УДК 519.112.71

# ОПТИМИЗАЦИЯ ПЕРЕДАЧИ ШИРОКОВЕЩАТЕЛЬНОЙ ИНФОРМАЦИИ В КОМПЬЮТЕРНЫХ СЕТЯХ

**Д. В. Ильинов, М. К. Чернышов, Г. Д. Чернышова**

*Воронежский государственный университет*

Сложная задача дискретной оптимизации формализуется в виде задачи о минимальном покрытии с дополнительными ограничениями. Предложены алгоритмы "жадного" типа для отыскания приближенного решения.

Рассматривается задача, возникающая в процессе организации массовых рассылок сообщений в компьютерных сетях крупных организаций. Предполагается существование большого количества подсетей, доступ к которым осуществляется при помощи компьютеров, обладающих свойствами маршрутизаторов, т.е. имеющими доступ как к серверу, обеспечивающему рассылку информации, так и к компьютерам в подразделениях организации, объединенным в подсети. При этом также предполагается, что сервер не имеет технической возможности получения прямого доступа ко всем компьютерам, находящимся в подразделениях. Требуется выбрать минимальное число маршрутизаторов, обеспечивающих передачу широковещательных сообщений всем объектам-компьютерам организации. Каждый объект должен получить сообщение. Считается, что $j$-ый маршрутизатор может обслуживать $i$-ую сеть, если он принадлежит данной сети. Исходные данные задачи можно представить в виде булевой матрицы $A = (a_{ij})$:

$$a_{ij} = \begin{cases} 1, \text{ если } j\text{ - й компьютер} \\ \quad \text{принадлежит } i\text{ - й сети}, \\ 0, \text{ в противоположном случае.} \end{cases}$$

Для математической формализации задачи переменные вводятся следующим образом: $x_{ij} = \begin{cases} 1 \\ 0 \end{cases}$, причем $x_{ij} = 1$, если $j$-й компьютер «обслуживает» $i$-ю сеть, и $x_{ij} = 0$ в противном случае. Условие $\sum\limits_{j=1}^{n} a_{ij} x_{ij} \geqslant 1,\ i = \overline{1,m}$ означает, что каждая сеть обслуживается. Технические возможности каждого маршрутизатора по количеству максимально возможного числа обслуживаемых им сетей заложены в условии $\sum\limits_{i=1}^{m} x_{ij} \leqslant d_j, = \overline{1,n}$. Стоит отметить, что в данной задаче матрица покрытий $A$ сильно разрежена, что вызывает дополнительные сложности при использовании точных алгоритмов.

Получена задача о минимальном покрытии с дополнительными ограничениями:

$$\sum_{j=1}^{n} \operatorname{sgn}\left(\sum_{i=1}^{m} x_{ij}\right) \to \min; \tag{1}$$

$$\sum_{j=1}^{n} a_{ij} x_{ij} \geqslant 1,\ i = \overline{1,m}; \tag{2}$$

$$\sum_{i=1}^{m} x_{ij} \leqslant d_j,\ j = \overline{1,n}; \tag{3}$$

$$x_{ij} = \begin{cases} 1 \\ 0 \end{cases}, i = \overline{1,m}, j = \overline{1,n}. \tag{4}$$

В работах [1], [2] предложены алгоритмы решения такой задачи, основанные на некоторых точных методах. В [1] за базовый взят симплекс-метод решения задачи линейного программирования. В [2] разработаны двойственные методы и их вероятностные аналоги. Алгоритмы такого рода предполагают возможность отыскания точного решения задачи. При этом метод, использующий симплекс-процедуру, предполагает дополнительно работу с алгоритмами переборного типа, что сопряжено с возможными большими временными затратами. Двойственные алгоритмы, рассматриваемые в работе [2], при использовании их для дискретной оптимизации имеют медленную сходимость. Останов по числу итераций таких алгоритмов не гарантирует допустимость полученной точки.

В настоящей работе предлагаются быстрые алгоритмы получения допустимых точек рассматриваемой задачи. Эти алгоритмы строятся с помощью корректирования данных, получаемых в результате решения задачи о минимальном покрытии (1), (2), (4) (без дополнительных ограничений) известными «жадными» алгоритмами (по максимуму мощности столбцов, по наиболее рискованному элементу) или точным (метод ветвей и границ). Эти результаты записываются в матрицу $B=(b_{ij})$ размера $m\times n$, формируемую по следующему правилу (ПФМ): пусть в итоге решения одним из указанных алгоритмов получен некий результат:

$$x_{i_1}, x_{i_2}, \dots, x_{i_k} = 1,\ x_j = 0,$$

$$j=\overline{1,n},\ j \neq i_1, i_2, \dots, i_k.$$

Таким образом в оптимальное покрытие должны быть включены столбцы $x_{i_1}, x_{i_2}, \dots, x_{i_k}$. Тогда в матрице $B$ столбцы с номерами $i_1, i_2, \dots, i_k$ совпадают с соответствующими столбцами исходной матрицы $A$, т.е. $b_{ii_l} = a_{ii_l}, \forall i, l=\overline{1,k}$. Остальные столбцы матрицы $B$ полагаются равными нулю.

Очевидно, что если мощности ненулевых столбцов $\beta_{i_l}$ не превосходят соответствующего числа $d_{i_l}, \beta_{i_l} \leqslant d_{i_l}, l=\overline{1,k}$, то полученный результат (минимальное покрытие) является допустимым в рассматриваемой задаче (1)—(4). В противном случае предлагаются два корректирующих алгоритма, в основе которых лежит процедура «усечения» ненулевых столбцов матрицы $B$.

Процедура «усечения» (ПУ) — процедура, предназначенная для уменьшения мощности $\beta_{i_l}$ столбцов матрицы $B$ до размера соответствующего требования $d_{i_l}$, путем замены «ненужных» единиц в ненулевых столбцах матрицы $B$ нулями. В качестве исходных данных для нее берется матрица $B$ и вектор дополнительных ограничений $D=(d_1, d_2, \dots, d_n)$.

Алгоритмическая схема ПУ:

**Шаг 1.** Ввод исходных данных: $m, n$ — размерность, $B$ — корректируемая матрица, $D$ — вектор ограничений.

**Шаг 2.** Определение множества $J$ номеров столбцов матрицы $B$, мощность которых не удовлетворяет ограничению, т. е. $J=\{p : \beta_p > d_p, p=\overline{1,j}\}$.

**Шаг 3.** Если $J$ не пусто, то последовательный выбор столбца из полученного множества $J$ и подсчет количества «лишних» единиц (kol) в нем, как разность между мощностью столбца $\beta_p$ и ограничителем $d_p$ ($\mathrm{kol}=\{q : q=\beta_p - d_p, p \in J\}$). Иначе переход к шагу 8.

**Шаг 4.** Построчный перебор элементов выбранного столбца. Если текущая единица в этом столбце не единственная в строке матрицы $B$, то замена ее нулем.

**Шаг 5.** Переопределение значения kol. Если $\mathrm{kol}=0$, то переход к шагу 7, иначе следующий шаг.

**Шаг 6.** Построчный перебор элементов выбранного столбца. Если текущая единица в этом столбце не единственная в строке матрицы $A$, то замена ее нулем. Переход к шагу 5.

**Шаг 7.** Удаление номера текущего столбца из множества $J$, переход к шагу 3.

**Шаг 8.** Процедура усечения закончена. Матрица $B$ состоит из допустимых столбцов.

Алгоритм корректировки 1 (АК1) предназначен для быстрого, но возможно менее точного поиска результата. Для этого выполняется процедура усечения матрицы $B$ ($B$ получена в результате решения задачи о покрытии без дополнительных ограничений). В случае, если после ПУ матрицы $B$ покрытие не получено, фиксируются столбцы матрицы $A$, которыми последовательно дополняется матрица $B$. При этом в первую очередь выбираются те столбцы, которые имеют наибольшую мощность (в случае если мощности одинаковые, выбирают столбцы, которые имеют наибольшее ограничение на размер $d_j$). Дополнение матрицы $B$ выполняется до тех пор, пока множество номеров еще не покрытых строк не станет пусто.

Алгоритмическую схему АК1 удобно представить следующим образом:

**Шаг 1.** Ввод исходных данных: $m, n$ — размерность, $B$ — матрица, сформированная по правилу ПФМ, $A$ — матрица покрытия.

**Шаг 2.** Выполнение процедуры усечения (ПУ) матрицы $B$.

**Шаг 3.** Определение множества $I$ номеров строк матрицы $B$, соответствующих еще не покрытым на данной итерации элементам. Если $I$ пусто, то переход к шагу 6.

**Шаг 4.** Определение столбца матрицы $A$, еще не используемого в $B$, с наибольшей мощностью $\beta_j$ и наибольшим ограничением $d_j$ ($j=\overline{1,n}, j \neq i_1, \dots, i_k$).

**Шаг 5.** Замена соответствующего столбца матрицы $B$ столбцом, определенным на предыдущем шаге. Переход к шагу 2.

**Шаг 6.** Останов. Решение задачи зафиксировано (получено допустимое покрытие).

Предлагается еще один алгоритм — алгоритм корректировки 2 (АК2). От предыдущего он отличается возможностью получения более точного результата. С этой целью вначале выполняется процедура усечения матрицы $B$ (матрица $B$ получена в результате решения задачи о покрытии без дополнительных ограничений). В случае, если после ПУ матрицы $B$ решение не получено, то фиксируются столбцы матрицы $A$, которыми последовательно дополняется матрица $B$. При этом, в первую очередь, выбираются те столбцы, которые имеют наибольшую мощность (в случае если мощности одинаковые, выбирают столбцы, которые имеют наибольшее ограничение на размер $d_j$). Каждый полученный результат запоминается, и при дальнейшем переборе всех допустимых вариантов отбирается лучший. Дополнение матрицы $B$ выполняется до тех пор, пока множество номеров еще не покрытых строк не станет пустым.

Алгоритмическую схему АК2 удобно представить следующим образом:

**Шаг 1.** Ввод исходных данных: $m, n$ — размерность, $B$ — матрица, сформированная по правилу ПФМ, $A$ — матрица покрытия.

**Шаг 2.** Определение множества $Q$ номеров столбцов матрицы $A$, еще не используемых в матрице $B$ (с наибольшей мощностью $\beta_j$ и с наибольшим ограничением на $d_j$ $(j = \overline{1, n}$, $j \neq i_1, \dots, i_k)$.

**Шаг 3.** Вызов процедуры усечения ПУ для матрицы $B$.

**Шаг 4.** Определение множества $I$ номеров строк матрицы $B$, соответствующих еще не покрытым на данной итерации элементам. Фиксирование множества $I$. Если $I$ пусто, то переход к шагу 8.

**Шаг 5.** Если $Q$ пусто, то переход к шагу 7, иначе следующий шаг.

**Шаг 6.** Перебор столбцов с номерами из $Q$. Замена текущим столбцом столбца матрицы $B$. Удаление номера столбца из $Q$. Переход к шагу 3.

**Шаг 7.** Среди всех, зафиксированных на шаге 4 множеств $I$, выбор наиболее малочисленного. Замена соответствующего столбца в матрице $B$. Переопределение $Q$. Переход к шагу 3.

**Шаг 8.** Останов. Решение задачи зафиксировано.

На основе этих алгоритмов корректировки предлагаются два метода решения задачи о минимальном покрытии с дополнительными ограничениями (1)—(4).

*Вариант 1 (ВА1).* В этом методе за основу получения первого покрытия для задачи без дополнительных ограничений берется точное решение, полученное методом ветвей и границ. Затем применяется алгоритм АК1 или алгоритм АК2. Преимуществом этого метода (в особенности при использовании АК2) является точность полученного результата. Однако для задач больших размеров применение этого варианта затруднено из-за большого времени решения.

*Вариант 2 (ВА2).* За основу получения первого покрытия для задачи без дополнительных ограничений берется приближенное решение, полученное одним из известных «жадных» алгоритмов. Затем применяется алгоритм АК1 или АК2. Вычислительный эксперимент показал, что задачи больших размеров ($m, n > 200$) удобно решать с помощью ВА2 с применением алгоритма корректировки АК1. Для сильно разреженных матриц покрытия (число единиц меньше 30 %) предпочтительней использовать ВА1 с любым алгоритмом корректировки.

СПИСОК ЛИТЕРАТУРЫ

1. *Мухин А.В.* Распределительная задача с разрывной целевой функцией / А. В. Мухин, Г. Д. Чернышова // Вест. Воронеж. ун-та. Сер. Физика, математика. — 2003. — № 1. — С. 161—164.

2. *Федорова И.В.* Алгоритмизация одной задачи передачи информации / И. В. Федорова, Т. В. Федорова, Г. Д. Чернышова // Вест. Воронеж. ун-та. Сер. Физика, математика. — 2003. — №1. — С. 189—193.