

Оптимизация, системный анализ и исследование операций

© 2026 г. Е.А. РЯБУШЕВ (forteamofdota@mail.ru),
А.А. ГАЛЯЕВ, чл.-корр. РАН (galaev@ipu.ru)
(Институт проблем управления им. В.А. Трапезникова РАН, Москва)

ПОСТРОЕНИЕ ПЛАНА ЗАЩИТЫ ПОДВИЖНОЙ ЦЕЛИ С ПОМОЩЬЮ ЗАЩИТНИКА-ПЕРЕХВАТЧИКА ОТ АТАКИ МНОЖЕСТВА АТАКУЮЩИХ МЕТОДОМ ПОИСКА ПО ДЕРЕВУ МОНТЕ-КАРЛО

Рассматривается задача защиты подвижной цели от атаки множества атакующих преследователей с помощью защитника-перехватчика. Приводится строгая математическая постановка задачи, вводятся критерии качества планов защиты, и формулируются математические свойства задачи, на основе которых предлагаются эвристические правила и алгоритмы их применения для построения субоптимальных решений.

Ключевые слова: математическая оптимизация, исследование операций, поиск по дереву Монте-Карло (MCTS), локальные правила.

DOI: 10.7868/S2413977726100056

1. Введение

Мир проходит через большие перемены. Дроны доминируют на суше и в воздухе, а безэкипажные суда и подводные аппараты представляют опасность в море. Автономные комплексы способны решать все больше задач как военных, так и гражданских, без какого-либо участия человека. Это наиболее заметно на поле боя, где беспилотные средства уже изменили характер войны и определяют новые правила. Массированные атаки дронов по объектам инфраструктуры и логистики представляют существенную и актуальную проблему. Становится очевидно, что экономически целесообразная защита от данной угрозы требует перехватчиков, способных последовательно, один за другим, уничтожать атакующие дроны. Значительную роль здесь играет выбор наилучшего порядка перехвата, что является задачей, близкой по смыслу к «динамической задаче коммивояжера» (ДЗК) [1, 2] или Moving Targets TSP (MTTSP) [3, 4]. На данный момент не существует эффективных методов точного решения ДЗК, а сама задача находится на начальных этапах исследования [5]. Современные постановки ДЗК концентрируются на условиях неопределенности в информации о положении целей и ограниченности временных окон для их наблюдения [6, 7].

Задача коммивояжера и, как следствие, многие ее динамические обобщения являются NP-полными [8], а значит, для них нет эффективных алгоритмов точного решения. Построение обороны требует принятия решений в условиях ограниченного времени, для чего требуется разработка методов быстрого субоптимального планирования. Алгоритмы такого планирования уже рассматривались авторами для случая защиты неподвижной точки в [9], цель настоящей работы – их обобщение на задачу защиты подвижной точки. Для этого предлагается обобщенная постановка задачи, которая включает в себя как планирование порядка обхода целей, так и построение маневра уклонения от атакующих защищаемой подвижной точки. Качество планов предлагается оценивать на основе векторных критериев, обобщающих таковые из [10] для защиты статичной точки на динамический случай. Планирование движения подвижного объекта защиты ведется на основе локальных правил, подобных тем, которые применялись в [11], а порядок перехвата атакующих выбирается посредством поиска по дереву Монте-Карло (MCTS) [12, 13].

MCTS – метод комбинаторной оптимизации, получивший известность после успешного применения в проекте AlphaGo [12]. Классические приложения MCTS подразумевали игровые постановки задач в конечных играх, но алгоритм хорошо зарекомендовал себя и для комбинаторной оптимизации в неигровых задачах [14]. Поиск по дереву Монте-Карло с PUCT-оценками (Predictor + Upper Confidence bounds for Treas) обычно применяется в паре с нейросетью и используется для ее обучения. Однако MCTS также способен работать на основе локальных правил, которые могут оценивать действия вместо нейросети [9]. Такая реализация алгоритма не только позволяет применять его без предварительного обучения, но и использовать его для вычисления q -оценок для обучения с подкреплением. Таким образом, адаптация поиска по дереву Монте-Карло не только предоставляет готовый алгоритм решения проблемы планирования, но и открывает путь для эффективного обучения нейросетевых моделей, способных решать данную задачу.

2. Математическая модель, постановка задачи и метод решения

В этом разделе обсуждается постановка задачи защиты подвижной цели защитником-перехватчиком от нападения множества атакующих, которая обобщает постановки задач из [9, 10] на случай подвижного объекта защиты.

2.1. Модель движения защищаемой точки, атакующих целей и защитника-перехватчика

Пусть необходимо защищать подвижную цель T , способную перемещаться со скоростью V_T , от нападения n атакующих, которые гонятся за T со скоростью $V_A > V_T$. Полагаем, что оборонять T требуется защитником-перехватчиком D , который может преследовать атакующих со скоростью $V_D > V_A$. Движение происходит в плоскости, цель T и защитник D в ну-

левой момент времени располагаются в начале координат, а атакующие – на границе круга радиуса R в слое толщиной $2\Delta R$.

Начальные положения атакующих задаются радиус-векторами $\mathbf{r}_1^0, \dots, \mathbf{r}_n^0$, а их скорости $\mathbf{V}_1, \dots, \mathbf{V}_n$ постоянно направлены на T :

$$(1) \quad \mathbf{V}_i = V_A \frac{\mathbf{r}_T - \mathbf{r}_i}{|\mathbf{r}_T - \mathbf{r}_i|},$$

где $\mathbf{r}_i$ и $\mathbf{r}_T$ – текущие положения i -атакующего и цели T . Атакующий считается пропущенным, если в некоторый момент времени его координаты совпадут с положением T .

Динамика защищаемой цели T описывается моделью простых движений

$$(2) \quad \begin{aligned} \dot{x}_T &= V_T \cos(u), \\ \dot{y}_T &= V_T \sin(u), \end{aligned}$$

где x_T и y_T – координаты T , а $u \in [0, 2\pi]$ – управление направлением движения.

Вектор скорости защитника D определяется атакующим, которого D перехватывает в данный момент времени:

$$(3) \quad \mathbf{V}_D = V_D \frac{\mathbf{r}_i - \mathbf{r}_D}{|\mathbf{r}_D - \mathbf{r}_i|},$$

где $\mathbf{r}_D$ и $\mathbf{r}_i$ – положение D и атакующего, на перехват которого выдвигается D . Атакующий считается уничтоженным, если его координаты в некоторый момент времени совпадают с положением D .

2.2. Понятие плана защиты, его критерии качества и постановка задачи

План перехвата $\pi = [\pi_1, \dots, \pi_n]$ – последовательность целей, где π_i – номер цели, которую D посещает в i -очередь. Пусть $\tau^*(\pi)$ – время выполнения π , тогда на $[0, \tau^*(\pi)]$ требуется построить управление $u(t)$ движением T согласно (2). Для упрощения задачи предлагается задавать движение точки T в дискретном времени с шагом Δt . Введем следующее

Определение 1. Планом защиты Π называется пара (π, ρ) , где π – последовательность перехвата целей защитником D , а $\rho = \{\mathbf{r}_T(t_k)\}_{k=0}^K$ – набор точек, которые T занимает в моменты времени $t_k = k\Delta T$. При этом на ρ накладываются ограничения

$$|\mathbf{r}_T(t_{k+1}) - \mathbf{r}_T(t_k)| = V_T \Delta t,$$

которые выражают условия достижимости точек ρ , при этом без потери общности всегда полагаем, что $\mathbf{r}_T(0) = \mathbf{r}_D(0) = (0, 0)$.

Введем вектор потерь $L(\Pi) = (m_0, \dots, m_5)$, где m_0 – число пропущенных целей, а m_i – число перехваченных на расстоянии не менее чем R_i от T , где $R_1 < R_2 < \dots < R_5$. Тогда сформулируем задачу.

Задача 1. По начальным положениям $\mathbf{r}_1^0, \dots, \mathbf{r}_n^0$ атакующих целей построить план защиты Π согласно определению 1 с минимально возможным вектором потерь $L(\Pi)$, где сравнение ведется в словарном порядке, т.е.

$$(m_0, \dots, m_5) < (m'_0, \dots, m'_5) \rightarrow \exists k : \forall i < k : m_i = m'_i \wedge m_k < m'_k.$$

На практике найти оптимальный план Π практически невозможно, и требуется найти субоптимальный Π с приемлемым $L(\Pi)$. В [10] доказана

Теорема 1. Для любого начального расположения атакующих при наличии динамического преимущества ($V_D > V_A$) защитника над атакующими для статичной защищаемой цели существует план перехвата π , который обеспечивает перехват всех атакующих без пропусков.

Следствие 1. Для любого начального расположения атакующих при наличии динамического преимущества ($V_D > V_A$) защитника над атакующими для подвижной защищаемой цели существует план перехвата π , который обеспечивает перехват всех атакующих без пропусков.

Доказательство. Действительно, поскольку динамическая цель способна оставаться на месте неподвижно, то ее способность к перемещению не может ухудшить результат теоремы 1, а значит, всегда существует план защиты без пропусков атакующих.

Таким образом, разумным требованием к субоптимальности плана защиты следует считать $m_0 = 0$, т.е. отсутствие пропусков целей.

2.3. Позиционные свойства атакующих и локальные правила принятия решений

Пусть в момент времени t защитник D и защищаемая цель T занимают позиции $\mathbf{r}_I(t)$ и $\mathbf{r}_T(t)$, и $\mathbf{V}_T(t)$ – скорость T , а $\mathbf{r}_i(t)$ – радиус-векторы атакующих. Для дальнейшего обсуждения потребуются величины

$$\tau_i(t) = \frac{V_A |\mathbf{r}_T(t) - \mathbf{r}_i(t)| + \mathbf{V}_T(t) [\mathbf{r}_T(t) - \mathbf{r}_i(t)]}{V_A^2 - V_T^2},$$

где $\tau_i(t)$ – время, необходимое i -атакующему на поражение T при условии, что T продолжит движение по прямой со скоростью $\mathbf{V}_T$, и

$$(4) \quad \theta_i(t) = \frac{V_I |\mathbf{r}_T(t) - \mathbf{r}_i(t)| + \mathbf{V}_i(t) [\mathbf{r}_T(t) - \mathbf{r}_i(t)]}{V_I^2 - V_A^2},$$

где $\theta_i(t)$ – время, необходимое защитнику D на перехват i -атакующего, при условии что тот будет двигаться по прямой со скоростью $\mathbf{V}_i$, определенной согласно (1) по текущим значениям $\mathbf{r}_i(t)$ и $\mathbf{r}_T(t)$.

На основе $\tau_i(t)$ и $\theta_i(t)$ введем следующие определения.

Определение 2. Под опасностью i -атакующего будем понимать величину

$$K_i(t) = \frac{1}{\tau_i(t)},$$

ко времени поражения $\tau_i(t)$.

Опасность показывает, насколько атакующий угрожает T – чем меньшее время ему необходимо для поражения T – тем выше его опасность.

Определение 3. Под удобством i -атакующего будем понимать величину

$$U_i(t) = \frac{1}{\theta_i(t)},$$

обратную времени перехвата $\theta_i(t)$.

Удобство показывает, насколько легко для защитника перехватить атакующего – чем меньшее на это необходимо время, тем выше его удобство.

Определение 4. Под преимуществом над i -атакующим будем понимать величину

$$A_i(t) = \tau_i(t) - \theta_i(t)$$

разности времен поражения $\tau_i(t)$ и перехвата $\theta_i(t)$.

Преимущество показывает запас времени на перехват – чем выше преимущество, тем дольше можно откладывать решение о назначении данной цели.

При планировании движения T разумно приоритетно уклоняться от атакующих, представляющих наибольшую угрозу, чтобы выиграть время на их перехват для D . Степень угрозы со стороны i -атакующего оценивается его опасностью $K_i(t)$, поэтому для нахождения вектора скорости цели логично использовать правило

$$(5) \quad \mathbf{V}_T(t) \sim \sum_{i=1}^n \frac{K_i(t)}{\sum_{j=1}^n K_j(t)} \frac{\mathbf{r}_T(t) - \mathbf{r}_i(t)}{|\mathbf{r}_i(t) - \mathbf{r}_T(t)|},$$

где знак пропорциональности означает, что величина справа задает направление скорости $\mathbf{V}_T(t)$, в то время как ее величина совпадает с V_T . Данное правило соответствует приоритетному убеганию от атакующих с наибольшей опасностью $K_i(t)$. Здесь следует сделать важную оговорку: использование данного правила лишает следствие 1 силы, так как теперь защищаемая цель не может прекратить движение, а значит, могут быть начальные условия, при которых не существует плана перехвата без пропусков, но такие конфигурации крайне редки.

Дополнительно мгновенные характеристики атакующих позволяют сформулировать локальные правила для защитника D :

$$(6) \quad P_Q(i) = \frac{Q_i(t)}{\sum_{j=1}^n Q_j(t)},$$

где вместо $Q_i(t)$ подставляют опасность $K_i(t)$ или удобство $U_i(t)$ i -цели. Величины $P_Q(i)$ могут использоваться как для детерминированного назначения атакующих защитнику по $\arg\max P_Q(i)$, так и для построения стохастической политики, когда $P_Q(i)$ трактуется как распределение вероятности назначений.

2.4. Понятие частичного плана и свойства задачи планирования

При использовании правила (5) для движения цели T – ее траектория полностью определяется начальными положениями и планом перехвата π . Введем для $\pi = [\pi_1 \dots \pi_n]$ последовательность времен $\mathbf{t} = [t_1 \dots t_n]$, необходимых D , чтобы обслужить первые k атакующих из π . Здесь под обслуживанием атакующего понимается либо его успешный перехват, либо поражение им T прежде его поимки. Тогда движение T вплоть до момента t_k зависит только от первых k элементов плана π , что мотивирует следующие определения.

Определение 5. Под *частичным планом* $\pi = [\pi_1, \dots, \pi_k]$ будем понимать упорядоченный набор из $k \leq n$ различных целей, задающий очередность их перехвата, которому соответствует промежуточное состояние $S(\pi)$. Под $S(\pi)$ будем понимать координаты $\mathbf{r}_T(t_k)$, $\mathbf{r}_D(t_k)$ и $\mathbf{r}_j(t_k)$, где t_k – время выполнения частичного плана π , $\mathbf{r}_j(t_k)$ – положения тех атакующих, которые не были перехвачены D и еще не поразили T к моменту завершения π , совместно с текущим вектором потерь $L(\pi) = (m_0(t_k), \dots, m_5(t_k))$, где $m_0(t_k)$ – число пропущенных целей, а $m_i(t_k)$ – числа перехваченных атакующих в соответствующей зоне к моменту t_k .

Введем для частичного плана оценку $l(\pi) = L(\pi) + Z(S)$, где $L(\pi)$ – вектор текущих потерь плана π , а $Z(S) = (0, z_1, \dots, z_5)$, где z_i – число целей, находящихся в соответствующей зоне потерь в промежуточном состоянии $S(\pi)$. Тогда верна теорема об оценке снизу.

Теорема 2. Итоговый вектор потерь $L(\pi^*)$ для любого продолжения плана π всегда не меньше, чем $l(\pi)$, т.е. $L(\pi^*) \geq l(\pi)$, если π^* есть продолжение π .

Доказательство. Действительно, так как скорость $V_A > V_T$, то расстояние $d = |\mathbf{r}_i(t) - \mathbf{r}_T(t)|$ между i -атакующим и целью T монотонно убывает с ростом времени t . Поэтому вектор потерь $L(\pi^*)$ любого продолжения частичного плана π будет включать как вектор потерь $L(\pi)$, так и потери по еще не перехваченным атакующим, которые не могут быть остановлены на расстоянии большем, чем текущая дистанция между ними и T .

Следствие 2. Если для некоторого частичного плана π оценка потерь $l(\pi) \geq L(\pi^*)$ превосходит потери для некоторого полного плана π^* , то тогда ни одно из продолжений π не может быть оптимальным.

Следствие 3. Если для некоторого частичного плана π верно, что оценки потерь $l(\pi')$ всех его продолжений фиксированной длины $\text{len}(\pi') = k$ превосходят оценку некоторого полного плана π^* (т.е. $\forall \pi' : \text{len}(\pi') = k \rightarrow l(\pi') \geq L(\pi^*)$), то ни одно продолжение π не может быть оптимальным.

Данные следствия позволяют отвергать заведомо неоптимальные направления перебора при поиске оптимального или субоптимального решения. При этом пользуясь тем, что заведомо известно существование плана, не пропускающего ни одного атакующего, в качестве базового значения недопустимой оценки $l(\pi)$ можно использовать вектор $l^* = [0, n, 0, 0, 0]$, соответствующий наихудшей оценке без пропусков.

Дополнительно, план перехвата π можно характеризовать набором локальных свойств его промежуточных состояний. Введем определения.

Определение 6. *Под сложностью плана π будем понимать максимальное время*

$$\theta(\pi) = \max_{k=1\dots n} \theta(t_k)$$

перехвата отдельной цели в рамках этого плана. Здесь величины $\theta(t_k)$ понимаются в смысле (4) относительно атакующего, который перехватывался в момент времени t_k из вектора $\mathbf{t}(\pi)$.

Определение 7. *Под надежностью плана π будем понимать минимальное преимущество*

$$A(\pi) = \min_{k=1\dots n} A(t_k)$$

над целью из плана π . Здесь $A(t_k)$ понимается в смысле определения 4 относительно атакующего, который перехватывался в момент времени t_k из вектора $\mathbf{t}(\pi)$.

Надежность и сложность плана – его глобальные свойства, которые могут использоваться для исследования зависимости функции потерь $L(\pi)$ от характеристик π .

3. Построение плана перехвата методом поиска по дереву Монте-Карло

В этом разделе обсуждается алгоритм построения субоптимальных планов по методу MCTS.

3.1. Дерево поиска по дереву Монте-Карло

Введем понятие дерева поиска.

Определение 8. *Под деревом поиска Ξ будем понимать множество частичных планов Π^* , снабженное отношением инцидентности: $\pi \rightarrow \pi'$, если π' отличается от π добавлением ровно одной цели в конец плана. Для частичного плана π множество $C(\pi) = \{\pi' \in \Pi^* : \pi \rightarrow \pi'\}$ всех инцидентных ему продолжений будем называть его потомками. Корнем дерева Ξ будем называть пустой план $\pi_0 = []$.*

Алгоритм MCTS выполняет интеллектуальный поиск по дереву Ξ , который направляется политикой (6) и q -оценками его узлов. Каждый план π помечается числом r_π продолжений, построенных для этого плана, и средней оценкой q_π , достигнутой на них. Здесь под средней q -оценкой понимается средняя

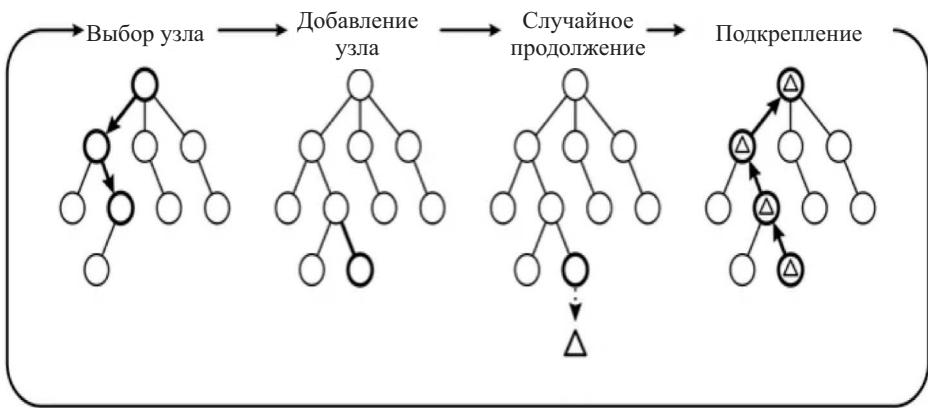


Рис. 1. Принципиальная схема алгоритма поиска по дереву Монте-Карло. Алгоритм состоит из чередования двух фаз: спуска по дереву к новому узлу и обратного распространения оценки этого узла обратно к корню.

скалярная оценка планов, построенных из этого узла (вектор потерь $L(\pi)$ преобразуется в скалярную оценку с помощью линейной функции, которая присваивает максимально возможным потерям оценку $-1,0$, а минимальным $+1,0$). Базовый цикл алгоритма состоит из двух процедур.

Алгоритм 1. Спуск по дереву

1. $\pi \leftarrow \pi_0$.
2. Пока $p_\pi > 0$: $\pi \leftarrow \arg \max_{\pi': \pi \rightarrow \pi'} q_{\pi'} + \frac{P_Q(\pi')\sqrt{n_\pi}}{1+n_{\pi'}}$.
3. Вернуть π .

Для найденного узла π строится случайное продолжение π^* по локальному правилу (6) и вычисляется вектор потерь $L(\pi^*)$, на основе которого формулируется q -оценка плана π , которая затем поднимается к корню.

Алгоритм 2. Обратное распространение оценки.

1. $r \leftarrow L_q(L(\pi^*))$, где $L_q(L(\pi^*))$ – q -оценка вектора потерь $L(\pi^*)$.
2. Пока $\pi = [\pi_1 \dots \pi_k]$ не π_0 : обновить p_π и q_π на основе r (число посещений увеличивается на единицу, а средняя q -оценка пересчитывается с учетом добавления r), $\pi \leftarrow \pi' = [\pi_1 \dots \pi_{k-1}]$.

Таким образом, данный алгоритм обновляет статистику, накопленную по узлам дерева, с учетом нового решения π^* , которое было построено из узла дерева, выбранного согласно алгоритму 1 спуска по дереву. Общая схема алгоритма изображается на рис. 1, более подробно об этом можно прочитать в [9].

3.2. Построение случайных продолжений

В предыдущем подразделе обсуждалось, что алгоритм MCTS полагается на случайные продолжения частичных планов для получения оценок узлов дерева. В этом подразделе подробно описывается методика построения таких продолжений и способ вычисления q -оценок.

Пусть требуется оценить узел дерева с частичным планом π , которому соответствует промежуточное состояние $S(\pi)$. Случайное продолжение строится как решение задачи планирования из состояния $S(\pi)$ на основе локальных правил (5) и (6), причем правило (6) используется как стохастическая политика назначения защитника на перехват атакующих.

Алгоритм 3. Построение случайного продолжения.

1. $\pi^* \leftarrow \pi$.

2. Пока есть свободные цели (т.е. неперехваченные и непропущенные): $\pi^* \leftarrow i$, которая выбирается по распределению вероятности $P_Q(i|S)$ для текущего промежуточного состояния $S(\pi^*)$.

Как только по данному алгоритму построен полный план π^* , то для него вычисляется вектор потерь $L(\pi^*)$. Затем $L(\pi^*)$ преобразуется в скалярную q -оценку в диапазоне $[-1, 1]$ с помощью линейной функции $L_q(L(\pi^*))$, которая действует по правилу

$$\begin{aligned} L_q([n, 0, 0, 0, 0, 0]) &= -1, \\ L_q([0, 0, 0, 0, 0, n]) &= 1, \end{aligned}$$

т.е. присваивает максимальным потерям оценку -1 , а минимальным $+1$. Данная оценка затем используется в алгоритме 2 обратного распространения оценки.

3.3. Отсечение неоптимальных узлов и построение итогового плана

Дополнительно результаты работы алгоритма поиска MCTS могут быть существенно улучшены на основе теоремы 2 и ее следствий. Для этого алгоритм на каждом шаге хранит память о наилучшем полном плане $\pi_{\max}$, который был построен при оценке одного из узлов дерева за все время работы, а также его вектор потерь $L(\pi_{\max})$. Узел дерева с частичным планом π считается неоптимальным, если его частичное состояние неоптимально относительно $L(\pi_{\max})$ согласно следствию 2 или если все его дочерние узлы были признаны неоптимальными.

В процессе спуска по дереву в алгоритме 1 выбор производится исключительно из узлов, которые не являются неоптимальными. Если же был достигнут узел, который необходимо признать неоптимальным, то алгоритм возвращается к его родительскому узлу и повторяет процесс поиска с учетом этой дополнительной информации. По исчерпанию вычислительного лимита алгоритм возвращает наилучший план $\pi_{\max}$, который был построен за все время его работы.

4. Экспериментальные результаты

Алгоритм поиска по дереву Монте-Карло совместно с эвристиками движения T и назначения целей D по опасности целей применялся для построения планов защиты от атаки от 10 до 30 целей. При этом построение эффек-

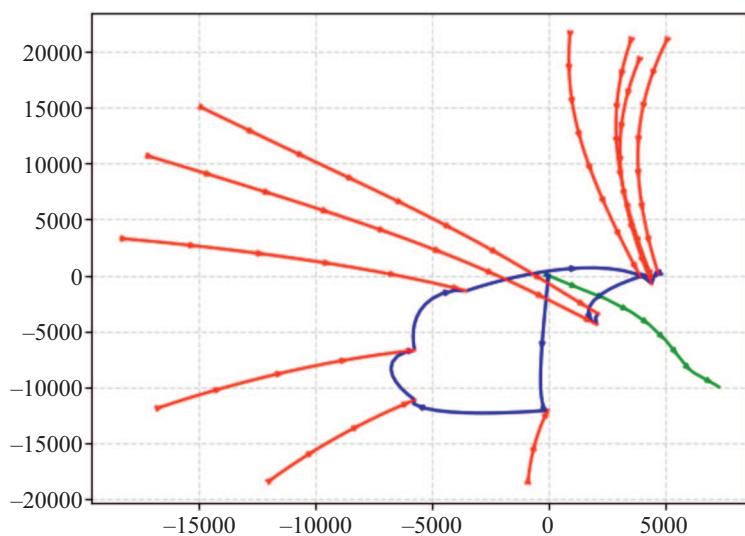


Рис. 2. План защиты от атаки 10 атакующих. Зеленой линией показана траектория защищаемой цели, синей – защитника-перехватчика, красными – атакующих. Стрелки указывают направление движения и взаимное расположение объектов на срезах равного времени.

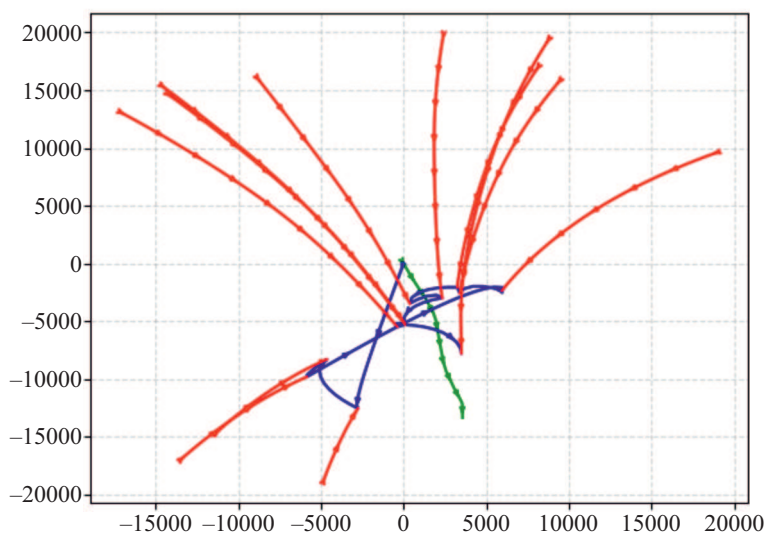


Рис. 3. План защиты от атаки 12 атакующих. Зеленой линией показана траектория защищаемой цели, синей – защитника-перехватчика, красными – атакующих. Стрелки указывают направление движения и взаимное расположение объектов на срезах равного времени.

тивных планов защиты в среднем требует от 1 до 10 с работы алгоритма в зависимости от размера задачи. Примеры характерных планов защиты приведены на рис. 2 и 3. Зависимость качества плана перехвата от времени ра-

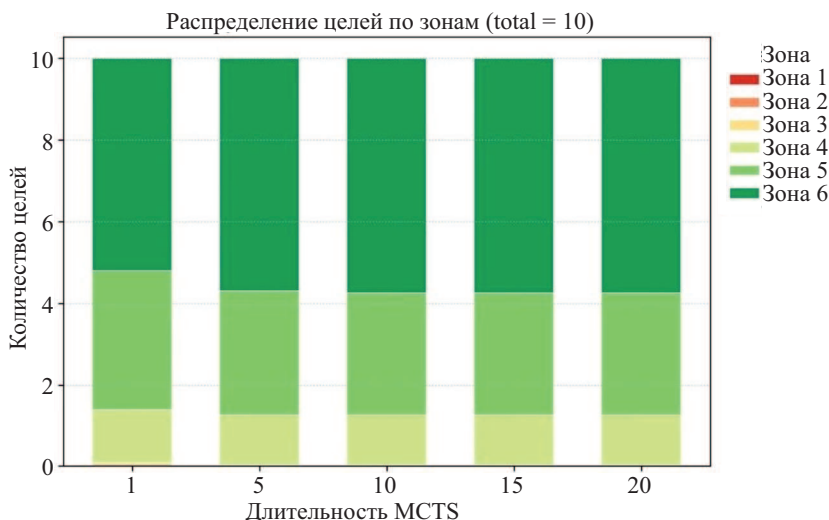


Рис. 4. Распределение точек перехвата по зонам близости в зависимости от времени работы MCTS для задач с $n = 10$ атакующих.

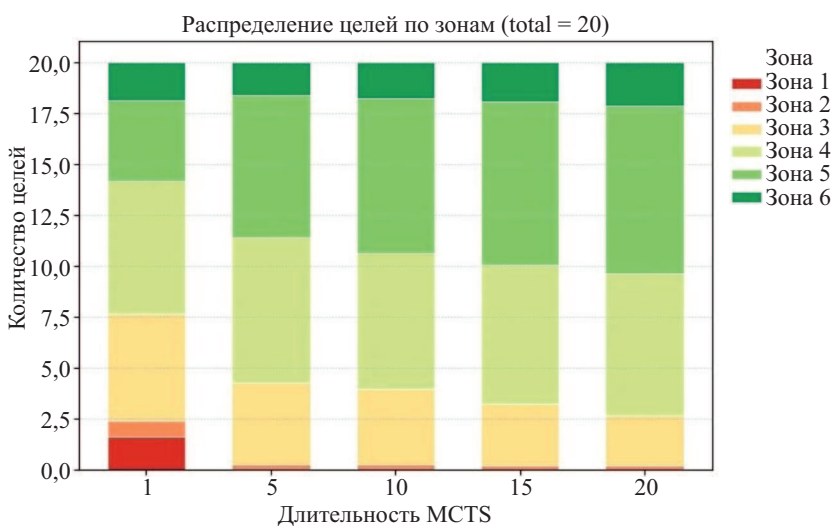


Рис. 5. Распределение точек перехвата по зонами близости в зависимости от времени работы MCTS для задач с $n = 20$ атакующих.

боты алгоритма для разного количества атакующих приведена на рис. 4–6, данные для построения которых набирались на выборке из 30 случайных начальных позиций с заданным числом атакующих. Таким образом можно видеть характерные времена построения субоптимальных планов для задач соответствующего размера.

Дополнительно, проводились исследования качества локальных правил (3) в сравнении с оптимальным планом, результаты которого приведены на

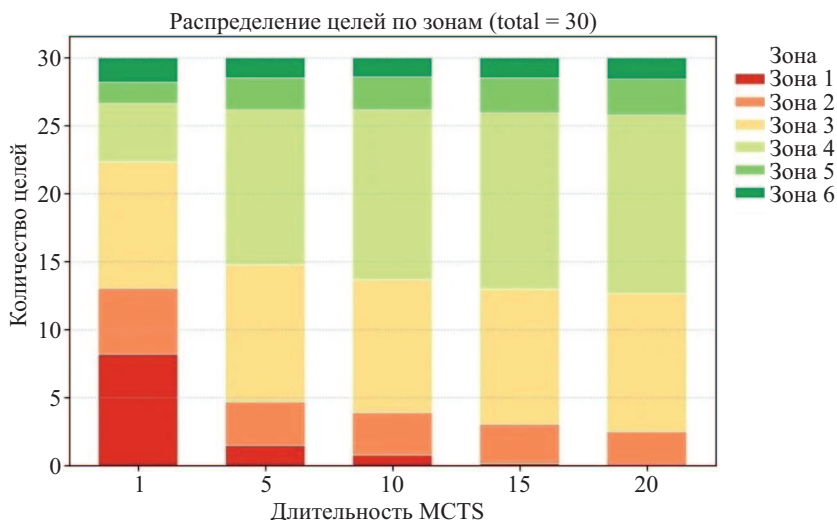


Рис. 6. Распределение точек перехвата по зонам близости в зависимости от времени работы MCTS для задач с $n = 30$ атакующих.

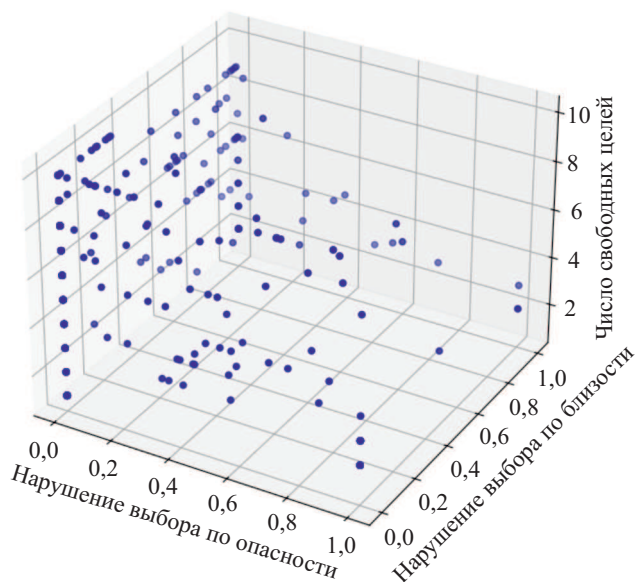


Рис. 7. Нарушение локальных правил опасности близости в оптимальном плане в зависимости от количества свободных целей в промежуточном состоянии.

рис. 7. Здесь под величиной нарушения правила полагается оценка, которая линейно изменяется от 0 до 1, где оценка 0 означает, что выбор цели в оптимальном правиле полностью согласован с эвристикой, а 1 – полностью несогласован (т.е. оптимальный выбор соответствует атакующему, который наименее привлекателен с точки зрения эвристики).

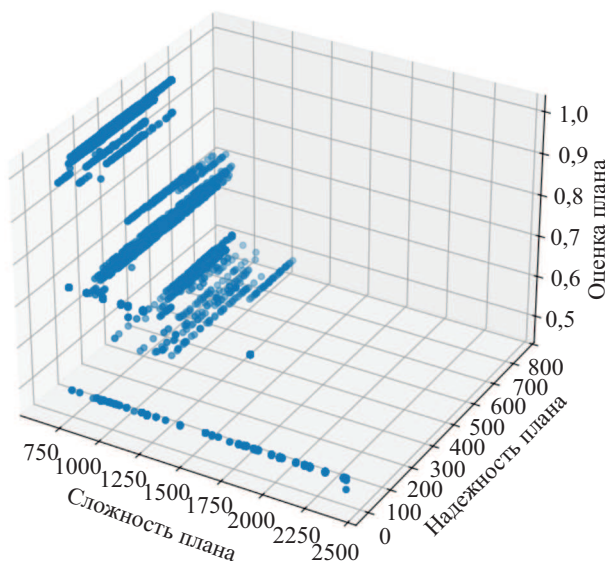


Рис. 8. Зависимость качества планов от их надежности и сложности.

Также для полноты картины приводим распределение планов по качеству в зависимости от их глобальных свойств сложности и надежности, построенное для множества всех допустимых планов для позиции с $n = 8$ атакующими.

5. Заключение

Данная работа была посвящена задаче защиты динамической цели от атаки множества атакующих с помощью защитника-перехватчика. Основные результаты работы можно кратко представить в следующих пунктах.

1) Задача защиты динамической цели – NP -сложная задача, для которой, однако, можно предложить эффективные алгоритмы построения субоптимальных решений.

2) Алгоритм поиска по дереву Монте-Карло показывает хорошие результаты на входных примерах с числом целей вплоть до $n = 30$ за разумное вычислительное время.

3) В качестве опорных локальных правил для MCTS логично использовать эвристики на основе опасности и близости, определенные согласно распределению вероятности 6.

Алгоритм MCTS строит эвристику повышенного качества на основе исходных эвристик за счет интеллектуального дополнительного перебора вариантов. Распределения чисел посещений дочерних узлов в корне дерева можно трактовать как улучшенную локальную политику, которая затем может использоваться для первичного обучения нейросетевых моделей, что открывает перспективы применения полученных в данной работе результатов для по-

строения больших нейросетевых моделей данной задачи. С другой стороны, возможность эффективно решать задачу защиты без привлечения нейросетевого обучения ценна сама по себе, так как нейросетевые подходы не всегда оказываются эффективнее простых эвристических решателей [15, 16] в ДЗК.

СПИСОК ЛИТЕРАТУРЫ

1. *Сихарулидзе Г.Г.* Об одном обобщении задачи коммивояжера. I // *АиТ.* 1971. № 8. С. 116–123.
2. *Сихарулидзе Г.Г.* Об одном обобщении задачи коммивояжера. II // *АиТ.* 1971. № 10. С. 142–147.
3. *Picard J.C., Queyranne M.* The time-dependent traveling salesman problem and its application to the tardiness problem in one-machine scheduling // *Oper. Res.* 1978. V. 26. No. 1. P. 86–110. <https://doi.org/10.1287/opre.26.1.86>
4. *Helvig C.S., Robins G., Zelikovsky A.* The moving-target traveling salesman problem // *J. Algorithm. Comput. Technol.* 2003. V. 49. No. 1. P. 153–174. [https://doi.org/10.1016/S0196-6774\(03\)00075-0](https://doi.org/10.1016/S0196-6774(03)00075-0)
5. *Li C., Yang M., Kang L.* A New Approach to Solving Dynamic Traveling Salesman Problems. In: Wang, TD., et al. *Simulated Evolution and Learning* // *Lecture Notes Comput. Sci.* 2006. V. 4247. Springer, Berlin, Heidelberg.
6. *Stieber A.* Dealing with time in the multiple traveling salesmen problem with moving targets // *Cottbus Mathematical Preprints* 2019. COMP. 4.
7. *Maskooki A., Kallio M.* A bi-criteria moving-target travelling salesman problem under uncertainty // *European Journal of Operational Research* 2023. Vol. 309. No. 1. С. 271–285. <https://doi.org/10.1016/j.ejor.2023.01.009>
8. *Garey M.R., Johnson D.S.* Computers and intractability: A guide to the theory of NP-completeness. San Francisco, Calif.: W. H. Freeman & Co., 1979.
9. *Рябушев Е.А., Галляев А.А.* Поиск субоптимального решения динамической задачи коммивояжера методом Монте-Карло // *АиТ.* 2024. № 2. С. 103–119.
10. *Галляев А.А., Яхно В.П., Берлин Л.М. и др.* Оптимизация плана перехвата прямолинейно движущихся целей // *АиТ.* 2023. № 10. С.18–36.
11. *Галляев А.А., Рябушев Е.А.* Локальное динамическое планирование для защиты подвижной цели от атаки скоростного перехватчика с ограниченным обзором // *Труды института математики и механики УрО РАН* 2026. Том 32. № 3.
12. *Silver D., Huang A., Maddison C., et al* Mastering the game of Go with deep neural networks and tree search // *Nature.* 28 January 2016. 529 (7587): P. 484–489. <https://doi.org/10.1038/nature16961>
13. *Schadd M.P.D., Winands M.H.M., van den Herik H.J., et al.* Single-Player Monte-Carlo Tree Search // *Computers and Games. CG* 2008. *Lecture Notes in Computer Science*, vol. 5131. Springer, Berlin, Heidelberg.
14. *Marco K.* Beyond Games: A Systematic Review of Neural Monte Carlo Tree Search Applications // *arXiv:2303.08060*, <https://doi.org/10.48550>
15. *Xuanhao P., Chenguang W., Chaolong Y., et al.* Rethinking the Heatmap + Monte Carlo Tree Search Paradigm for Solving Large Scale TSP // 2025. <https://openreview.net/forum?id=TMHOHRR0FA>

16. Xia Y., Xianliang Y., Zichuan L., et al. Position: Rethinking Post-Hoc Search-Based Neural Approaches for Solving Large-Scale Traveling Salesman Problems // Proceedings of the 41st International Conference on Machine Learning, 2024. Vol. 235 C. 54178–54190. <https://proceedings.mlr.press/v235/xia24f.html>

Статья представлена к публикации членом редколлегии А.А. Лазаревым.

Поступила в редакцию 24.06.2026

После доработки 06.07.2026

Принята к публикации 08.07.2026