

## МИНИМИЗАЦИЯ ВЕСОВОЙ СУММЫ РАССТОЯНИЙ ДЛЯ СЕТЕВОГО ОБЪЕКТА

Назаренко В.И., Кравченко А.Г.

Кафедра ЭВМ ДонГТУ

vn@cs.dgtu.donetsk.ua

### Abstract

*Nazarenko V.I., Kravchenko A.G. Minimization of weighting sum of distances for a network object.*

*The problem of minimization of weighting sum of distances to  $r > 1$  points on a plane, known as Steiner-Weber problem, is considered. The solution of this problem is used as one of the stages of computeraided design of routes in network objects (e.g. power supply networks, gas- and oil pipeline networks, etc.).*

**Постановка задачи.** Задан ориентированный граф в виде бинарного дерева (рис.1), имеющий  $m$  узлов и  $n$  ветвей.

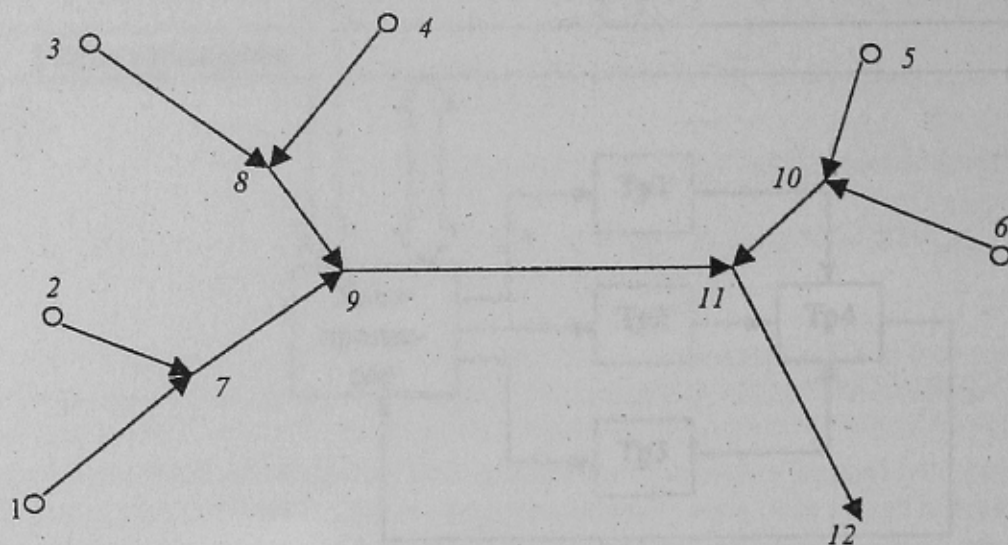


Рисунок 1 - Граф сетевого объекта

Узлы разделяются на два типа: фиксированные (миноранты и мажоранты) и нефиксированные. Для каждой ветви указан ее удельный вес. Требуется определить такие координаты нефиксированных узлов, при которых обеспечивается минимизация целевой функции

$$F = \min \sum_{j=1}^n C_j L_j, \quad (1)$$

$$L_j = \sqrt{(x_j - x_k)^2 + (y_j - y_k)^2}.$$

где

$C_j$  - удельный вес  $j$ -ой ветви;

$L_j$  - длина  $j$ -ой ветви;

$j, k$  - номера начального и конечного узлов  $j$ -ой ветви.

Сформулированная выше задача известна как задача Штейнера-Вебера, заключающаяся в минимизации весовой суммы расстояний до  $r > 1$  точек на плоскости. М.Аоки [1], рассматривая переход от задачи Штейнера ( $r = 1$ ) к задаче Штейнера-Вебера, отмечал, что в последнем случае ее решение становится удивительно сложным.

Задача Штейнера-Вебера имеет место, в частности, при автоматизированном проектировании трасс сетевых объектов (газо- и нефтесборные сети, сети энергоснабжения и др.), поэтому ее решение является практически важным.

Примем следующую нумерацию узлов и ветвей дерева сети:

- мажоранте (корню дерева) присвоим номер  $m$ ;
- минорантам в произвольном порядке присвоим номера  $1 \dots k-1$ ;
- нефиксированным узлам поставим в соответствие номера  $k \dots m-1$ ;
- ветвям сети присвоим номера, совпадающие с номерами их начальных узлов.

Указанная система нумерации узлов и ветвей упрощает формализацию алгоритма оптимизации сети и его программную реализацию.

Граф сети однозначно отображается матрицей инциденций  $A$  и векторами  $X$  и  $Y$ , где

$$A = (a_{ij}), \quad X = (x_j), \quad Y = (y_j), \quad i = 1 \dots m, \quad j = 1 \dots n,$$

$$a_{ij} = \begin{cases} 1, & \text{если ветвь } j \text{ исходит из узла } i, \\ -1, & \text{если ветвь } j \text{ входит в узел } i, \\ 0, & \text{если ветвь } j \text{ не инцидентна узлу } i. \end{cases}$$

$x_j, y_j$  - координаты  $j$ -го узла.

Для рис. 1 матрица инциденций имеет вид

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}$$

Для выбора метода решения задачи Штейнера-Вебера из современного арсенала методов оптимизации произведем детальное исследование свойств ее целевой функции (1).

Функция (1) является выпуклой, унимодальной [1]. Как показано в [2], в случае выпуклой функции необходимым и достаточным условием оптимальности для задач без ограничений является  $\nabla F(x^*, y^*) = 0$ , где  $(x^*, y^*)$  - координаты многомерной оптимальной точки,  $\nabla F$  - градиент функции  $F$ .

Задание топологии сети в виде матрицы  $A$  при вводе информации в ЭВМ является неудобным из-за ее громоздкости. Более удобным является представление топологии в виде условной матрицы инциденций  $T = (t_{ij})$ ,  $i = 1 \dots n$ ,  $j = 1, 2$ . Здесь номер строки соответствует номеру ветви, элементы  $t_{i1}$  и  $t_{i2}$  - номера начального и конечного узлов  $i$ -ой ветви.

Для вычисления градиента функции (1) целесообразно применять усеченную матрицу инцидентий  $B$ , получаемую из матрицы  $A$  вычеркиванием строк, соответствующих фиксированным узлам.

Для рис. 1 имеем

$$B = \begin{pmatrix} -1 & -1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 \end{pmatrix}$$

Матрицу  $B$  нетрудно сформировать в программе по введенной матрице  $T$ . Форму представления матрицы  $B$  также можно сделать компактной, оставив в каждой ее строке лишь номера ненулевых элементов:

$$U = \begin{pmatrix} 1 & 2 & 9 \\ 3 & 4 & 9 \\ 7 & 8 & 11 \\ 5 & 6 & 11 \\ 9 & 10 & 12 \end{pmatrix}$$

В матрице  $U$  для каждого нефиксированного узла элементы первых двух столбцов определяют номера входящих ветвей; элемент третьего столбца - номер конечного узла исходящей ветви; номер строки матрицы  $U$  - это номер нефиксированного узла, уменьшенный на  $k-1$ .

Введем векторы  $V = (v_1, v_2, \dots, v_n)$  и  $W = (w_1, w_2, \dots, w_n)$ , определяющие проекции длин ветвей на оси  $X$  и  $Y$ :  $V = T_x \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ ;  $W = T_y \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ , где  $T_x, T_y$  - матрицы, получаемые из  $T$  подстановкой координат узлов вместо их номеров.

Тогда элементы вектора длин ветвей  $L = (l_1, l_2, \dots, l_n)$ :

$$l_j = \sqrt{v_j^2 + w_j^2 + \varepsilon}$$

Малое положительное число  $\varepsilon$  введено для того, чтобы исключить получение нулевой длины ветви, если при оптимизации сети нефиксированный узел совпадает с одним из других узлов (в этом случае градиент целевой функции имел бы точку разрыва).

Для вычисления градиентов определим элементы весовых векторов  $S_x$  и  $S_y$ :

$$S_x = \begin{pmatrix} c_1 \frac{v_1}{l_1} \\ c_2 \frac{v_2}{l_2} \\ \dots \\ c_n \frac{v_n}{l_n} \end{pmatrix}; \quad S_y = \begin{pmatrix} c_1 \frac{w_1}{l_1} \\ c_2 \frac{w_2}{l_2} \\ \dots \\ c_n \frac{w_n}{l_n} \end{pmatrix}$$

Тогда проекции градиента на оси  $X$  и  $Y$ :



$$D_x = \begin{pmatrix} \frac{\partial F}{\partial x_1} \\ \frac{\partial F}{\partial x_2} \\ \dots \\ \frac{\partial F}{\partial x_n} \end{pmatrix} = B \cdot S_x; \quad D_y = \begin{pmatrix} \frac{\partial F}{\partial y_1} \\ \frac{\partial F}{\partial y_2} \\ \dots \\ \frac{\partial F}{\partial y_n} \end{pmatrix} = B \cdot S_y$$

Для рис. 1

$$D_x = \begin{pmatrix} -S_{x1} - S_{x2} + S_{x7} \\ -S_{x3} - S_{x4} + S_{x8} \\ -S_{x7} - S_{x8} + S_{x9} \\ -S_{x5} - S_{x6} + S_{x10} \\ -S_{x9} - S_{x10} + S_{x11} \end{pmatrix}$$

Модуль градиента

$$\nabla F = \sqrt{\sum_{i=k}^m \left( \frac{\partial F}{\partial x_i} \right)^2 + \left( \frac{\partial F}{\partial y_i} \right)^2}$$

Методы минимизации, применяемые для решения задачи Штейнера-Вебера, могут быть разделены на два основных класса: первого и второго порядков.

Как известно, для  $n$ -мерной функции  $f(z)$ , градиент  $f'(z)$  в точке  $z$  указывает направление наиболее быстрого ее возрастания. Методы минимизации первого порядка основаны на движении в направлении антиградиента и отличаются друг от друга в основном способом выбора шага в заданном направлении [2, 3]. В градиентном методе с дроблением шага значение шага автоматически увеличивается или уменьшается в зависимости от изменения значения целевой функции при движении в направлении антиградиента. В методе наискорейшего спуска значение шага предварительно вычисляется таким образом, чтобы целевая функция в направлении антиградиента приняла минимально возможное значение.

Метод наискорейшего спуска обладает более высокой скоростью сходимости и устойчивостью по сравнению с градиентным методом с дроблением шага, но тем не менее, как и другие методы первого порядка, мало пригоден для минимизации функций, которым присущен эффект "оврага". Для иллюстрации такого случая рассмотрим рис. 2, на котором изображены линии уровня функции двух переменных. Здесь целевая функция имеет явно выраженный "овраг". При мало изменяющемся шаге мы можем в процессе движения к оптимальной точке делать шаги с одного склона оврага на другой. При этом значение  $F(x, y)$  будет убывать весьма медленно, что предопределяет низкую сходимость процесса решения.

Покажем, что целевая функция в задаче Штейнера-Вебера обладает овражным характером.

Квадратичная функция

$$f = \sum_{i=1}^n k_i (x_i - a_i)^2$$

при  $k_1 = k_2 = \dots = k_n$  представляет собой  $n$ -мерную сферу. Для нее линии уровня (сечение

гиперплоскостью) имеют вид  $(n-1)$ -мерной окружности. При  $k_i \neq k_j$ ;  $i, j = 1..n$ ;  $i \neq j$  линии уровня принимают форму эллипсов. Если коэффициенты  $k$  сильно отличаются друг от друга, линии уровня становятся вытянутыми вдоль соответствующих осей координат и образуют "овраги".

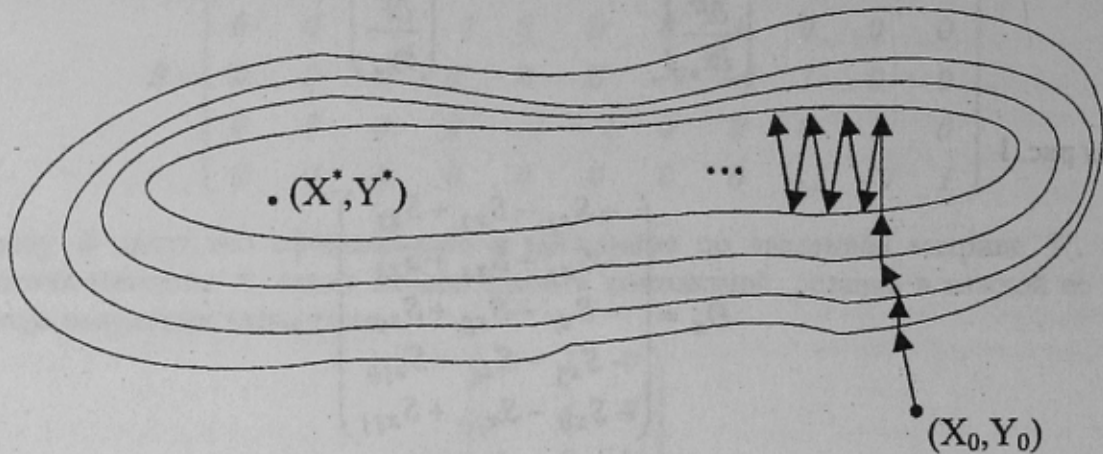


Рисунок 2 - Линии уровня функции двух переменных

Функция (1) близка к квадратичной. Следовательно, при резко отличающихся друг от друга весовых коэффициентах  $C_j$ , она также имеет явно выраженный овражный характер.

Градиентные методы основаны на тейлоровской аппроксимации функции  $f(x)$  первого порядка, т.е. на линейной аппроксимации. Такая аппроксимация в  $n$ -мерной точке  $x_1$  имеет вид (первые два элемента ряда Тейлора)

$$f(x) = f(x_1) + \nabla f(x_1) (x - x_1)$$

Более точной является квадратичная аппроксимация, особенно в окрестности оптимальной точки  $x^*$ . Разложение в ряд Тейлора в точке  $x_1$ :

$$f(x) = f(x_1) + \nabla f(x_1)^T (x - x_1) + \frac{1}{2} (x - x_1)^T H(x_1) (x - x_1)$$

Для  $n$ -мерной задачи минимизации требуется определение корня уравнения  $\nabla f(x) = 0$ . Тогда разложение в ряд Тейлора может быть записано следующим образом:

$$0 = \nabla f(x_{k+1}) \approx \nabla f(x_k) + H(x_k)(x_{k+1} - x_k)$$

Последовательность точек  $\{x_k\}$  определяется формулой

$$x_{k+1} = x_k - H^{-1}(x_k) \nabla f(x_k) \quad (2)$$

при условии, что обратная матрица  $H^{-1}(x_k)$  существует.

Непосредственное использование формулы (2), на которой основан метод Ньютона-Рафсона [3], требует на каждой итерации вычислять вторые частные производные целевой функции и выполнять обращение полученной таким образом матрицы Гессе, что в свою очередь требует больших затрат машинного времени.

Рассмотрим два метода минимизации второго порядка, не использующие непосредственно матрицу Гессе: метод переменной метрики и метод сопряженных градиентов. В методе переменной метрики [4] вторые частные производные целевой функции не вычисляются. Матрица  $H^{-1}(x)$  определяется постепенно, путем построения последовательности матриц, являющихся обращением матрицы  $H(x)$  на все больших и



больших подпространствах евклидова пространства  $E_n$ . В методе сопряженных градиентов [4, 5] необходимая информация о гессиане также формируется в течение нескольких итераций, но в отличие от предыдущего метода здесь используется последовательность направлений поиска  $\{r\}$ , каждое из которых является линейной комбинацией антиградиента в текущей точке, и предыдущего направления поиска. Экспериментальная апробация указанных двух методов на ЭВМ показала, что метод сопряженных градиентов применительно к задаче Штейнера-Вебера более предпочтителен как по скорости сходимости, так и по устойчивости процесса решения.

Алгоритм метода сопряженных градиентов выполняется в следующей последовательности.

1. В начальной многомерной точке  $x_0$  определяется направление поиска  $r_0 = -f'(x_0)$ , где  $f'(x_0)$  - градиент целевой функции.

2. По направлению  $r_0$  методом "золотого сечения" решается задача одномерной минимизации

$$f(x_0 + \rho \cdot r_0) = \min f(x_0 + \rho_0 r_0)$$

в результате чего определяются шаг  $\rho^*$  и точка  $x_1 = x_0 + \rho r_0$ .

3. Вычисляются значения целевой функции  $f(x_1)$  и ее градиента  $f'(x_1)$  в точке  $x_1$ .

4. Вычисляется корректирующий коэффициент направления поиска

$$\beta = \frac{(f'(x_1), f'(x_1) - f'(x_0))}{(f'(x_0), f'(x_1))},$$

где  $(u, v)$  - скалярное произведение векторов  $U$  и  $V$ .

5. Новое направление движения:  $r_1 = -f'(x_1) + \beta r_0$

6. Определяются оптимальный шаг  $\rho_1^*$  в направлении  $r_1$

$$f(x_1 + \rho_1 r_1) = \min f(x_1 + \rho_1 r_1),$$

и точка  $x_2 = x_1 + \rho_1 r_1$ .

7. Вычисляются  $f(x_2)$ ,  $f'(x_2)$  и т.д. Пункты 2 .. 6 повторяются  $n$  раз, где  $n$  - количество оптимизируемых координат, после чего производится переход на шаг 1 при  $x_0 = x_{n+1}$ .

Рассмотренный алгоритм оптимизации с использованием метода сопряженных градиентов реализован на ПЭВМ IBM PC и показал высокую эффективность при расчетах применительно к реальным сетевым объектам.

#### ЛИТЕРАТУРА

1. Аоки М. Введение в методы минимизации. - М.: Наука, 1977. - 344 с.
2. Зангвилл У.И. Нелинейное программирование. - М.: Сов.радио, 1973. - 312 с.
3. Моисеев Н.Н., Иванилов Ю.П., Столярова Е.М. Методы оптимизации. - М.: Наука, 1978. - 352 с.
4. Химмельблау Д. Прикладное нелинейное программирование. - М.: Мир, 1975. - 534 с.
5. Базара М., Шетти К. Нелинейное программирование. Теория и алгоритмы. - М.: Мир, 1982. - 583 с.