УДК 518.9

С.В. Пашко, А.Л. Яловец

ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ ЗАДАЧ ОПТИМИЗАЦИИ ПРЕСЛЕДОВАНИЯ

Рассмотрены дифференциальные игры преследования на плоскости, в которых для каждого убегающего создается группа преследователей. Сформулированы задачи оптимизации групп преследования. Построены численные методы решения таких задач, выполнены вычислительные эксперименты и сделаны выводы об эффективности методов.

Введение

В данной работе рассматриваются методы решения задач оптимизации преследования на плоскости, применяемые в разрабатываемом программном обеспечении.

Одно из центральных мест в теории динамических игр занимают задачи преследования и убегания. Динамические игры, согласно [1], разделяются на игры качества и игры степени. В играх качества представляют интерес два исхода игры. Например, требуется определить, могут ли преследователи захватить цель до определенного момента времени или нет. В играх степени требуется построить оптимальные стратегии игроков, для которых достигается минимакс функции платы. В настоящей работе задача преследования в пределах одной группы рассматривается как игра степени.

Следует отметить достигнутые успехи в построении роботизированных систем преследователей на практике, в которых несколько роботов преследуют некоторое количество целей [2].

В работе [3] изучается сложность задач преследования, в которых число преследователей и число преследуемых больше одного. В этих задачах критерием качества является время окончания игры, т. е. время захвата всех целей. Движения игроков считаются простыми, при этом предполагается, что максимальная скорость каждого преследователя превосходит максимальную скорость любого убегающего. Каждому преследователю разрешается захватить не более одного убегающего. Множество преследователей разбивается на группы, причем для каждой цели создается одна группа [4]. В любой момент времени группы могут быть переформированы. После захвата цели вся группа вместе с целью выбывают из игры. Все цели должны быть захвачены. Требуется найти оптимальный состав групп, т. е. такой, для которого момент захвата последней цели минимален. В каждой группе преследователи и убегающий игрок применяют оптимальные или близкие к ним стратегии движения; такого рода стратегии изучались в [1, 5–9].

В работе [3] показано, что задача оптимизации состава групп преследования является NP-трудной в сильном смысле. Из этого сделаны выводы о том, какие численные методы следует применять для ее решения.

В настоящей работе описаны метод случайного поиска с локальной оптимизацией и метод ветвей и границ для оптимизации групп преследования. Выполнены численные эксперименты, из которых следуют выводы об эффективности этих методов.

1. Преследование одного убегающего

Рассмотрим задачу оптимального преследования одного убегающего группой преследователей. Пусть на плоскости в точке X_0 находится преследуемый игрок *E*, а в точках X_i находятся преследующие его игроки P_i, $i = 1, 2, \ldots, k$. Обозначим V_i скорости игроков, $i = 0, 1, 2, \ldots, k$ (нулевое значение индекса *i* относится к игроку *E*); здесь V_i – двумерные векторы. Уравнения движения игроков имеют вид

$$\dot{X}_i = V_i, \quad i = 0, 1, 2, \dots, k. \qquad (1)$$

Будем считать, что выполняются ограничения

$$\|V_i\| \le w_i < \infty, \quad i = 0, 1, 2, \dots, k, \qquad (2)$$

где w_i – максимальная величина скорости, $\|X\| = \sqrt{x^2 + y^2}$ – длина вектора $X = (x, y)$. Кроме того, скорость V_i считается кусочно-непрерывной функцией от времени. Это означает, что в каждом ограниченном временном интервале существует не больше конечного числа точек разрыва первого рода.

Игрок *i* управляет своими координатами, выбирая в каждый момент времени скорость V_i. Функция $V_i(t)$ – управление *i*-го игрока. Движение, осуществляемое таким образом, называется простым. Игра начинается в момент времени $t = 0$ и считается законченной, когда координаты одного из преследователей в некоторый момент $t = T$ совпадают с координатами *E*, случай $T = \infty$ не исключается. Преследователи стремятся уменьшить время *T*, преследуемый стремится его увеличить. Считаем, что

$$w_0 < w_i, \quad i = 1, 2, \dots, k. \qquad (3)$$

Стратегию игрока *i* можно определить как функцию $S_i(t, I(t))$, где $I(t)$ – информация, доступная игроку в момент *t*. Значением этой функции является управление $V_i(t)$; игрок *i* вычисляет свое управление по формуле $V_i = S_i(t, I(t))$. Пусть $X^u(t) = (X_0(t), X_1(t), \dots, X_k(t))$ – фазовый вектор, содержащий координаты игроков, зависящие от времени. Далее считается, что стратегия убегающего зависит только от времени и фазового вектора, а стратегии преследователей зависят еще и от скорости убегающего игрока. Управления $V_i(t)$ вычисляются по формулам:

$$V_0(t) = S_0(t, X^u(t)), \qquad (4)$$

$$V_i(t) = S_i(t, X^u(t), V_0(t)), \; i = 1, 2, \dots, k. \qquad (5)$$

Используя уравнения (1), (4), (5), приходим к системе дифференциальных уравнений

$$\dot{X}_0(t) = S_0(t, X^u(t)), \qquad (6)$$

$$\dot{X}_i(t) = S_i(t, X^u(t), S_0(t, X^u(t))),$$

$$i = 1, 2, \dots, k, \qquad (7)$$

$$0 \le t \le T, \quad X^u(0) = X^{u0}. \qquad (8)$$

Здесь X^{u0} – заданное начальное значение вектора X^u. Решением системы (6)–(8) считается абсолютно непрерывная функция $X^u(t) = (X_0(t), X_1(t), \dots, X_k(t))$, производная которой всюду удовлетворяет соотношениям (6), (7) за исключением, быть может, конечного числа моментов времени в каждом ограниченном временном отрезке.

Пусть $S = (S_1, S_2, \dots, S_k)$ – стратегия преследования. Пара стратегий убегания и преследования (S_0, S) называется совместной, если существует единственное решение $X^u(t)$ системы уравнений (6)–(8), причем управления $V_i(t)$, вычисляемые по формулам (4), (5), являются кусочно-непрерывными функциями от *t* и удовлетворяют соотношениям (2). Приведенное определение совместной пары стратегий аналогично определению, данному в [6].

Пусть величина $T(X^u(0), S_0, S)$ равна времени захвата цели, если стратегии S_0 и *S* совместны, и равна ∞ в противном случае. Оптимальным временем преследования назовем число

$$t^* = \inf_S \sup_{S_0} T(X^u(0), S_0, S). \qquad (9)$$

Ясно, что из соотношений (3) следует $t^* < \infty$. Назовем стратегию S^* оптимальной стратегией преследования, если для каждой стратегии S_0 выполняется неравенство $T(X^u(0), S_0, S^*) \le t^*$. Назовем стратегию S_0^* оптимальной стратегией убегания, если для каждой стратегии S справедливо неравенство:

$$T(X^u(0), S_0^*, S) \ge t^*.$$

Обозначим $V = (V_1, V_2, ..., V_k)$ управление преследования, соответствующее стратегии S. Предположим, преследуемый объект и преследователи применяют оптимальные стратегии S_0^* и S^*. Получающиеся при этом управления назовем парой оптимальных управлений (V_0^*, V^*).

В случае $k = 2$ оптимальные управления V_0^* и V^* указаны в [1] (раздел 6.8). Для произвольных значений k задачи оптимального преследования изучаются в [5, 6].

Если имеется только один преследователь, т. е. $k = 1$, то оптимальное время преследования вычисляется по формуле:

$$t^* = d/(w_1 - w_0),$$

где d – расстояние между игроками.

Рассмотрим подробнее игру с двумя преследователями, т. е. при условии $k = 2$. Обозначим A_i окружность Аполлония, относящуюся к паре игроков E, P_i, $i = 1, 2$. Она представляет собой геометрическое место точек, в которых могут встретиться игроки E и P_i, если они двигаются равномерно и прямолинейно с максимальными скоростями. Пусть Q_i – замкнутый круг Аполлония, соответствующий окружности A_i. Тогда $Q = Q_1 \cap Q_2$ – множество таких точек плоскости, до которых игрок E успевает дойти не позже каждого преследователя. Внутренность множества Q представляет собой множество точек, до которых игрок E успевает дойти раньше любого преследователя. Пусть $X^* \in Q$ – наиболее удаленная точка от точки X_0 из множества Q, т. е. для каждого $X \in Q$ справедливо неравенство $\|X^* - X_0\| \ge \|X - X_0\|$.

Обозначим S_0^* стратегию убегающего игрока, которая состоит в том, что он двигается равномерно и прямолинейно с максимальной скоростью по направлению к точке X^*. До момента времени, равного $\|X^* - X_0\|/w_0$, захват произойти не может, поэтому

$$t^* \ge \|X^* - X_0\|/w_0.$$

В данной игре каждый из преследователей может применить стратегию параллельного сближения [8]. В работе [9] доказано, что применение каждым преследователем этой стратегии приводит к захвату цели за время, которое не превосходит величины $\|X^* - X_0\|/w_0$, откуда следует неравенство:

$$t^* \le \|X^* - X_0\|/w_0.$$

Из двух последних неравенств вытекает соотношение

$$t^* = \|X^* - X_0\|/w_0.$$

Управление убегающего игрока, при котором он движется равномерно и прямолинейно с максимальной скоростью по направлению к точке X^*, обозначим V_0^*. Управление преследователей, при котором они движутся равномерно и прямолинейно с максимальной скоростью по направлению к той же точке X^*, обозначим V^*. Пара (V_0^*, V^*) – это пара оптимальных управлений.

2. Задача оптимизации групп преследования

Рассмотрим задачу формирования оптимальных групп преследования. Пусть на плоскости имеются m преследователей и n убегающих, причем $m \geq n > 1$. Все игроки обладают простым движением, максимальная скорость каждого преследователя больше максимальной скорости любого убегающего. Для каждого убегающего образуется группа преследователей, которая после его захвата выбывает вместе с ним из игры. Цель считается захваченной, если ее координаты совпадают с координатами одного из преследователей, входящих в группу. Для каждой цели заданы минимальное и максимальное количества игроков, которые могут принадлежать группе. Все цели должны быть захвачены. Требуется составить группы преследования таким образом, чтобы общее время завершения игры принимало минимальное значение.

Как и в [3], считаем, что в каждой группе игроки применяют оптимальные (или почти оптимальные) стратегии, а время захвата t^* вычисляется по формуле (9).

Пусть целочисленный вектор $J = (j_1, j_2, ..., j_m)$ задает распределение преследователей по группам. Число j_i означает номер цели, которую преследует i-й преследователь, $i \in \{1, 2, \dots, m\}$, $j_i \in \{0, 1, 2, ..., n\}$; если $j_i = 0$, то i-й преследователь не принимает участия в погоне. Обозначим $t_j^*(J)$ оптимальное время преследования в j-й группе (относящейся к j-й цели). Пусть натуральное число $k_j(J)$ равно количеству преследователей в j-й группе. Задачу оптимизации групп преследования можно записать следующим образом:

$$\max_{j=1,2,...,n} t_j^*(J) \underset{J}{\rightarrow} \min, \qquad (10)$$

$$k_j^{(1)} \leq k_j(J) \leq k_j^{(2)}, \ j = 1,2,...,n. \qquad (11)$$

Здесь $k_j^{(1)}, k_j^{(2)}$ задают соответственно минимально возможное и максимально возможное число преследователей в j-й группе.

Входными данными этой задачи являются координаты на плоскости всех игроков в начале игры, их максимальные скорости, числа $k_j^{(1)}$, $k_j^{(2)}$, m, n. Считаем, что входные данные представляют собой целые числа; максимальные скорости и числа $k_j^{(1)}$, $k_j^{(2)}$, m, n считаются положительными. Предполагаем, что система ограничений (11) совместна, т. е. выполняются соотношения $k_j^{(1)} \leq k_j^{(2)}$, $j = 1, 2, \dots, n$; $m \geq \sum_{j=1}^{n} k_j^{(1)}$.

Задачу (10), (11) назовем ОПТИМИЗАЦИЯ ГРУПП ПРЕСЛЕДОВАНИЯ (сокращенно ОГП). В задаче требуется найти оптимальное значение вектора J, состоящего из целых чисел. Поскольку входные данные также являются целыми числами, то имеет смысл вопрос о том, какова алгоритмическая сложность задачи ОГП. В работе [3] доказана следующая теорема.

Теорема. Задача ОГП – NP-трудная в сильном смысле.

Заметим, что в [3] эта теорема доказана при условии, что $j_i \in \{1,2,...,n\}$ (все преследователи назначаются на цели). Однако такая задача тривиальным образом сводится по Тьюрингу к рассматриваемой здесь задаче ОГП. Поэтому теорема остается справедливой при условии $j_i \in \{0,1,2,...,n\}$.

Задача ОГП – NP-трудная в сильном смысле уже в случае, когда для каждого убегающего необходимо назначить ровно двух преследователей.

Из приведенной теоремы можно сделать выводы о том, какие методы следует применять для решения задачи ОГП. В общем случае задача ОГП не может быть решена полиномиальным или псевдополиномиальным алгоритмом (при условии P $\neq$ NP). Для решения подобных задач применяются эвристические алгоритмы, методы полного перебора, ветвей и границ, случайного поиска с локальной оптимизацией.

3. Методы оптимизации групп преследования

Опишем методы поиска оптимального решения J^* задачи ОГП: метод ветвей и границ и методы случайного поиска с локальной оптимизацией. Изложение общих схем этих методов имеется в [10].

Метод ветвей и границ в процессе работы разбивает все множество допустимых решений задачи ОГП на подмножества. Каждое подмножество решений определяется вектором *J*, отождествляющимся с некоторой вершиной дерева ветвления. Корневой вершине дерева соответствует множество всех допустимых решений; все компоненты соответствующего вектора *J* равны нулю. Листу дерева соответствует одно допустимое решение задачи ОГП.

В процессе ветвления вершина дерева, не являющаяся листом, порождает несколько вершин. Считаем, что в исходной вершине (не являющейся корнем дерева) полностью сформированы группы преследования для целей с номерами $1,2,\ldots,n_0$, где $n_0 \equiv n_0(J)$, $1 \le n_0 \le n$. Порожденная вершина отличается тем, что в ней полностью сформирована группа преследования для (n_0+1)-й цели. Рассмотрим пример. Пусть для задачи ОГП выполняются условия $m=4$, $n=2$, $k_1^{(1)}=k_2^{(1)}=1$, $k_1^{(2)}=k_2^{(2)}=2$. Корневая вершина $J=(0,0,0,0)$ порождает десять вершин $(1,0,0,0)$, $(0,1,0,0)$, $(0,0,1,0)$, $(0,0,0,1)$, $(1,1,0,0)$, $(1,0,1,0)$, $(1,0,0,1)$, $(0,1,1,0)$, $(0,1,0,1)$, $(0,0,1,1)$. Первая из этих вершин означает, что для поимки первой цели сформирована группа, состоящая из одного преследователя (первого). Последняя вершина означает, что группу для первой цели составляют третий и четвертый преследователи. Первая вершина $(1,0,0,0)$ порождает шесть вершин $(1,2,0,0)$, $(1,0,2,0)$, $(1,0,0,2)$, $(1,2,2,0)$, $(1,2,0,2)$, $(1,0,2,2)$, являющихся листьями. Последний лист $(1,0,2,2)$ означает, что на первую цель назначен первый преследователь, а вторую цель преследует группа, состоящая из третьего и четвертого преследователей. Заметим, что число n_0 равно значению максимальной компоненты вектора *J*.

В процессе работы метода ветвей и границ для вершин, не являющихся листьями, вычисляются оценки $t_0 \equiv t_0(J)$, $t_1 \equiv t_1(J)$, $t_2 \equiv t_2(J)$.

Оценка t_0 для вершины *J* строится следующим образом. Пусть $\{1,2,...,n_0\}$ – множество целей, для которых вектором *J* сформированы группы преследования, $\{i_1,i_2,...,i_s\}$ – множество всех задействованных преследователей. Из этих множеств образуем новую задачу ОГП в форме (10), (11), заменяя число *m* числом *s*, а число *n* числом n_0. Найдем допустимое, близкое к оптимальному решение J' этой задачи, применяя алгоритм локальной оптимизации A_0, который описан далее. Значение целевой функции в точке J' выбирается в качестве оценки t_0. Ясно, что при условии $t_0 < \max\limits_{j=1,2,...,n_0} t_j^*(J)$ множество допустимых решений, определяемое вектором *J*, не содержит оптимального решения и вершину *J* можно исключить из рассмотрения.

Величина t_1 – оценка снизу значения целевой функции (10) на множестве допустимых решений, заданных вектором *J*. Она вычисляется следующим образом. Для каждой цели *j* из множества $\{n_0+1, n_0+2,...,n\}$ выбираем группу преследователей из числа незадействованных такую, что выполняются условия (11) и время захвата t_j^* принимает минимальное значение. Полагаем

$$t_1 = \max\{\, t(J), t_{n_0+1}^*, t_{n_0+2}^*,..., t_n^*\}.$$

Величина t_2 – оценка сверху значения целевой функции (10) на множестве допустимых решений, заданных вектором *J*. Она получается следующим образом. Используя множество незадействованных преследователей, вектор *J* преобразуем в допустимый вектор задачи ОГП с помощью эвристического алгоритма A_1, формирующего оставшиеся несформированными группы преследователей с номерами $n_0+1, n_0+2, ..., n$. Этот алгоритм описан далее. Затем выполняем алгоритм A_0 локальной оптимизации, в результате работы которого получаем точку локального минимума J', представляющую собой допустимое решение задачи ОГП. Значение целевой функции (10) в точке J' принимаем в качестве оценки t_2.

Текущее множество векторов *J*, рассматриваемых методом, обозначим *Q*. Пусть J^* – оптимальное решение задачи ОГП, t^* – оптимальное значение целевой функции. Обозначим $n_1 \equiv n_1(J) = n_0(J)+1$.

Метод ветвей и границ состоит из следующих шагов.

1. Полагаем $J=(0,...,0)$, $t^* = t_2(J)$, $J^* = J'$, где J' – точка локального минимума, соответствующая значению $t_2(J)$.

2. Если множество *Q* пусто, то вычисления прекращаем; величина t^* равна минимуму общего времени преследования, а соответствующий ей вектор J^* определяет оптимальное распределение преследователей по группам.

3. Из множества *Q* выбираем вектор *J* с минимальным значением $t_1(J)$. Вектор *J* удаляем из множества *Q*. Если выполняется соотношение $t_1(J) > t^*$, переходим к шагу 2.

4. Выполняем шаг ветвления. Вектор *J* преобразуем во множество векторов $R=\{J_1, J_2, ..., J_r\}$ следующим образом. Из всех незадействованных преследователей составляем всевозможные группы преследования для цели с номером n_1, удовлетворяющие условиям (11). Пусть количество таких групп равно *r*. Каждая группа представляет собой множество номеров преследователей $\{i_1, i_2, ..., i_k\}$. Используя вектор *J* и *s*-ю по порядку группу преследования $\{i_1, i_2, ..., i_k\}$, образуем вектор J_s из множества *R*. Компоненты вектора J_s равны компонентам вектора *J*, за исключением того, что компоненты с номерами $i_1, i_2, ..., i_k$ вектора J_s равны числу n_1.

5. Предположим, $n_1 = n$ (все группы сформированы). Для каждого вектора J_s из множества *R* вычисляем значение $t(J_s) = \max\limits_{j=1,2,...,n} t_j^*(J_s)$ целевой функции (10) и, если это значение меньше t^*, выполняем присвоения $t^* = t(J_s)$, $J^* = J_s$. Переходим к шагу 2.

6. Предположим, $n_1 < n$. Для каждого вектора J_s из множества *R* выполняем следующие действия.

6.1. Если согласно вектору J_s число незадействованных преследователей меньше нужного количества, требуемого ограничениями (11), удаляем вектор J_s из множества *R*.

6.2. Если $t_0(J_s) < \max\limits_{j=1,2,...,n_1} t_j^*(J_s)$, удаляем вектор J_s из множества *R*.

6.3. Если $t_2(J_s) < \max\limits_{j=1,2,...,n_1} t_j^*(J_s)$, удаляем вектор J_s из множества *R*. Если $t_2(J_s) < t^*$, полагаем $t^* = t_2(J_s)$, $J^* = J'_s$, где J'_s – точка локального минимума, соответствующая значению $t_2(J_s)$.

6.4. Если $t_1(J_s) > t^*$, удаляем вектор J_s из множества *R*.

Все оставшиеся во множестве *R* векторы добавляем к множеству *Q*. Переходим к шагу 2. □

Алгоритм локальной оптимизации A_0, использующий начальное допустимое решение *J* задачи ОГП, состоит из следующих шагов.

1. Находим номер j_0 группы, время преследования в которой максимально среди всех групп.

2. Поочередно для групп $j = 1, 2, ..., j_0 - 1, j_0 + 1, ..., n$ выполняем следующее. Пусть $I_0 = \{i_{01}, i_{02}, ..., i_{0k_0}\}$ – множество преследователей, составляющих группу j_0, а $I_1 = \{i_{11}, i_{12}, ..., i_{1k_1}\}$ – множество преследователей, составляющих группу j.

Перераспределяя преследователей множества $I_0 \bigcup I_1$ между группами j_0 и j всеми возможными способами, удовлетворяющими ограничениям (11), образуем множество пар групп. В каждой паре первая группа преследует цель j_0, а вторая – цель j. Пусть время преследования в первой группе равно $\bar{t}_0$, а во второй – $\bar{t}_1$. Выберем пару, в которой общее для обеих групп время преследования $t = \max(\bar{t}_0, \bar{t}_1)$ минимально. Пусть первая группа образована множеством преследователей $I_2 = \{i_{21}, i_{22}, ..., i_{2k_2}\}$, а вторая – множеством $I_3 = \{i_{31}, i_{32}, ..., i_{3k_3}\}$; при этом $I_0 \bigcup I_1 = I_2 \bigcup I_3$. Если время t меньше времени преследования в исходной группе j_0, то группу I_0 заменим группой I_2, а группу I_1 заменим группой I_3. Произведем соответствующие изменения в векторе J и перейдем к шагу 1.

Если не найдена группа j, позволяющая уменьшить максимальное время преследования в группах j_0 и j, вычисления прекращаем. □

Рассмотрим **эвристический алгоритм** A_1 построения близкого к оптимальному допустимого решения задачи ОГП. Предполагается, что имеется начальный вектор J, в котором первые n_0 групп преследования уже сформированы. Алгоритм A_1 формирует только группы преследования $(n_0+1), (n_0+2), ..., n$. В частности, возможно $n_0 = 0$, т. е. алгоритм формирует все группы. Алгоритм A_1 состоит из следующих шагов.

1. Просматриваем все группы от (n_0+1)-й до n-й. Если в j-й группе число преследователей меньше величины $k_j^{(1)}$, входящей в (11), то в эту группу добавляется незадействованный преследователь, обеспечивающий наибольшее уменьшение времени преследования в группе. В модифицированной таким образом группе время преследования определяется согласно формуле (9). Шаг 1 выполняется до тех пор, пока имеется хотя бы одна цель j, число преследователей которой меньше, чем $k_j^{(1)}$.

2. Просматриваем все группы от (n_0+1)-й до n-й. Если в j-й группе число преследователей меньше величины $k_j^{(2)}$, входящей в (11), то в эту группу добавляется незадействованный преследователь, обеспечивающий наибольшее уменьшение времени преследования в группе. Шаг 2 выполняется до тех пор, пока имеется хотя бы одна цель j, число преследователей которой меньше, чем $k_j^{(2)}$, и есть незадействованные преследователи. □

Метод случайного поиска с локальной оптимизацией состоит из следующих шагов.

1. Случайным образом формируем допустимое решение J задачи ОГП. Используя решение J в качестве начального, с помощью алгоритма локальной оптимизации A_0 вычисляем локальный оптимум J'.

2. Из всех полученных локальных оптимумов J' запоминаем наилучший. Если выполняется критерий останова, прекращаем вычисления, иначе переходим к шагу 1. □

На шаге 1 решение J выбирается согласно равномерному (или близкому к нему) распределению вероятностей на множестве допустимых решений.

В качестве критерия останова можно выбрать достижение заранее заданного числа повторения шага 1 или достижение заранее заданного числа шагов, на протяжении которых наилучшее достигнутое значение целевой функции не изменяется.

4. Результаты численных экспериментов

Рассмотрим результаты численных экспериментов с методами полного перебора, ветвей и границ, случайного поиска с локальной оптимизацией. Ограничимся случаем, когда в каждой группе преследования допускается участие не более двух преследователей, т. е. справедливы соотношения

$$1 \le k_j^{(1)} \le k_j^{(2)} \le 2,\ j = 1,2,...,n. \qquad (12)$$

Для этого имеются две причины. Во-первых, в задачах преследования на плоскости такой численный состав групп представляется актуальным. Во-вторых, в случае, если число преследователей в группе больше двух, алгоритм расчета оптимального времени согласно соотношению (9) известен не для всех способов взаимного расположения преследователей и цели.

Оценим количество допустимых решений задачи ОГП при условиях (12). Пусть число N_1 равно количеству допустимых решений при условиях $k_j^{(1)} = 1$, $k_j^{(2)} = 1$, $j = 1,2,...,n$; число N_2 равно количеству допустимых решений при условиях $k_j^{(1)} = 2$, $k_j^{(2)} = 2$, $j = 1,2,...,n$; число N_3 равно количеству таковых при условиях $k_j^{(1)} = 1$, $k_j^{(2)} = 2$, $j = 1,2,...,n$. Нетрудно вывести формулы

$$N_1 = \frac{m!}{(m-n)!},\quad N_2 = \frac{m!}{(m-2n)!2^n},$$

$$N_3 = \sum_{j=0}^{n} \frac{m!}{(m-2n+j)!2^{n-j}}.$$

Значения величин N_1, N_2, N_3 при некоторых значениях m и n приведены в табл. 1.

Вычислительные эксперименты проводились на персональном компьютере с процессором Pentium® Dual-Core 2,5 GHz. Задачи ОГП формировались случайным образом. При этом все координаты (т. е. x или y) целей и преследователей считались независимыми случайными величинами, равномерно распределенными на отрезке [–1;1]. Величины скоростей целей и преследователей считались независимыми случайными величинами, равномерно распределенными на отрезках [1; 1,9] и [2; 2,9] соответственно.

Задачу ОГП при условиях $m = 10$, $n = 5$, $k_j^{(1)} = 1$, $k_j^{(2)} = 2$, $j = 1,2,...,n$, метод полного перебора решил за 2,5 сек. процессорного времени. При условиях $m = 12$, $n = 6$, $k_j^{(1)} = 1$, $k_j^{(2)} = 2$, $j = 1,2,...,n$, этим методом для решения использовано больше, чем 5 минут. Эти данные и табл. 1 позволяют приблизительно оценить количество времени, необходимое для работы метода полного перебора. Метод пригоден только для задач малой размерности и значительно уступает методу ветвей и границ по скорости.

В табл. 2 приведены характеристики работы метода ветвей и границ в зависимости от количества участников. Для каждой пары значений m и n решено сто задач ОГП, после чего вычислены данные, приведенные в табл. Символами E, M, σ обозначены соответственно среднее время решения задачи, максимальное время решения задачи и стандартное отклонение времени решения в секундах. Символом S обозначено среднее число решений, помещенных методом ветвей и границ во множество Q в процессе решения задачи.

Вычислительные эксперименты говорят о том, что среднее время решения задачи ОГП методом ветвей и границ быстро растет с ростом чисел m и n, что подтверждается данными табл. 2. Величины M и σ, приведенные в данной таблице, свидетельствуют о нестабильном характере работы метода. При фиксированных значениях величин m и n время решения задачи сильно зависит от входных данных и колеблется в широких пределах.

В табл. 3 приведены результаты испытаний метода случайного поиска с локальной оптимизацией, описанного в разделе 3. Для каждой пары значений m и n решено сто задач ОГП. Для каждой задачи методом ветвей и границ вычислялось точное оптимальное значение целевой функции t^*. Затем эта же задача решалась методом случайного поиска с локальной оптимизацией.

Пусть C_0 – количество шагов, выполненных методом случайного поиска с локальной оптимизацией в процессе решения задачи ОГП, т. е. количество выполненных шагов 1 данного метода. Пусть C_1– суммарное количество выполненных алгоритмом локальной оптимизации A_0 шагов 1 этого метода в процессе решениязадачи ОГП. Далее символами E_0, M_0, σ_0 обозначены соответственно среднее, максимальное значение и стандартное отклонение величины C_0, а символами E_1, M_1, σ_1 обозначены аналогичные характеристики величины C_1.

Данные табл. 3 показывают, что для получения оптимального решения методу случайного поиска с локальной оптимизацией достаточно небольшого количества итераций. Количество использованного процессорного времени по сравнению со временем метода ветвей и границ незначительно. Стабильность работы этого метода по сравнению с методом ветвей и границ выше, так как величины σ_i / E_i меньше величин σ / E из табл. 2.

Опишем еще один вычислительный эксперимент. Случайным образом генерировались задачи ОГП при условии $m = 2n$. После этого для каждой задачи координаты n преследователей полагались равными координатам целей, так что оптимальное значение целевой функции каждой задачи ОГП оказывалось равным нулю. Задачи решались методом случайного поиска с локальной оптимизацией. Для каждой пары значений m и n решено сто задач ОГП. Отличие от предыдущего эксперимента заключается в больших значениях чисел m и n, при которых решить задачу методом ветвей и границ не удается. Результаты приведены в табл. 4. Данные табл. 4 говорят о стабильной работе метода при указанных размерностях задач. Во всех случаях найдено оптимальное решение. Среднее время решения одной задачи при условиях $m = 400$, $n = 200$ равно 1,0 сек.

Из результатов экспериментов можно сделать вывод о высокой точности и скорости метода случайного поиска с локальной оптимизацией даже при значительных размерностях решаемых задач ОГП.

Таблица 1. Зависимость количества допустимых решений задачи ОГП от чисел m и n

№	m	n	N_1	N_2	N_3
1	10	5	30240	113400	824040
2	12	6	665280	7484400	5.5×10^{7}
3	14	7	1.7×10^{7}	6.8×10^{8}	5.0×10^{9}
4	16	8	5.1×10^{8}	8.1×10^{10}	6.0×10^{11}
5	18	9	1.7×10^{10}	1.2×10^{13}	9.2×10^{13}
6	20	10	6.7×10^{11}	2.3×10^{15}	1.7×10^{16}
7	100	50	3.0×10^{93}	8.2×10^{142}	6.1×10^{143}

Таблица 2. Зависимость времени решения задачи ОГП методом ветвей и границ от чисел m и n

№	m	n	E	M	σ	S
1	12	6	0.2	8.3	0.9	1069
2	13	6	0.4	13.6	1.5	1225
3	13	7	1.0	27.7	3.6	4072
4	14	7	4.5	108.3	15.7	13262

Таблица 3. Зависимость количества итераций метода случайного поиска с локальной оптимизацией от чисел m и n

№	m	n	E_0	M_0	σ_0	E_1	M_1	σ_1
1	12	6	1.3	6	0.8	10.4	50	7.9
2	13	6	2.2	17	2.5	18.1	150	22.9
3	13	7	1.9	15	2.2	18.3	196	26.6
4	14	7	1.5	17	1.8	15.8	217	22.2

Таблица 4. Зависимость трудоемкости метода случайного поиска с локальной оптимизацией от чисел m и n

№	m	n	E_0	M_0	σ_0	E_1	M_1	σ_1
1	100	50	2.2	12	2.2	501.0	2493	452.1
2	200	100	5.0	32	5.4	2650.2	15573	2748.7
3	400	200	38.1	275	51.0	46769.2	327643	61905.4

5. Применение оптимизации групп преследования в программной системе ”Навигация”

Программная система ”Навигация” предназначена для моделирования поведения множества реактивных агентов в сложной динамической среде. В частности, система позволяет решать задачи оптимизации преследования на плоскости, в том числе оптимизировать состав групп преследования. Приведем пример вычисления оптимальных траекторий и групп преследования. Предположим, имеется четыре преследователя и два убегающих. Скорости преследователей превышают скорости убегающих. Считаем, что в каждой группе количество преследователей должно быть не меньше одного и не больше двух. Следует решить задачу ОГП в форме (10), (11) при условиях $m=4$, $n=2$, $k_j^{(1)}=1$, $k_j^{(2)}=2$, $j=1,2$.

На рисунке два убегающих расположены выше четырех преследователей. Оптимальные траектории игроков принадлежат прямым линиям. В каждой группе объекты двигаются с максимальными скоростями по направлению к точке X^*, которая является наиболее удаленной от убегающего игрока точкой в пересечении кругов Аполлония (см. раздел 1).

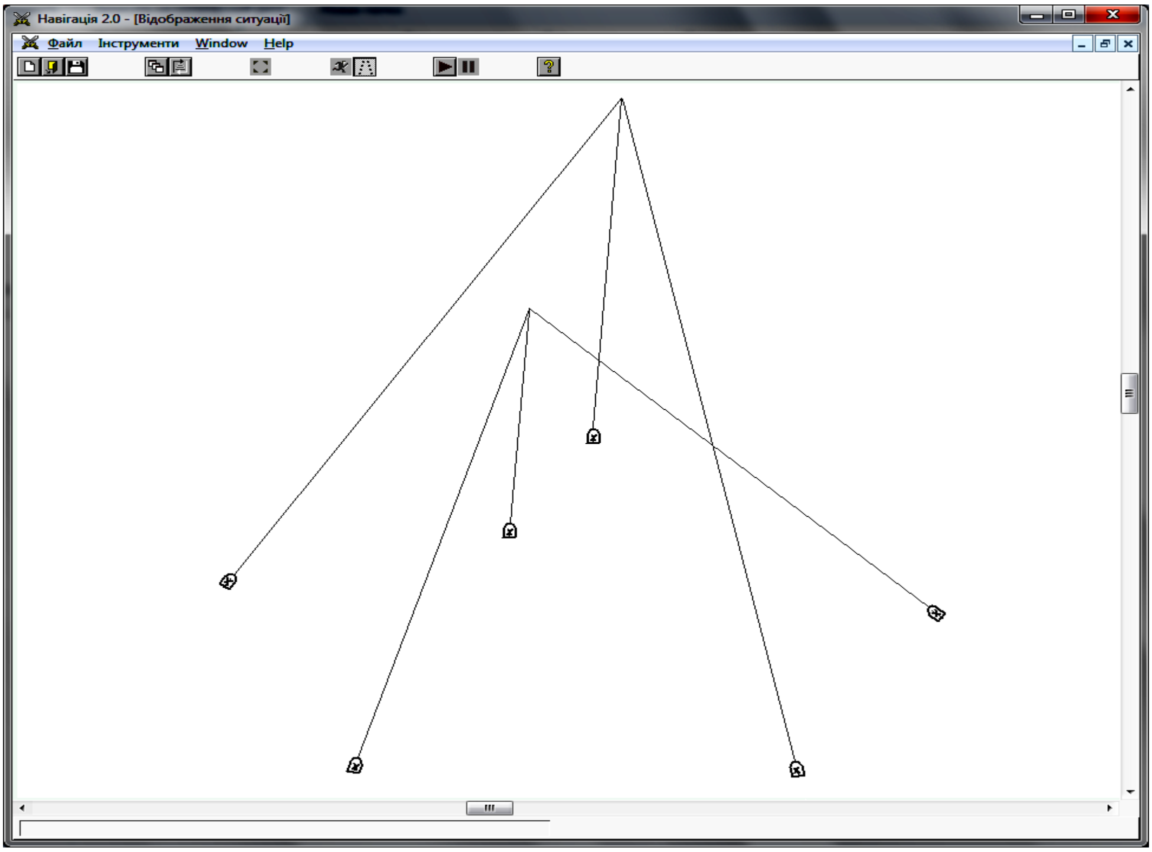

Рисунок. Оптимальные траектории преследования и убегания

Заключение

В данной работе рассмотрены игры преследования на плоскости с простым движением, в которых принимают участие несколько преследователей и убегающих. Считается, что количество преследователей больше числа убегающих, и скорости преследователей больше скоростей убегающих.

Для захвата целей множество преследователей разбивается на группы, причем для каждого убегающего создается одна группа. После захвата цели все преследователи группы и убегающий выбывают из игры. Считается, что в каждой группе игроки используют оптимальные стратегии. В качестве критерия используется время захвата.

Приведена теорема о NP-трудности задач оптимизации групп преследования. Эти задачи являются NP-трудными в сильном смысле уже в случае, когда для каждого убегающего необходимо назначить ровно двух преследователей. На основе этой теоремы сделаны выводы о том, какие методы следует применять для решения задачи оптимизации групп преследования.

Описаны методы ветвей и границ и случайного поиска с локальной оптимизацией. Выполнены численные эксперименты, на основании которых сделаны выводы о высокой скорости и точности метода случайного поиска с локальной оптимизацией для рассматриваемых задач.

1. *Айзекс Р.* Дифференциальные игры. – М.: Мир, 1967. – 480 с.
2. *Marcos A. M. Vieira and Ramesh Govindan and Gaurav S. Sukhatme*. "Scalable and Practical Pursuit-Evasion with Networked Robots" // Journal of Intelligent Service Robotics. Special Issue on Networked Robots. – 2009. – N 2. – P. 247–263.
3. *Пашко С.В.* Сложность задач оптимизации преследования на плоскости // Проблемы управления и информатики. – 2013. – № 3. – С. 27–39.
4. *Чикрий А.А.* Конфликтно управляемые процессы. – Киев: Наук. думка, 1992. – 384 с.
5. *Иванов Р.П., Ледяев Ю.С.* Оптимальность времени преследования в дифферен-

циальной игре многих объектов с простыми движениями // Труды МИАН СССР. – 1981. – 158. – С. 87–97.
6. *Рихсиев Б.Б.* Дифференциальные игры с простыми движениями. – Ташкент: ФАН, 1989. – 232 с.
7. *Ибрагимов Г.И., Рихсиев Б.Б.* О некоторых достаточных условиях оптимальности времени преследования в дифференциальной игре со многими преследующими // Автоматика и телемеханика. – 2006. – № 4. – С. 16–24.
8. *Петросян Л.А., Томский Г.В.* Геометрия простого преследования. – Новосибирск: Наука, 1983. – 140 с.
9. *Пашко С.В.* Квазиоптимальные стратегии в дифференциальных играх преследования на плоскости // Проблемы управления и информатики. – 2012. – № 6. – С. 30– 43.
10. *Пападимитриу Х., Стайглиц К.* Комбинаторная оптимизация // Алгоритмы и сложность. – М.: Мир, 1985. – 510 с.

Получено 04.02.2013

Об авторах:

Пашко Сергей Владимирович,
кандидат физико-математических наук,
старший научный сотрудник,

Яловец Андрей Леонидович,
доктор технических наук,
заместитель директора института.

Место работы авторов:

Институт программных систем
НАН Украины,
03187, Киев-187,
проспект Академика Глушкова, 40.

Тел.: (044) 526 6025,
E-mail: pashko55@yahoo.com

Тел.: (044) 526 1538,
E-mail: yal@isofts.kiev.ua