УДК 519.8

УДК 519.8

Г.А. ШУЛИНОК

О ПОИСКЕ КРАТЧАЙШИХ ПУТЕЙ В ЧИСЛОВЫХ ГРАФАХ

Рассматриваются натуральные модульные графы. Исследуется проблема нахождения путей между вершинами связного графа. Предлагаются алгоритмы решения данной задачи.

Введение. Известно [1], что алгоритмы на числовых графах могут быть намного эффективнее, чем для обычных графов, заданных матрицами или списками связности. И если экономия на памяти является очевидной (линейная память вместо квадратичной), то с вычислительной сложностью все далеко не так очевидно. В [2] была высказана гипотеза, что для всех NP-полных задач на графах найдется полиномиальный алгоритм на числовых графах. И данная работа вносит определённый вклад в этом направлении.

Целью данной работы было найти аналог алгоритма Дейкстры [3] для натуральных модульных графов. Алгоритм Дейкстры используется достаточно широко, и в многих случаях на графах, которые могли быть представлены в виде модульных или арифметических.

Определение. Числовым графом назовем такой граф $G(X,U,f)$, в котором X – множество вершин, U – множество образующих, а f – функция связности, заданная на множестве вершин, т.е. любые вершины $x, y \in X$ будут соединяться ребром лишь тогда, когда $f(x,y) \in U$. Если $X = \mathrm{N}_n$, то граф называют натуральным. А если $f(x,y) = |x-y|$, то такой граф называют модульным.

Рассмотрим сначала натуральный модульный граф (*NM*-граф) с двумя образующими. Обозначим такой граф $\mathrm{NM}_n(u_1,u_2)$, где n – число вершин графа, а множество образующих $U = \{u_1,u_2\}$.

Пусть вершины x, y соединены путём. Обозначим $P(X,Y)$ – путь, соединяющий вершины x и y.

Пусть вершины, через которые проходит путь P, будут иметь коды $(x = x_1^P, x_2^P, \ldots, x_k^P = y)$. В таком случае, из определения графа вытекает, что $\left|x_1^P - x_2^P\right| \in U$, $\left|x_2^P - x_3^P\right| \in U$, …, $\left|x_{k-1}^P - x_k^P\right| \in U$ т.е. $\left|x_i^P - x_{i+1}^P\right| = u_1$ или $\left|x_i^P - x_{i+1}^P\right| = u_2$, $0 < i < k$. Таким образом можно предложить простой способ нахождения пути в связном *NM*-графе.

Алгоритм 1. Нахождение пути в связном графе $\mathrm{NM}_n(u_1, u_2)$.

Исходные данные: *NM*-граф $\mathrm{NM}_n(u_1, u_2)$, $u_1 < u_2$. Необходимо найти путь из вершины x в y, $0 < x < y \le n$.

Шаг 1. Определим $d = y - x$.

Шаг 2. Если $d(\mathrm{mod}\, u_1) \equiv 0$ или $d(\mathrm{mod}\, u_2) \equiv 0$, то стоп. Путь найден. Иначе – переходим на следующий шаг.

Шаг 3. Если $d > 0$, то перейти к вершине $x = x + u_2$, иначе – к вершине $x = x - u_1$. Перейти на шаг 1.

Докажем, что данный алгоритм 1 позволяет найти путь в графе.

Пусть вершины x и y соединены ребром. В таком случае алгоритм 1 остановится на шаге 2. Действительно, из того, что x и y соединены ребром вытекает, что $d = u_1$ или $d = u_2$, а значит, выполняется либо условие $d(\mathrm{mod}\, u_1) \equiv 0$, либо $d(\mathrm{mod}\, u_2) \equiv 0$.

Подобным образом, если вершины x и y соединены рёбрами, порождёнными только одной из образующих, то условие на шаге 2 тоже будет выполняться. Более того, из связности графа [4] вытекает, что $\text{НОД}(u_1, u_2) = 1$, а значит, путь будет кратчайшим. Исключением будет случай когда $d = u_1 \cdot u_2$, в этом случае следует выбрать большую образующую.

Пусть вершины x и y соединены двумя рёбрами, причём эти ребра порождены двумя разными образующими. В этом случае можно сказать, что $x \pm u_1 \pm u_2 = y$. Исходя из того, что $x < y$ и $u_1 < u_2$, имеет место равенство $x + u_2 - u_1 = y$, что соответствует итерации алгоритма 1.

В случае, когда между вершинами x и y более 2 рёбер, применяем индукцию по вершинам, и, очевидно, что алгоритм 1 обеспечивает построение пути. В то же время нельзя сказать, что все пути, полученные в результате работы алгоритма 1, будут кратчайшими.

Рассмотрим диофантово уравнение

$$ru_1 + su_2 = d\,. \tag{1}$$

Кратчайший путь из вершины x в y отвечает условию

$$|r| + |s| \to \min . \qquad (2)$$

В самом деле, путь из x в y исходя из алгоритма 1 можно представить в виде $(-a, b)$, где a – количество рёбер, порождённых образующей u_1, а b – рёбер, порождённых образующей u_2. Поскольку произвольный путь из вершины x в y ^{y} будет удовлетворять уравнению (1), то для найденного пути $(-a, b)$ можно предложить такое соотношение:

$$\left(-a + u_2 k, \quad b - u_1 k\right),\ k \in \mathrm{Z} . \qquad (3)$$

Из соотношения (3) становится очевидным обоснованность алгоритма 1. В то же время можно построить алгоритм, который будет находить кратчайший путь в графе.

Рассмотрим граф $\mathrm{NM}_{12}(3, 5)$.

Найдем путь из вершины 1 в вершину 9. Такие пути можно построить следующим образом:

1) 1→6→9;
2) 1→4→9;
3) 1→6→11→8→5→2→7→12→9
4) 1→4→7→10→5→8→3→6→9 и т.д.

Исходя из первого пути и применив формулу (3) имеем:

$$\left(1 + 5k, \quad 1 - 3k\right),\ k \in \mathrm{Z} . \qquad (4)$$

Из соотношения (4) видно, что пути (1) и (2) эквивалентны, путь (3) получается при $k = -1$, а (4) при $k = 1$. Из линейности соотношений (3) вытекает, что минимум соотношения (2) достигается в данном случае при $k = 0$, т. е. кратчайший путь, соединяющий вершины 1 и 9, состоит из двух рёбер.

Если в графе найти путь из вершины 1 в 2, то получится следующее решение уравнения: $3r + 5s = 1$, решения которого по алгоритму 1 можно записать так: $\left(-3 + 5k, \quad 2 - 3k\right),\ k \in \mathrm{Z}$.

И как результат, при $k = 1$ длина пути будет 3, при $k = 0$ – 5, а при $k = 2$ – 11. Итак, из частного решения, которое мы получаем из алгоритма 1, можно получить минимальное решение.

Заметим, что образующие входят в кратчайший путь с разными знаками, в отличие от алгоритма 1, где образующая u_1 всегда со знаком «-», а образующая u_2 – со знаком «+». Построим итеративную процедуру, при которой выбирается подходящая образующая и её знак на каждый элемент пути.

Рассмотрим путь из вершины 1 в 8. В этом случае кратчайший путь будет иметь вид (-1,2) или 1→6→3→8. Определим разности $d_1 = 8 - 1 = 7$, $d_2 = 8 - 6 = 2$, $d_3 = 8 - 3 = 5$. Как видно, $d_1 > u_2$, $d_2 < u_1$, а $d_3 = u_2$. Таким образом видно следующий подход: если $d_1 > u_2$, то имеет смысл использовать образующую u_2 до тех пор, пока это условие выполняется. Иначе говоря, пока $0 < d(\mathrm{mod}\, u_2) < u_2$. В самом деле, если $d(\mathrm{mod}\, u_2) \equiv 0$, то путь представляет собой $\frac{d}{u_2}$ рёбер, порождённых образующей u_2, а в противоположном случае – $d < u_2$. С другой стороны, если $0 < d < u_1$, то кратчайший путь состоит из последовательности применения обеих образующих, при этом образующая u_1 входит со знаком «+», а образующая u_2 в – со знаком «–». Если $u_1 < d < u_2$, то кратчайший путь состоит из ряда применений обеих образующих с разными знаками. Так для графа $\mathrm{NM}_{12}(3,5)$ возможны такие пути длины 4 из 1 в 5 ($(3 < d < 5)$):

1) 1→4→7→10→5 (+3,+3,+3,-5);
2) 1→4→7→2→5 (+3,+3,-5,+3);
3) 1→6→11→8→5 (+5,+5,-3,-3);
4) 1→6→3→8→5 (+5,-3,+5,-3).

Таким образом, возможны различные кратчайшие пути. Однако обнаруживается следующее свойство: если $d(\mathrm{mod}\, 2) \equiv 2$, то образующая u_1 входит в кратчайший путь со знаком «+», а если $d(\mathrm{mod}\, 2) \equiv 1$, то со знаком «–». Также особый случай $d = (u_2 - u_1)$, когда выбирается образующая u_1 со знаком «–» и при этом чётность d не играет роли.

Например, при нахождении кратчайшего пути из 1 в 2 по такому варианту будем иметь следующий сценарий:

$d_1 = 1 < u_1$. Выбираем образующую 3, переходим в вершину 4.

$d_2 = 2 - 4 = -2 < 0$. Меняем направление поиска: идём из 2 в 4.

$d_2' = 2 < u_1$, но $2 = u_2 - u_1$. Значит, имеем либо -3, либо +5. Образующая -3 не подходит (2-3 <1), значит, из вершины 2 в 7 идём с помощью 5.

$d_3 = 7 - 4 = u_1$ – поиск окончен.

Полученный путь (1, 4, 7, 2) является кратчайшим, что можно определить из соотношения (3) полученного подстановкой частного решения (2, -1).

Таким образом, можно предложить алгоритм построения кратчайшего пути между произвольными вершинами графа $\mathrm{NM}_n(u_1, u_2)$.

Алгоритм 2. Нахождение кратчайшего пути в графе $\mathrm{NM}_n(u_1, u_2)$.

Исходные данные: связный граф $\mathrm{NM}_n(u_1, u_2)$; ищется кратчайший путь из вершины x в y, $0 < x < y \le n$.

Шаг 1. Вычислим $d = y - x$.

Шаг 2. Если $d(\mathrm{mod}\, u_1) \equiv 0$ или $d(\mathrm{mod}\, u_2) \equiv 0$, то стоп – путь найден. Иначе – перейти на следующий шаг.

Шаг 3. Если $d(\mathrm{mod}(u_2 - u_1)) \equiv 0$, то путь найден, стоп. Иначе – перейти на шаг 4.

Шаг 4. Если $d < 0$ – изменить направление поиска, $d' = -d$, $x' = y$, $y' = x$.

Шаг 5. Если $d < u_1$, то $x = x + u_1$, иначе если $d > u_2$, то $x = x + u_2$, иначе, если $d(\mathrm{mod}\,2) \equiv 1$, то $x = x + u_2$, иначе $x = x + u_1$. Перейти на шаг 1.

Заметим, что вычислительная ёмкость алгоритма 2 не превышает ёмкости алгоритма 1, а в среднем даже будет меньше, поскольку пути, получаемые с помощью алгоритма 2 короче.

С другой стороны, легко увидеть, что в алгоритме 2 образующие входят в путь только со знаком «+». Чтобы определиться с этим требуется учитывать знак первой итерации.

Заключение. Полученные результаты показывают, что использование числовых графов позволяет получить более эффективные алгоритмы, требующие меньшего времени вычисления и памяти.

Развитие данной работы будет направлено на нахождение подобных алгоритмов и для других известных подклассов числовых графов.

І.Е. Шулінок, Г.О. Шулінок

ПРО НАЙКОРОТШІ ШЛЯХИ У ЧИСЛОВИХ ГРАФАХ

Розглядаються натуральні модульні графи. Досліджується проблема знаходження шляхів у таких графах. Пропонуються алгоритми, що розв'язують дану задачу.

I.E. Shulinok, G.A. Shulinok

ABOUT SHORTEST PATHS IN NUMERICAL GRAPHS

Natural Modular Graphs are considered. A problem to find paths in such graphs is investigated. Algorithms to solve this problem are proposed.

1. *Донец Г.А.,Шулинок И.Э.* О сложности алгоритмов поиска в глубину на модульных графах // Теория оптимальных решений. – К.: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2002. – С. 105–110 .
2. *Донец Г.А.,Шулинок И.Э.* Об оценке сложности алгоритмов для натуральных модульных графов // Теория оптимальных решений. – К.: Ин-т кибернетики им. В.М.Глушкова НАН Украины, 2001. – С. 61–68.
3. *Харари Ф.* Теория графов. – М.:Мир, 1973. – 301 с.
4. *Шулинок И.Э.* О связности натуральных модульных графов // Кибернетика и системный анализ. – 1998. – № 5. – С. 50–53 .

Получено 14.05.2012