

УДК 004.942

Итерационное решение задачи покрытия территории

*Малышева Д.В., бакалавр
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Программное обеспечение ЭВМ и информационные технологии»*

*Научный руководитель: Горин С.В., к.т.н, доцент
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана
Bauman@bmstu.ru*

1. Введение

Одним из вариантов задачи для группового управления мобильными роботами является задача оптимального покрытия территории. Например, это задача обработки поверхности в зоне каких-либо поражений, когда нет доступа для человека, например, в зонах химического или радиоактивного поражения, или задача картографирования и создания модели какой-либо поверхности, с заранее неизвестной конфигурацией, такой, что нет возможности заранее составить программу обработки, или, наконец, это может быть задача поражения сил противника во время военных действий. Одной из целей в данном случае будет покрытие заданной территории без пропусков и с минимальными затратами на перемещение.

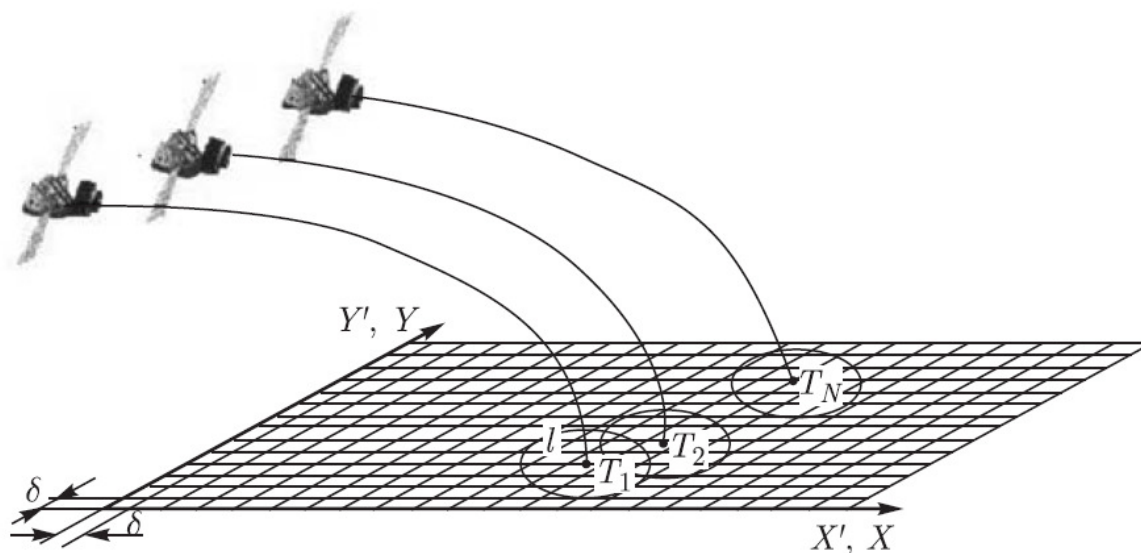


Рис. 1. Задача картографирования местности

2. Математическая постановка задачи

Для формализации вопроса необходимо ввести величину, характеризующую общую площадь пересечения зон покрытия всех роботов группы. Эта величина должна быть минимальна. Определяется она следующим образом:

$$Y_c = \sum_{i=1}^N \sum_{j=i+1}^N F(T_i, T_j)$$

Здесь предполагается, что имеется группа роботов R из N роботов R_j , каждый из которых способен обрабатывать площадь $s_j = \pi l^2$, где l – радиус области, покрываемой одним роботом.

Далее в данной формуле $T_i \in S$ целевая точка, точка попадания робота R_i , а функция $F(T_i, T_j)$ определяет, насколько пересекаются зоны покрытия i -го и j -го роботов, при этом считается, что робот R_i попадает в точку $T_i \in S$, а робот R_j попадает в точку $T_j \in S$. Для функции $F(T_i, T_j)$ возможны три варианта, они представлены на рис. 2.

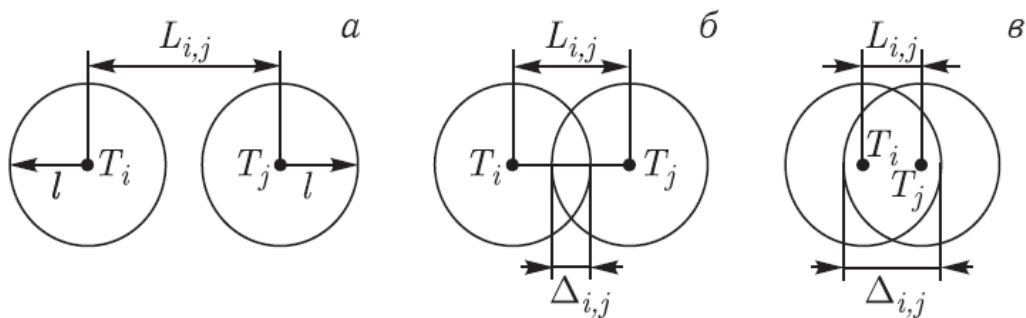


Рис. 2. Возможные пересечения зон покрытия двух роботов

Первый вариант возникает в случае, когда $L_{i,j} \geq 2l$, где $L_{i,j}$ – расстояние между точками T_i и T_j . В этом случае области, покрываемые роботами, вообще не пересекаются, поэтому $F(T_i, T_j) = 0$ (см. рис.2, а).

Второй вариант возникает, когда $l \leq L_{i,j} < 2l$. При этом значение функции $F(T_i, T_j)$ будет определяться величиной $\Delta_{i,j}$, которая в данном случае равна

$$\Delta_{i,j} = L_{i,j} - 2(L_{i,j} - l) = L_{i,j} - 2L_{i,j} + 2l = 2l - L_{i,j}$$

Другими словами, при $l \leq L_{i,j}$ пересечение областей $F(T_i, T_j) = \varphi(2l - L_{i,j})$, где функция $\varphi(2l - L_{i,j}) = \varphi(\Delta_{i,j})$ определяется выражением

$$\varphi(\Delta_{i,j}) = 2l^2 \arctg\left(\frac{\sqrt{2l\Delta_{i,j} - \Delta_{i,j}^2}}{2l - \Delta_{i,j}}\right) - \frac{1}{2}(2l - \Delta_{i,j})\sqrt{2l\Delta_{i,j} - \Delta_{i,j}^2}$$

Наконец, в третьем варианте $L_{i,j} < l$. При этом значение функции $F(T_i, T_j)$ также определяется выражением, где $\Delta_{i,j}$ в данном случае будет равна

$$\Delta_{i,j} = 2(l - L_{i,j}) + L_{i,j} = 2l - 2L_{i,j} + L_{i,j} = 2l - L_{i,j}$$

Условия отсутствия пропусков покрываемой поверхности можно получить на основе рис.3.

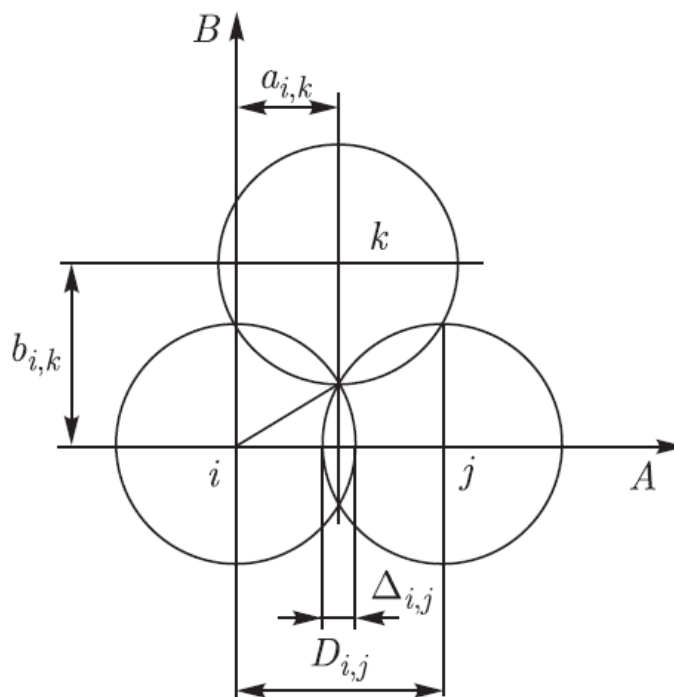


Рис. 3. Пересечение зон покрытия трех роботов

На рис.3 буквами i, j, k обозначены точки расположения роботов R_i, R_j, R_k .

Нетрудно видеть, что если хотя бы одно из расстояний $D_{i,k}, D_{i,j}$ и $D_{k,j}$ между роботами будет больше, чем показано на рисунке, появится пропуск покрываемой ими поверхности. Так как при $D_{i,j} < 2l$ отрезки $a_{i,k}$ и $b_{i,k}$ определяются выражениями

$$a_{i,k} = l - \frac{\Delta_{i,j}}{2} = 0,5D_{i,j}, \quad D_{i,j} < 2l,$$

$$b_{i,k} = l + \frac{1}{2}\sqrt{4l\Delta_{i,j} - \Delta_{i,j}^2}, \quad D_{i,j} < 2l,$$

то условие отсутствия пропусков покрываемой роботами R_i, R_j, R_k поверхности можно записать в виде

$$2l + \sqrt{4l\Delta_{i,j} - \Delta_{i,j}^2} \geq \sqrt{4D_{i,k}^2 - D_{i,j}^2}, \quad D_{i,j} < 2l,$$

где

$$D_{i,j} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2};$$

$$D_{i,k} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}$$

3. Внешняя среда

Внешняя среда представляется сеткой, ячейки (или элементарные участки) которой имеют размер $\delta \times \delta$ (м $\times$ м). Целочисленные размеры участка среды, в котором функционируют роботы, определяются выражениями

$$X = \left\lfloor \frac{X'}{\delta} - 1 \right\rfloor, Y = \left\lfloor \frac{Y'}{\delta} - 1 \right\rfloor,$$

где X' и Y' — размеры участка среды в метрах, а $\lfloor \rfloor$ — операция выделения целой части.

Ячейки могут быть доступными и недоступными для роботов. Только доступные ячейки могут быть выбраны роботами в качестве целевых. Каждая ячейка описывается номером i и дискретными координатами x_i и y_i по горизонтали и вертикали соответственно. При этом номер ячейки определяется по формуле:

$$i = y_i(X + 1) + x_i + 1 \quad (y_i = \overline{0, Y}, x_i = \overline{0, X})$$

Это позволяет по номеру ячейки i определить ее координаты x_i и y_i . Все ячейки среды образуют множество T мощностью $M = (X + 1)(Y + 1)$. Множество ячеек T состоит из подмножества доступных ячеек $T^s \subset T, T_k^s \in T^s \ (k = \overline{1, K})$, подмножества недоступных целей (вначале это могут быть ячейки с препятствиями) $T^d \subset T, T_l^d \in T^d \ (l = \overline{1, L})$ и подмножества точек, выбранных в качестве целевых точек (центров зон покрытия) $T^c \subset T, T_m^c \in T^c \ (m = \overline{1, M})$ (в начальный момент времени $T^c = \emptyset$). Причем $T^s \cup T^d \cup T^c \in T$ и $T^s \cap T^d \cap T^c = \emptyset$. Кроме того, в подмножество T^c могут входить только ячейки, принадлежавшие ранее подмножеству T^s . Поэтому ячейки, кроме номера i и координат x_i, y_i имеют дополнительный признак p_i , значения которого определяются выражением:

$$p_i = \begin{cases} 1, & \text{если } T_i \in T^s, \\ -1, & \text{если } T_i \in T^d, \\ 0, & \text{если } T_i \in T^c. \end{cases}$$

Таким образом, состояние каждой ячейки среды описывается четверкой $\langle i, x_i, y_i, p_i \rangle$.

4. Математическая модель роботов

Состояние робота можно определить четверкой $\langle j, x_j, y_j, q_j \rangle$, где x_j, y_j – координаты ячейки, в которой находится робот $R_j \in R$, а q_j – признак выбора роботом какой-либо ячейки в качестве целевой точки, который определяется следующим образом:

$$q_j = \begin{cases} 0, & \text{если робот } R_j \text{ выбрал целевую точку} \\ 1, & \text{в противном случае} \end{cases}$$

Каждый робот имеет два списка: список всех ячеек среды T , элементами которого являются четверки $t_i = \langle i, x_i, y_i, p_i \rangle, i = \overline{1, M}$, и список роботов группы, элементами которого являются четверки $R_j = \langle j, x_j, y_j, q_j \rangle, j = \overline{1, N}$, где $M = XY + X + Y + 1$.

5. Итерационная процедура оптимизации

Эффективность решения задачи покрытия поверхности во многом зависит от «точки привязки», т.е. от самой первой целевой точки. Можно использовать несколько вариантов начала выбора роботами своих целевых точек (последовательный выбор каждым роботом ячейки в радиусе обзора, одновременный выбор всеми роботами и проч.), однако наиболее оптимальным вариантом является выбор ячейки в центре области, занимаемой роботами (рис.4), координаты которой определяются следующим образом:

$$x_c = \left\lfloor \frac{\left(\max_{j=\overline{1,N}}(x_j) + \min_{j=\overline{1,N}}(x_j) \right)}{2} \right\rfloor,$$

$$y_c = \left\lfloor \frac{\left(\max_{j=\overline{1,N}}(y_j) + \min_{j=\overline{1,N}}(y_j) \right)}{2} \right\rfloor$$

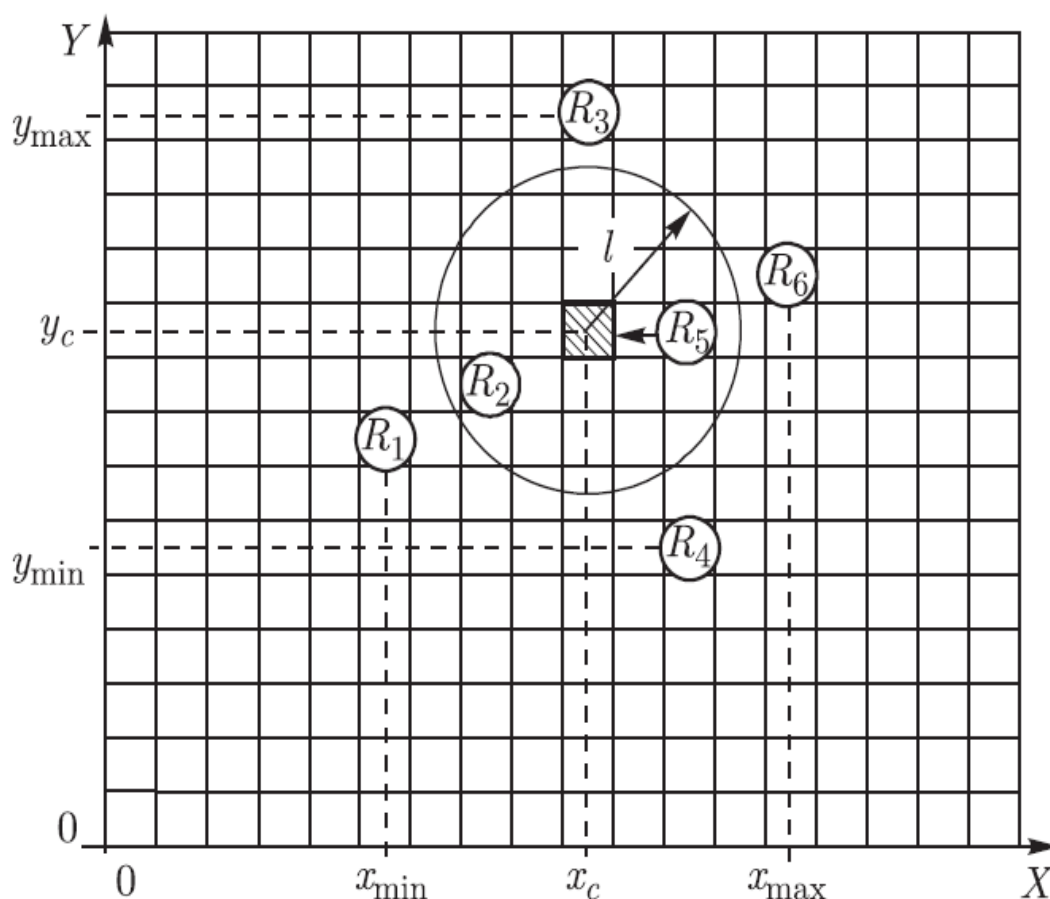


Рис. 4. Определение "первой цели"

Каждый робот $R_j \in R(j = \overline{1, N})$ передает свои координаты (x_j, y_j) всем остальным роботам группы. По формуле определения «первой цели» всеми роботами определяются координаты x_c, y_c . Если ячейка $T(x_c, y_c) \in T^d$ (имеющая признак $p_{i*} = 1$), т.е. все роботы полагают $x_c = x_{i*}, y_c = y_{i*}$. Далее всеми роботами определяется оценка эффективности выбора ими первой цели $(i^*, x_c, y_c - 1)$ в качестве целевой точки. Эта оценка $d_{jc}^{(1)}$ может зависеть от расстояния между роботом и первой целью, следующим образом:

$$d_{jc}^{(1)} = \begin{cases} 1 - \frac{D_{jc}^{(1)}}{D_{max}}, & \text{если } D_{jc}^{(1)} < D_{max} \\ 0, & \text{в противном случае} \end{cases}$$

где D_{max} - максимальное расстояние, на которое могут перемещаться роботы группы за один шаг, может также определяться запасом энергоресурса либо скоростью движения роботов и отведенным для достижения цели времени;

$D_{jc}^{(1)} = \left\lfloor \sqrt{(x_c - x_j)^2 + (y_c - y_j)^2} \right\rfloor, j = \overline{1, N}; D_{jc}^{(1)}$ – расстояние от j -го робота до первой цели.

Затем каждый робот передает свое значение $d_{jc}^{(1)}$ всем остальным роботам группы.

Таким образом, у каждого робота имеется массив оценок эффективности. Первая цель, согласно алгоритму, закрепляется за роботом, имеющим максимальное значение оценки эффективности $d_{jc}^{(1)}$ и, следовательно, за роботом, наиболее близким к первой цели.

После того, как какой-либо робот R_j выбрал целевую точку $\langle j_1, x_{j_1}, y_{j_1}, 0 \rangle$, роботы, определяют выбранные роботом R_j ячейки. Для оставшихся доступными ячеек $T_i \in T$ остальные роботы определяют элементы d_{ji} матрицы D , т.е. оценки эффективности, которые они внесут в достижение групповой цели, если выберут эти ячейки в качестве целевых точек по следующей формуле:

$$d_{ji} = p_i \cdot q_j \cdot d_{ji}^{(1)} \cdot d_i^{(2)}$$

где $d_{j,i}^{(1)}$ – оценка, зависящая от расстояния от робота до ячейки, определяемая по формуле, приведенной выше; $d_i^{(2)}$ – оценка, зависящая от расстояния между i ячейкой и ближайшей выбранной ранее, определяется следующим образом:

$$d_i^{(2)} = \begin{cases} 1 - \frac{|2r - D_i^{(2)}|}{2r}, & \text{если } D_i^{(2)} \leq 2r, \\ 1 - \frac{D_i^{(2)}}{D_{max}}, & \text{если } 2r < D_i^{(2)} \leq D_{max}, \\ 0, & \text{если } D_i^{(2)} \geq D_{max} \end{cases}$$

Величина $D_i^{(2)}$ определяется как расстояние между ячейкой T_i , для которой рассчитывается оценка $d_i^{(2)}$ и ближайшей целевой точкой $T_h \in T^c$, т.е.

$$D_i^{(2)} = \min_{T_h \in T^c} \left(\left| \sqrt{(x_h - x_i)^2 + (y_h - y_i)^2} \right| \right)$$

После составления матрицы D начинается работа алгоритма: последовательно роботы делают выбор следующей целевой точки – в строке, соответствующей своему номеру отыскивается максимальное положительное значение, фиксируется этот номер столбца i_1 . Если значений несколько, то выбирается одно из них. Затем максимальный элемент отыскивается в том столбце i_1 , если номер строки совпадает с номером текущего робота, то цель выбрана оптимально, она утверждается роботом, и информация отправляется всем остальным участникам группы. В соответствии с алгоритмом все роботы после этого обнуляют (или полагают отрицательными) все значения в строке, соответствующей роботу, пославшему сообщение, это означает, что он больше не участвует в выборе. Кроме того, обнуляются значения в столбце i_1 , что означает, что цель больше не участвует в распределении. В случае, если номер строки с максимальным элементом столбца не совпадает с номером робота, делающего выбор, то робот просто отказывается от выбора цели в текущем итерационном цикле, о чем информирует остальных роботов группы. Работа алгоритма продолжается до тех пор, пока все доступные для роботов ячейки не будут распределены между ними.

6. Вывод

Таким образом, алгоритмы распределения целей на основе итерационной процедуры оптимизации коллективных действий группы роботов позволяют значительно повысить эффективность решения задачи покрытия площадей в сравнении со стандартными алгоритмами решения подобных задач, такими как «жадный» алгоритм и алгоритм случайного выбора целей в определенном радиусе.

Список литературы

1. Каляев И.А., Гайдук А.Р., Капустян С.Г. Модели и алгоритмы коллективного управления в группах роботов. Москва: ФИЗМАТЛИТ, 2009. 280 с.
2. Shoham Y. Agent Oriented Programming // Journal of Artificial Intelligence. 1993. № 60 (1). P. 51-92.
3. Юревич Е.И. Управление роботами и робототехническими системами: учеб. пособие. СПб.: Изд-во СПбГТУ, 2001. 168 с.