

УДК 004.421.023

В. Ю. КУЗНЕЦОВ

ЗАДАЧИ ПОКРЫТИЯ ОРТОГОНАЛЬНЫХ МНОГОУГОЛЬНИКОВ С ЗАПРЕТНЫМИ УЧАСТКАМИ

Рассматриваются две задачи покрытия многоугольных областей объектами произвольной формы и ее приложения. Обе задачи являются NP-трудными и для решения предложены эвристические алгоритмы. Проведены численные эксперименты, представлены выводы на основе результатов экспериментов.

Методы оптимизации ; задача покрытия

ВВЕДЕНИЕ

Задачи покрытия представляют собой важный подраздел задач раскроя-упаковки. Они являются достаточно новыми. Впервые задача максимального покрытия была представлена в 2002 г. Е. Заком [1]. В 2004 г. А. С. Мухачева предложила некоторые эвристические подходы к решению задачи максимального покрытия [4]. В настоящей работе для решения этой задачи применяются эволюционные стратегии в сочетании с эвристиками, предложенными А. С. Мухачевой.

В [5] рассматривается задача редчайшего покрытия плоскости равными кругами. А дальше в той же книге [5] рассматривается задача покрытия выпуклой области минимальным числом равных кругов. В [2] обсуждаются вопросы, связанные с оценкой качества полученного решения. В настоящей статье рассматривается модификация задачи покрытия выпуклой области равными кругами с учетом обхода препятствий. В этой постановке в литературе она пока не была приведена. Кроме того, в этой статье рассматриваются различные практические применения. Рассматриваемая задача является NP-трудной, поэтому для ее решения предлагаются различные метаэвристики и приближенные методы. Один из таких методов рассматривается в данной статье.

1. ЗАДАЧА О МИНИМАЛЬНОМ ПОКРЫТИИ ПРЯМОУГОЛЬНОЙ ОБЛАСТИ С ЗАПРЕТНЫМИ УЧАСТКАМИ

Дана прямоугольная область A ширины W и длины L . Введем прямоугольную систему координат так, чтобы оси Ox и Oy совпали соответственно с нижней и левой сторо-

нами этой области. На области A определено множество Z запретных зон. Зоны заданы прямоугольниками, стороны которых параллельны граням области: $Z = \bigcup_{i=1}^n z_i$, где $z_i = \langle (x_i, y_i), (w_i, l_i) \rangle$ — прямоугольник с номером i , в котором (x_i, y_i) — координаты нижнего левого угла, (w_i, l_i) — ширина и длина прямоугольника. Задан радиус r . Требуется покрыть фигуру $A \setminus Z$ наименьшим числом кругов радиуса r . Центры кругов должны лежать в области $A \setminus Z$.

Приведем некоторые интерпретации этой задачи.

Проектирование технологических установок. Производственные установки по сбору и переработке газа представляют собой совокупность несвязанного оборудования, размещенного на некоторой территории (полигоне). Для контроля возможной утечки используются газоанализаторы, их монтируют на территории установок. Они обладают круговой зоной действия. Требуется минимизировать количество газоанализаторов при условии покрытия зонами их действия всего полигона и размещения в обход запрещенных участков.

Размещение базовых станций сотовой связи. Требуется обеспечить сотовой связью определенную область, например, город. Сотовая связь доступна внутри «зоны видимости» базовой станции, ее можно моделировать кругом заданного радиуса действия. Требуется разместить наименьшее число базовых станций в обход естественных преград (рек, дорог и т. д.).

Поиск пути для персонажей в компьютерных играх. Имеется трехмерный ландшафт

сложной формы. На ландшафте могут размещаться препятствия. По ландшафту перемещаются персонажи, управляемые компьютером (ИИ, искусственный интеллект). Нужно разработать структуру данных и построить алгоритм, которые позволят в реальном времени прокладывать характерный для человека маршрут между любыми двумя точками ландшафта. Есть общепринятый подход к решению задачи поиска пути, основанный на вейпоинтах (waypoint). Создание карты вейпоинтов сводится к задаче покрытия произвольной области кругами заданного радиуса: карту можно разбить на зоны так, чтобы произвольная вертикальная прямая пересекала ландшафт не более чем в одной точке; препятствия — запретные зоны, центры кругов — вейпоинты, а ребра графа строятся по принципу — если два круга пересекаются, то между вершинами, образованными этими кругами, строится ребро. Автоматическая генерация карт вейпоинтов значительно ускоряет разработку компьютерных игр.

Передача информации. Квантование данных, полученных от непрерывно изменяющегося источника, тесно связано с задачей о минимальном покрытии пространства перекрывающимися шарами. Например, чтобы выполнять квантование в двумерном пространстве, входные данные объединяют в пары и каждую пару рассматривают как координаты точки на плоскости. Затем каждую точку A , отвечающую паре данных, округляют до ближайшей отмеченной точки (B), лежащей в той же заранее выбранной области плоскости, что и A . Проблема состоит в том, чтобы выбрать разбиение плоскости и отмеченные точки таким способом, при котором была бы минимальной средняя ошибка квантования. Если данные распределены равномерно, а отмеченные точки лежат в центрах квадратов, то средняя ошибка равна $1/12$. При помощи минимального покрытия плоскости кругами можно получить лучший способ квантования (см. рис. 1). Такое покрытие порождается гексагональной упаковкой (т. е. упаковкой, когда каждого круга касается еще 6 кругов) при увеличении радиусов кругов настолько, чтобы каждая точка плоскости оказалась внутри или на границе хотя бы одного круга. Радиус R кругов покрытия равен расстоянию от центра до самой «глубокой ямы» в соответствующей упаковке. Если соединить эти «глубокие ямы» подходящими прямыми, то плоскость разобьется на правильные шестиугольники. Выбор отмеченных точек в центрах правильных шестиугольников при квантовании рав-

номерно распределенных данных дает среднюю ошибку $\frac{5\sqrt{3}}{108}$, т. е. около 0,0802, что немного меньше, чем $\frac{1}{12}$.

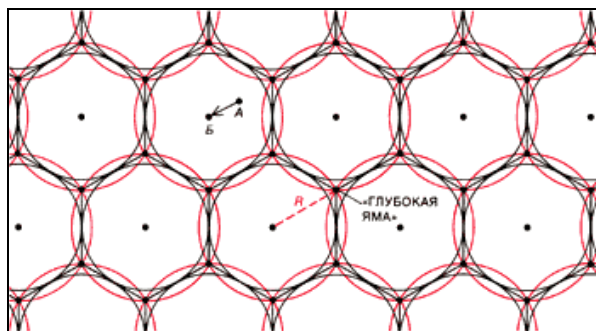


Рис. 1. Задача покрытия в теории информации

В [1] показано, что эта задача NP-трудная.

Допустимое решение задачи может быть получено в результате замены кругов квадратами с центрами в $A \setminus Z$. Это решение далее будет использовано в качестве верхней границы функции цели и для оценки некоторых характеристик. Покрытие осуществляется с использованием блок-структур (BS) [3]. Технология блочных структур была разработана А. С. Филипповой для решения задач раскроя-упаковки. В данной работе предлагается использовать модификацию блок-структур для решения задачи покрытия.

Пусть имеется схема частичного покрытия: запретные зоны и часть размещенных квадратов. Через вертикальные стороны фигур проводим прямые. Таким образом, область разбивается на вертикальные блоки.

Блок-структура BS — это последовательность блоков различной длины и одинаковой ширины. $BS = (B_1, B_2, \dots, B_n)$, где B_i — блок с номером i . Блок — совокупность объектов или их фрагментов заданной ширины. $B_i = ((i_1, i_2, \dots, i_j, \dots, i_m) L_i)$, где i_j — j -й элемент i -го блока, L_i — длина блока. В блоке могут размещаться элементы трех типов: запретные участки (Z), покрывающие прямоугольники (D), и свободные участки (F). При этом j -й элемент задается парой (T_j, W_j) , где T_j — тип элемента (Z, F или D), W_j — ширина j -го элемента.

По блок-структуре можно вычислить координаты любого элемента. Пусть требуемый элемент находится в i -м блоке в позиции j , тогда $x = \sum_{k=1}^i l_k$, $y = \sum_{k=1}^j w_k$, где l_k — длина k -го блока, w_k — ширина k -го элемента в i -м блоке.

При использовании блок-структур задача обхода запретных участков сводится к по-

иску первого попавшегося F -элемента. Вся область будет покрыта тогда, когда в блок-структуре не окажется ни единого F -элемента.

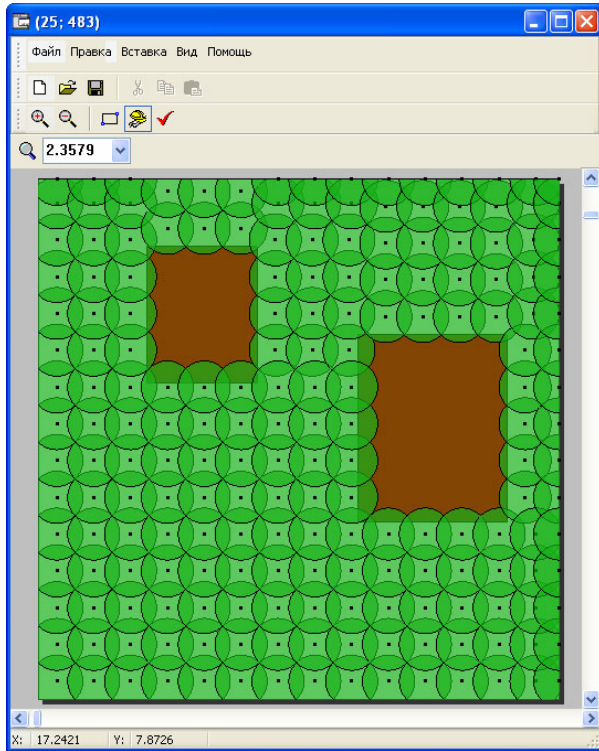


Рис. 2. Пример решения задачи покрытия при помощи блочного алгоритма

Алгоритм покрытия состоит в выполнении следующих шагов:

1. *Инициализация.* Блок-структура состоит из единственного блока ширины W (ширина покрываемой области) и содержит единственный пустой (F) элемент длины L (длина покрываемой области);

2. *Размещение запретных участков.* В блок-структуру помещаются все запретные участки;

3. *Радиус.* Радиус покрывающих кругов равен R . Сторона вписанного квадрата $a = R\sqrt{2}$;

4. Если в блок-структуре нет F -элементов, то перейти к (8);

5. *Координаты.* Получить координаты (X_f, Y_f) F -элемента;

6. *Вставка.* Поместить квадрат S_i в позицию (X_f, Y_f) ;

7. Перейти к (4);

8. *Построение кругов.* Для каждого квадрата S_i построить круг радиуса R с координатами центра $O_i = \left(X_i + \frac{\sqrt{2}}{2}R, Y_i + \frac{\sqrt{2}}{2}R \right)$,

где (X_i, Y_i) — координаты левого нижнего угла квадрата S_i .

9. Конец алгоритма.

Для проведения численного эксперимента разработана программа, реализующая блочный алгоритм покрытия. На рис. 2 приведен пример решения задачи покрытия, найденного при помощи этой программы.

Алгоритм был испытан на различных случайных задачах. Каждая задача описывается набором параметров. Обозначим параметры задачи через $\mathbf{P}_i, i = \overline{1, n}$, где n — количество параметров, \mathbf{P}_i — множество возможных значений i -го параметра. $\mathbf{P} = \mathbf{P}_1 \times \mathbf{P}_2 \times \dots \times \mathbf{P}_n$ — пространство задач. Вектор $p \in \mathbf{P}$ — индивидуальная задача. Количество индивидуальных задач бесконечно, а число измерений пространства \mathbf{P} — велико, поэтому на практике точно определить качество работы алгоритма невозможно. Для идентификации эффективности алгоритма выполним следующие действия.

Выберем небольшое подмножество $\mathbf{PP} \subset \mathbf{P}$ существенных параметров задачи;

Каждый вектор $p \in \mathbf{PP}$ будем рассматривать как проекцию множества сходных задач из \mathbf{P} на пространство \mathbf{PP} ;

Будем вычислять значения характеристик не во всех векторах \mathbf{PP} , а только в узлах решетки, равномерно покрывающей \mathbf{PP} .

Были выбраны следующие параметры: длина и ширина покрываемой области; минимальные длина и ширина запретных участков; максимальные длина и ширина запретных участков; радиус покрывающих кругов: измеряется в % от максимального из размеров покрываемой области; заполненность: $F = \frac{S_Z}{S_A}$, где S_Z — площадь фигуры \mathbf{Z} ($\mathbf{Z} = \bigcup_{i=1}^k Z_i$, где k — количество запретных участков, Z_i — i -й запретный участок), S_A — площадь покрываемой фигуры A .

Для оценки качества работы алгоритма используется фактор аппроксимации $FA = \frac{N}{N_{opt}}$, где N — количество кругов в покрытии, N_{opt} — количество кругов в оптимальном покрытии. Фактор аппроксимации сложно вычислить, поскольку для вычисления требуется знать точное решение задачи. Однако нижнюю оценку FA легко получить, заменяя N_{opt} оценкой N'_{opt} минимального числа кругов, $N'_{opt} = \left\lceil \frac{S_A K}{S} \right\rceil$, где S_A — площадь фигуры $A \setminus Z$, $K = 1,22$ — коэффициент, учитывающий перекрытие кругов в гексаго-

Таблица 1

Результаты численного эксперимента

Покрываемая область		Запретные участки, %				Радиус кругов, %	Заполненность, %	Фактор аппроксимации
Длина	Ширина	Минимум		Максимум				
		Длина	Ширина	Длина	Ширина			
50	50	1	1	50	50	10	10	1,72
50	50	1	1	50	50	50	10	2
50	50	1	1	50	50	10	70	2,84
50	50	1	1	50	50	50	70	3,29
50	1000	1	1	50	50	10	10	3,01
50	1000	1	1	50	50	50	10	2
50	1000	1	1	50	50	10	70	3
50	1000	1	1	50	50	50	70	2
1000	50	1	1	50	50	10	10	2,99
1000	50	1	1	50	50	50	10	2
1000	50	1	1	50	50	10	70	3
1000	50	1	1	50	50	50	70	2
1000	1000	1	1	50	50	10	10	1,7
1000	1000	1	1	50	50	50	10	2
1000	1000	1	1	50	50	10	70	2,75
1000	1000	1	1	50	50	50	70	2,79

нальном покрытии, S — площадь круга. Количество тестов на каждый вектор p — 100.

Предложенная методика тестирования алгоритмов реализована в разработанной программе. На рис. 3 показана форма, в которой задаются параметры оценки качества алгоритмов.

Рис. 3. Форма настройки параметров тестирования алгоритма

Результаты эксперимента приведены в табл. 1, они позволяют сформулировать следующие выводы:

1. Фактор аппроксимации адекватен качеству полученного покрытия;
2. С ростом заполненности возрастает фактор аппроксимации. Это может быть вызвано как неточностью оценки фактора аппроксимации, так и ограничениями блочной технологии;
3. Если радиус кругов намного меньше покрываемой области, то алгоритм находит решения с фактором аппроксимации, близким к 2. Потери в качестве решения вызваны переходом к покрытию квадратами.

2. ЗАДАЧА МАКСИМАЛЬНОГО ПОКРЫТИЯ ОДИНАКОВЫХ ЛИСТОВ ПРЯМОУГОЛЬНЫМИ ЗАГОТОВКАМИ

Задача максимального покрытия формулируется следующим образом: необходимо покрыть листы заданных размеров (ширина и длина) различными прямоугольными заготовками таким образом, чтобы количество полностью покрытых листов было максимально. Эта задача возникает, например, при покрытии полов в помещении линолеумом или паркетом. Форма полов задается покрываемой областью с запретами; листы линолеума — покрывающими объектами, то-

гда решением задачи покрытия будет схема покрытия пола.

Имеются листы заданной ширины W и заданной длины L и набор из m прямоугольных предметов заданных размеров $(w_i; l_i)$, $i = \overline{1, m}$, где w_i — ширина стороны i -го предмета, а l_i — длина стороны i -го предмета. Введем прямоугольную систему координат: оси Ox и Oy совпадают соответственно с нижней и левой боковой сторонами листа. Положение каждого прямоугольника P_i зададим координатами (x_i, y_i) его левого нижнего угла. Требуется найти максимальное количество N полностью покрытых листов на множестве допустимых покрытий, при этом любая точка на листе принадлежит одному или нескольким покрывающим листам прямоугольникам. Для любой точки $(x, y) \in Q$, где Q — покрываемая область, найдется прямоугольник $P(x_i, y_i)$, $i = \overline{1, m}$, для которого справедливы условия: $x_i \leq x \leq x_i + l_i$ и $y_i \leq y \leq y_i + w_i$. Допустимое покрытие оптимальное, если общее число покрытых листов N достигает максимума.

В [1] показано, что данная задача относится к классу NP-трудных задач.

Для решения задачи выбран эволюционный алгоритм (1+1)-ЕА, рассмотренный в [3]. Особенность алгоритма в том, что популяция состоит из единственной особи, к которой на каждой итерации применяется оператор мутации.

Базовый алгоритм был адаптирован следующим образом:

- Особь — некоторое допустимое покрытие листов;

- Хромосома — приоритетный список прямоугольников;

- Популяция состоит из единственной особи;

- Условие останова — заданное число итераций. Чем больше число итераций, тем лучше будет получаемый результат и тем дольше будет работать алгоритм.

- Оператор мутации. Приоритетный список меняется случайным образом;

- Оператор селекции.

- Выбирается покрытие с наибольшим числом полностью покрытых листов (алгоритм (1+1)-ЕА I);

- Выбирается покрытие с наименьшим остатком. Остаток — сумма площадей частей прямоугольников, выходящих за пределы листа или накрывающих другие прямоугольники (алгоритм (1+1)-ЕА II).

- Оператор репродукции — покрытие листов блочным алгоритмом (предложен А. С. Мухачевой и описан в [4]) на основе приоритетного списка. Блочный алгоритм использует эвристики NF (следующий подходящий) и BF (лучший подходящий) [4].

Для анализа эффективности разработанных алгоритмов был выполнен численный эксперимент.

При этом были заданы параметры:

- $W = 100$ — ширина и $L = 100$ — длина листа (на результат решения не влияет, для всех задач одинакова);

- $m = 50, 100, 150, 200$ — размерность (количество укладываемых заготовок);

- w_i, l_i — размеры i -й заготовки, $i = \overline{1, m}$;

- u_1 — нижнее ограничение весов, т. е. $l_i, w_i \geq u_1 \cdot 100$;

- u_2 — верхнее ограничение весов, т. е. $l_i, w_i \leq u_2 \cdot 100$;

Численный эксперимент проводился для трех групп задач:

- группа задач, для которых $u_1 = 0,1, u_2 = 0,9$;

- группа задач, для которых $u_1 = 0,4, u_2 = 0,8$;

- группа задач, для которых $u_1 = 0,25, u_2 = 0,4$;

- при числе итераций, равном 1000.

Качество решения индивидуальной задачи оценивалось при помощи коэффициента покрытия:

$$K = \frac{S}{\sum s_i},$$

где K — коэффициент покрытия, S — суммарная площадь покрываемых листов, s_i — площадь i -го прямоугольника.

Результаты эксперимента отражены на рис. 4–6.

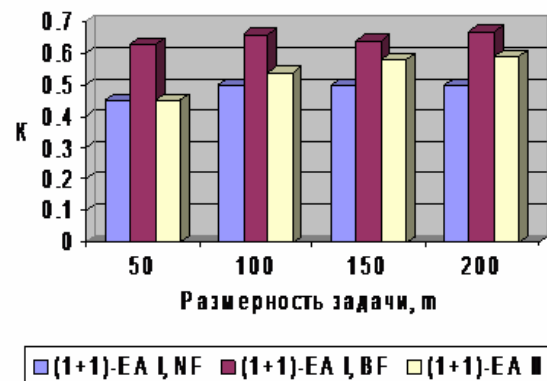
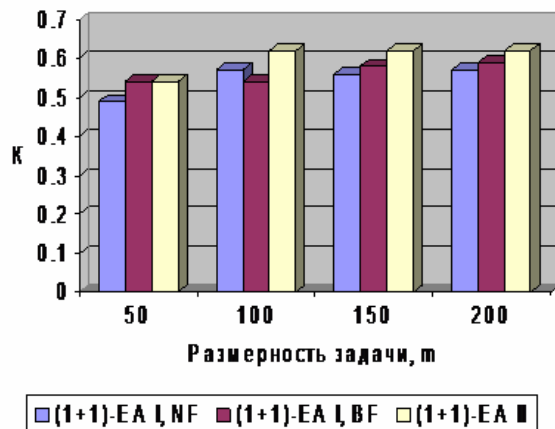
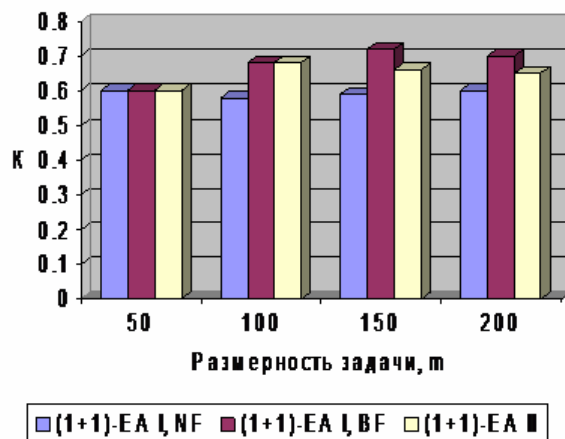


Рис. 4. $u_1 = 0,1, u_2 = 0,9$

Рис. 5. $u_1 = 0,4, u_2 = 0,8$ Рис. 6. $u_1 = 0,25, u_2 = 0,4$

Результаты эксперимента позволили сделать следующие выводы:

1) для разнородных заготовок лучшие результаты дает алгоритм (1+1)-EA I в сочетании с эвристикой BF, а для больших однородных заготовок лучше применять алгоритм (1+1)-EA II;

2) в целом, лучшие результаты показал алгоритм (1+1)-EA II, однако этот алгоритм работает дольше, чем (1+1)-EA I.

ЗАКЛЮЧЕНИЕ

Предложен и реализован метод покрытия области кругами с обходом препятствий, основанный на технологии блочных структур. Экспериментально показано, что полученные

решения оказываются хуже нижней границы не более чем в три раза. Учитывая, что нижняя граница является грубой, реальная оценка должна быть лучше. Намечены следующие пути повышения эффективности метода: разработать и включить в алгоритм процедуры сдвига фигур с ориентацией на их выталкивание из зоны покрытия; разработать блочную схему покрытия области кругами; распараллелить некоторые процедуры, например, вставки фигур в блок-структуру.

Эволюционные алгоритмы, применяемые для решения задачи покрытия одинаковых листов прямоугольными заготовками дают решения, близкие к оптимальному. Пользователь может изменять время работы эволюционных алгоритмов, получая более точное решение за счет увеличения времени обработки.

СПИСОК ЛИТЕРАТУРЫ

1. **Zak, E.** Relationship between skiving stock and cutting stock problem / E. Zak, C. Rennick // The 16th Triennial Conf. of the Int. Federation of Operational Research Societies. Edinburgh, UK, 2002. P. 27.
2. **Lovasz, L.** On the ratio of optimal integral and fractional covers / L. Lovasz // J. of Discrete Math. 1975. 13. P. 383–390.
3. **Филиппова, А. С.** Моделирование эволюционных алгоритмов решения задач прямоугольной упаковки на базе технологии блочных структур / А. С. Филиппова // Информационные технологии. 2006. Приложение. № 6. 31 с.
4. **Мухачева, А. С.** Простые эвристики для решения двумерной задачи максимального покрытия / А. С. Мухачева // Принятие решений в условиях неопределенности. Уфа, 2004. С. 38–43.
5. **Тот, Л. Ф.** Расположения на плоскости, на сфере и в пространстве / Л. Ф. Тот. М.: Физматлит, 1958.

ОБ АВТОРЕ



Кузнецов Вячеслав Юрьевич, асп. каф. выч. мат. и киб. Дипл. мат.-прогр. (УГАТУ, 2005). Готовит дис. в обл. задач покрытия, раскроя-упаковки.