оптимизации последовательности и времен посадок ВС:

- для динамического случая, когда набор ВС изменяется в результате посадок некоторых ВС и подлетов новых ВС;
- для случая нескольких взлетно-посадочных полос;
- для использования одной взлетно-посадочной полосы для взлетов и посадок;
- для учета интервалов закрытия взлетно-посадочной полосы.

Литература

1. *Вересников Г.С., Егоров Н.А., Кулида Е.Л., Лебедев В.Г.* Методы построения оптимальных очередей воздушных судов на посадку. Ч. 1. Методы точного решения // Проблемы управления. 2018. № 4. – С. 2–13.
2. *Вересников Г.С., Егоров Н.А., Кулида Е.Л., Лебедев В.Г.* Методы построения оптимальных очередей воздушных судов на посадку. Ч. 2. Методы приближенного решения // Проблемы управления. 2018. № 5. – С. 2–13.
3. *Kulida E.L., Lebedev V.G., Egorov N.A.* The heuristic algorithm for planes queue optimization / Proceedings of the 11th International Conference "Management of Large-Scale System Development" (MLSD). M.: IEEE, 2018. С. https://ieeexplore.ieee.org/document/8551908.
4. *Кулида Е.Л.* Генетический алгоритм решения задачи оптимизации последовательности и времен посадок воздушных судов // Автоматика и телемеханика. 2022. № 3. С. 156-168.
5. *Pang Y., Zhao P., Hu J., Liu Y.* Machine Learning-Enhanced Aircraft Landing Scheduling under Uncertainties // Transportation Research Part C Emerging Technologies. January 2024.
6. *Кулида Е.Л., Лебедев В.Г.* Методы решения задач планирования и регулирования воздушного движения. Ч. 2. Применение методов глубокого обучения с подкреплением // Проблемы управления. – 2023. - № 2. – С. 3-18.
7. *Beasley J.E., Krishnamoorthy M., Sharaiha Y.M., and Abramson D.* Scheduling aircraft landings — the static case // Transportation Science. — 2000. — Vol. 34, N 2. — P. 180—197.

8. *Саттон Р.С., Барто Э.Г.:* Обучение с подкреплением. ДМК-Пресс, 2020 г. 552 с.
9. *Watkins, C. J. Dayan, P.* Q-learning // Machine Learning. – 1992. – Vol. 8. – P. 279–292.
10. *Sewak M.* Deep Q Network (DQN), Double DQN, and Dueling DQN: A Step Towards General Artificial Intelligence // Deep Reinforcement Learning: Frontiers of Artificial Intelligence. – Singapore: Springer Singapore – 2019. – P. 95–108.