

УДК 519.968

doi 10.26089/NumMet.v18r213

НОВЫЙ АЛГОРИТМ ОПТИМИЗАЦИИ ДИЗАЙНА ТРАНСПОРТНЫХ СЕТЕЙ С УЧЕТОМ ОГРАНИЧЕНИЙ

А. А. Ананьев¹, П. В. Ломовицкий², Д. В. Ужegov³, А. Н. Хлюпин⁴

Предложен эвристический алгоритм построения транспортной сети сбора оптимальной геометрии с ограничениями. Транспортная сеть представляется ориентированным взвешенным деревом Штейнера. Ограничения накладываются на максимальную суммарную длину участков коммуникаций от любой терминальной вершины до точки сбора. Учет ограничений происходит с помощью метода штрафных функций. Приведен анализ влияния параметров модели на оптимальную геометрию сети.

Ключевые слова: транспортные сети, задача Штейнера, алгоритмы на графах, оптимизация, задача с ограничениями.

1. Введение. Задача нахождения минимального взвешенного дерева Штейнера на евклидовой плоскости (задача Штейнера) является упрощенной моделью задачи проектирования транспортной сети сбора, которая состоит в минимизации капитальных издержек на строительство сети. Такая транспортная магистраль должна связывать множество источников ресурсов с одним потребителем и обеспечивать протекание потока в объеме, удовлетворяющем спросу. Задача Штейнера состоит в минимизации длины взвешенного дерева Штейнера, которая характеризуется повышением веса отдельного участка дерева (стоимости отрезка коммуникации) при увеличении потока сырья по данному ребру.

Данная задача является NP-сложной, т.е. время работы алгоритма для точного решения экспоненциально увеличивается с ростом числа параметров. Существуют алгоритмы приближенного решения задачи Штейнера (эвристические алгоритмы). В работе [1] предлагается алгоритм решения задачи Штейнера для построения сети, соединяющей сток и терминальные вершины с заданной мощностью. В [2] на основе свойств минимального дерева Штейнера описывается процедура определения конфигурации с начальным размещением дополнительных точек Штейнера для заданного множества терминальных вершин; затем получившееся начальное размещение уточняется с помощью локальной оптимизации. В [3] предлагается использовать подход, позволяющий изменять начальную ориентацию графа (глобальная оптимизация) для решения задачи нахождения транспортного пути между двумя множествами источников и стоков. К сожалению, алгоритмы, разработанные в упомянутых работах, решают безусловную задачу оптимизации и, как следствие, они не могут быть применены при проектировании реальных транспортных сетей. На практике может возникнуть множество различных ограничений, таких как, например, ограничение на падение давления в трубах для поддержания потока. Подобные ограничения встречаются при проектировании сетей сбора в нефтяной и газовой отраслях. Их можно аппроксимировать ограничениями на максимальную суммарную длину ребер от источника до точки сбора. В [4, 5] предлагается алгоритм построения дерева Штейнера на графе с заданными точками Штейнера, источниками и стоком, которое обеспечивает максимальный доход от источников при заданных ограничениях на стоимость и максимальное количество ребер в дереве. В [6] решается задача нахождения минимального взвешенного дерева Штейнера на графе при заданном ограничении на максимальное количество ребер в дереве. В [4–6] задачи решаются на заданном графе при фиксированном положении точек Штейнера. Алгоритмы построения решения не учитывают возможности перемещения точек Штейнера для удовлетворения ограничениям и максимизации прибыли (минимизации длины взвешенного дерева Штейнера).

¹ Инжиниринговый центр МФТИ по трудноизвлекаемым полезным ископаемым, ул. Первомайская, 5, г. Долгопрудный, Московская обл., 141700; инженер, e-mail: ananев.aa@cet-mipt.ru

² Инжиниринговый центр МФТИ по трудноизвлекаемым полезным ископаемым, ул. Первомайская, 5, г. Долгопрудный, Московская обл., 141700; инженер, e-mail: lomovitskiy.pv@cet-mipt.ru

³ Инжиниринговый Центр МФТИ по трудноизвлекаемым полезным ископаемым, ул. Первомайская, 5, г. Долгопрудный, Московская обл., 141700; инженер, e-mail: uzhegov.dv@cet-mipt.ru

⁴ Инжиниринговый Центр МФТИ по трудноизвлекаемым полезным ископаемым, ул. Первомайская, 5, г. Долгопрудный, Московская обл., 141700; руководитель исследовательской группы, e-mail: khlyupin.an@cet-mipt.ru

В настоящей статье предложен эвристический итерационный алгоритм построения оптимального взвешенного дерева Штейнера на однородной территории с ограничением. Алгоритм объединяет подходы, рассмотренные в работах [2, 3] и адаптированные при помощи метода штрафных функций [7]. Для примера рассматривается случай с одним потребителем (стоком) и множеством источников. Кроме того, в статье приведен анализ влияния параметров модели на оптимальную геометрию сети. В качестве ограничения используется максимальная суммарная длина по ребрам сети от любой терминальной вершины до корня дерева.

2. Постановка задачи. В работе [8] показано, что решение задачи Штейнера следует искать в виде древовидного ориентированного взвешенного дерева, которое растет из некоторой терминальной вершины (корня дерева). Для этого введем следующие обозначения: $I' = \{i: i = 0, \dots, n\}$ — множество терминальных вершин, заданных на плоскости; $I'' = \{i: i = n+1, \dots, n+N\}$ — множество дополнительных точек (точек Штейнера); $V = I' \cup I''$ — множество всех вершин графа; $i_0 \in I'$ — корень дерева.

Кроме того, необходимо иметь информацию о конфигурации сети: U — множество всех пар терминальных вершин (i, j) , $i, j \in V$, $E(i) = (in, out)$, $\{in, out\} \in V$, $0 \leq i \leq |E|$, $E \in U$, где E — множество пар вершин, каждая из которых является ребром дерева Штейнера, $|E|$ — количество ребер графа, а i -е ребро выходит из вершины out в вершину in . Для взвешенного дерева Штейнера определим W^V, W^E — множества весов, определенных на множествах вершин и ребер соответственно. В реальных транспортных сетях сбора в качестве этих множеств выступают дебиты источников (W^V) и стоимости всех отрезков коммуникации сети (W^E). Будем полагать линейную зависимость между потоком по ребру и стоимостью соответствующего участка сети. Определим элементы множества W^V : w_i^V считается известной, если $i \in I' \setminus \{i_0\}$; если же $i \in I''$, то $w_i^V = \sum_{j \in V: (i,j) \in E} w_j^V$.

Теперь определим элементы множества W^E : $w_k^E = w_{(i,j)}^E = w_j^V \forall k \in E: E(k) = (i, j)$, где j — вершина, из которой выходит ребро k . Тогда $G = (V, E, W)$ — ориентированный взвешенный граф, задающий транспортную сеть, где $W = W^V \cup W^E$.

Введем две дополнительные нелокальные функции L^+ и L^- для терминальных вершин. Сначала определим для каждой вершины графа два множества точек (ближайшие в графе родительские и дочерние вершины):

$$Parents(k) = \{j: (k, j) \in E\} \quad \forall k \in V, \quad Child(k) = \{j: (j, k) \in E\} \quad \forall k \in V.$$

С помощью построенных таким образом множеств родительских и дочерних вершин $\forall v \in V$ можно определить рекурсивным образом дополнительные нелокальные функции:

$$\forall i \in V(G): \begin{cases} L^+(i) = \begin{cases} \max_{j \in Parents(i)} (L^+(j) + d_{i,j}), & \text{если } Parents(i) \neq \emptyset, \\ 0, & \text{если } Parents(i) = \emptyset, \end{cases} \\ L^-(i) = \begin{cases} d_{i,j} + L^-(j), \quad j \in Child(i), & \text{если } Child(i) \neq \emptyset, \\ 0, & \text{если } Child(i) = \emptyset. \end{cases} \end{cases}$$

Здесь $d_{a,b}$ — длина ребра между вершинами a и b . Эти рекурсивно заданные функции определяют соответственно расстояние по ребрам до максимально удаленной родительской вершины графа, достижимой из данной, и расстояние по ребрам до точки сбора (рис. 1).

В работе [8] поставлена и математически формализована задача минимизации взвешенного дерева Штейнера (1)–(5):

$$D = \sum_{(in,out) \in E} w_{in,out}^E \cdot d_{in,out} \rightarrow \min_{(x_i, y_i), i \in I''}, \tag{1}$$

$$\forall E(i) = (in, out), \quad in \neq out, \tag{2}$$

$$\forall v \in I', \quad \exists i: \{E(i) = (in, v), \quad in \in I''\}, \tag{3}$$

$$\forall v \in I'', \quad \exists \{(in, v), (v, out_1), (v, out_2)\} \in E, \quad out_1, out_2 \in I', \quad in \in I'', \tag{4}$$

$$|I''| = N = n - 2, \tag{5}$$

где D — длина взвешенного дерева Штейнера, $w_{in,out}^E = w_{out}^V$ — вес ребра, определяющийся весом вершины, из которой выходит ребро графа.

Поставим задачу с ограничениями. Для этого формализуем задачу (1)–(5) на основе введенных нелокальных функций L^-, L^+ :

$$L^-(i) + L^+(i) < limit, \quad i \in V, \quad (6)$$

где $limit$ — ограничения на максимальную длину по ребрам графа от любой терминальной вершины до корня. Для решения задачи с ограничением используется метод штрафных функций, который относится к группе непрямых методов решения задач нелинейного программирования:

$$\begin{cases} f(x) \rightarrow \min_{x \in G}: (a \leq x \leq b), \\ h_i(x) < limit, \quad i = 1 \dots k, \\ g_j(x) = 0, \quad j = 1 \dots n. \end{cases}$$

Данный метод преобразует задачу с ограничениями в последовательность задач безусловной оптимизации некоторых вспомогательных функций. Последние получаются путем модификации целевой функции с помощью ограничивающих функций таким образом, чтобы ограничения в явном виде в задаче оптимизации не фигурировали. Это обеспечивает возможность применения методов безусловной оптимизации. В общем случае вспомогательная функция имеет вид $F(x, a) = f(x) + L(x, a)$, где $L(x, a)$ — штрафная функция с параметром $a > 0$.

Адаптируем задачу (1)–(6) для ее решения с учетом ограничений на максимальную суммарную длину по ребрам сети от любой терминальной вершины до корня дерева. Для этого введем штрафной член $Penalty$ в целевую функцию уравнения (1):

$$Penalty = a \sum_{k \in I' \setminus \{i_0\}} \left(\max(0, L^-(k) - limit) \right)^2, \quad (7)$$

где константа a должна определяться исходя из характерной длины взвешенного дерева. Тогда задача (1) будет выглядеть следующим образом:

$$DP = D + Penalty \rightarrow \min_{(x_i, y_i), i \in I''}. \quad (8)$$

В работе [2] приводится и доказывается теорема, которая утверждает, что длину взвешенного дерева Штейнера можно выразить следующим образом: $D = H_3 - \sum_{k=n+1}^{2n-2} H_k$, где $H_3 = \sum_{i \in I'} w_{i_0} d_{i_0 i}$ — длина звездного дерева, а H_k — эффективность k -й точки Штейнера, определяющаяся уравнением:

$$H_k(i, j) = w_{i_0, i} d_{i_0, i} + w_{i_0, j} d_{i_0, j} - w_{k, i} d_{k, i} - w_{k, j} d_{k, j} - w_{i_0, k} d_{i_0, k} \quad \forall k \in I''. \quad (9)$$

На основе данной теоремы можно полагать, что задача (1)–(5) эквивалентна следующей задаче, где максимизация осуществляется как по конфигурациям, так и по координатам точек Штейнера:

$$\max_{\substack{(x_i, y_i), i \in I'', \\ E \in F}} \sum_{k=n+1}^{2n-2} H_k, \quad (10)$$

где F — множество всех возможных конфигураций дерева Штейнера. Таким образом, начальное приближенное решение может быть построено путем последовательного выбора пары точек, обеспечивающей максимальную эффективность точки Штейнера.

3. Начальное решение. Алгоритм начального решения необходим для определения начальной конфигурации транспортной сети. Его идея состоит в изменении порядка максимизации и суммирования в выражении (10): $\max \sum_{k=n+1}^{2n-2} H_k \rightarrow \sum_{k=n+1}^{2n-2} \max H_k$.

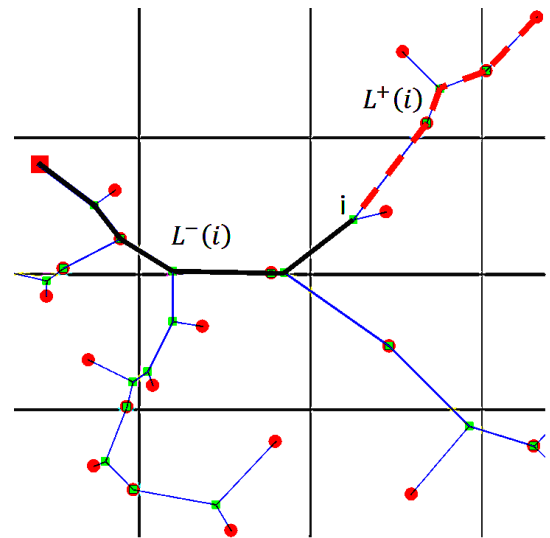


Рис. 1. Для вершины i отмечены два участка транспортной сети до максимально удаленной родительской вершины графа и до корня дерева, помеченных пунктирной и сплошной линией соответственно. $L^+(i)$ — длина участка, отмеченного пунктирной линией, $L^-(i)$ — сплошной линией

Алгоритм 1. Начальное решение

Вход: I' ;

Выход: G ;

- 1: $I_0 = I' \setminus \{i_0\}; V = I'; E = \emptyset; I'' = \emptyset$ // инициализация
- 2: **повторять**
- 3: $\{i, j\} = \arg \max_{i, j \in I_0} H_k(i, j)$;
- 4: $\tilde{k} = \text{three_point_problem}(i, j, i_0)$; // решение трехточечной задачи
- 5: $I'' = I'' \cup \{k\}$
- 6: $I_0 = (I_0 \setminus \{i, j\}) \cup \{k\}$ // добавление точки Штейнера в множество I_0
// на рассмотрение
- 7: $V = V \cup \{k\}$
- 8: $E = E \cup \{(i, k), (j, k)\}$ // добавление ребер из i и j в k
- 9: **пока** $|I_0| > 1$
- 10: $E = E \cup \{(k^*, i_0)\}$ // k^* — последняя добавленная точка Штейнера
- 11: $G = (V, E, W)$

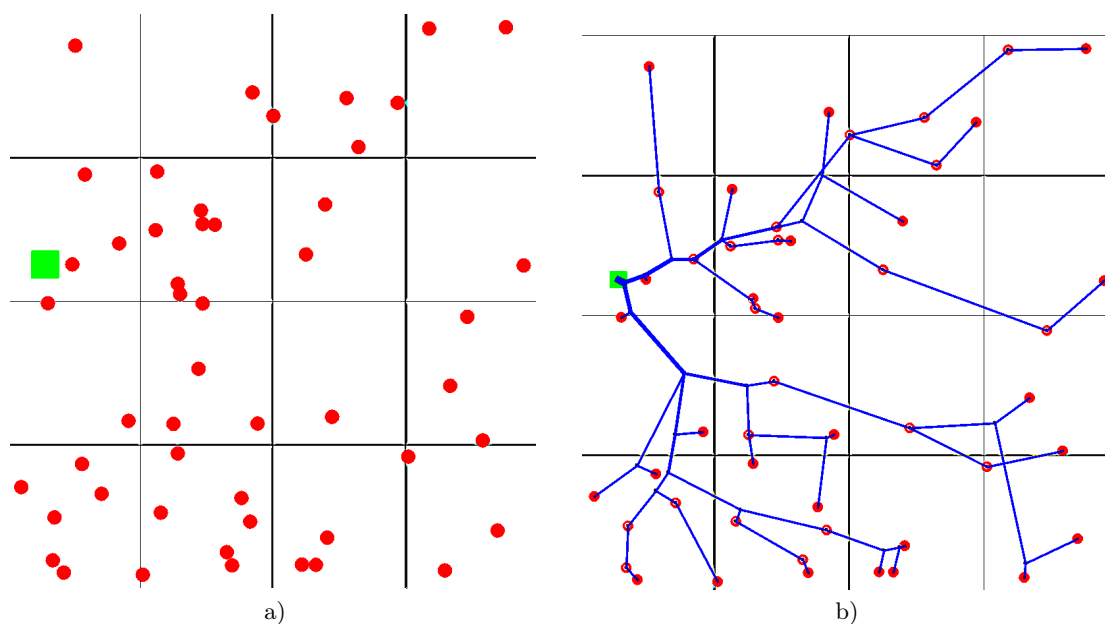


Рис. 2. Демонстрация начального решения: 50 терминальных вершин с корнем дерева, положение которых сгенерировано случайно (а), начальное приближенное решение (б)

Начальным приближенным решением является ориентированный граф, определяющий конфигурацию дерева, и некоторое размещение точек Штейнера. С помощью свойств минимального дерева Штейнера (2)–(5) определяется его конфигурация. Положение точки Штейнера для каждой пары вершин находится из решения трехточечной задачи, в которой третьей точкой является корень дерева.

Алгоритм решения трехточечной задачи описан в работе [9]. Выбирается пара вершин по условию максимума эффективности смежной к ним точки Штейнера: $\max_{i, j \in I'} H_k(i, j)$. Положение точки Штейнера влияет только на последние три слагаемых в уравнении (9). Следовательно, ищется такое положение точки Штейнера, при котором будет достигаться минимум суммы последних трех членов. После того как пара, удовлетворяющая условию, будет выбрана, эта пара вершин удаляется из множества I_0 , а точка Штейнера добавляется. Кроме того, создаются соответствующие ребра графа, соединяющие добавленную

точку Штейнера с выбранной парой вершин. Далее, выбирается следующая пара вершин из множества I_0 . После $N - 1$ подобных операций останется одна точка Штейнера, которая соединяется ребром с корнем дерева.

Результат работы алгоритма продемонстрирован на рис. 2. Как видно из рисунка, получившееся решение имеет пересечения, чего не должно быть в оптимальном дереве Штейнера согласно теореме, доказанной в работе [2]. Цель начального решения — ввести ориентированный взвешенный граф, определяющий транспортную сеть, чтобы на последующих этапах оптимизации изменять уже построенную сеть так, чтобы она удовлетворяла ограничениям и имела минимальное значение взвешенной длины. Таким образом, на этапе локальной оптимизации уточняется положение точек Штейнера.

4. Локальная оптимизация. Начальное решение никак не учитывает геометрических ограничений, поэтому на этапе локальной оптимизации на основе итерационного алгоритма строится локально оптимальное дерево Штейнера, удовлетворяющее ограничениям, с использованием конфигурации дерева, заданной после получения начального решения. Под локально оптимальным деревом Штейнера понимается дерево, в котором длину взвешенного дерева невозможно уменьшить изменением положений точек Штейнера.

Алгоритм 2. Локальная оптимизация

Вход: G ;

Выход: V_G^{new} ;

```

1:  $DP_{new} = \sum_{(i,j) \in E_G} w_{ij} * d_{ij} + a \left( \sum_{k \in I' \setminus \{i_0\}} \max(0, L^-(k) - limit) \right)^2$ 
2:  $DP_{old} = \infty$ ; // инициализация
3: повторять
4:    $DP_{old} = DP_{new}$ 
5:   для всех  $k \in I''$ 
6:      $\{M_k\} = \text{adjacent\_vertex}(k, E_G)$  // определение вершин, смежных с вершиной  $k$ 
7:      $\tilde{k}_{new} = \text{three\_point\_problem\_penalty}(M_k)$ ; // решение трехточечной задачи
8:      $E_{new} = \left\{ E \mid E(\{(i, k), (j, k), (k, s)\}) = \{(i, k_{new}), (j, k_{new}), (k_{new}, s)\} \right\}, \{i, j, s\} = M_k$ 
9:      $E_{new} \Rightarrow G_{new} = (V, E_{new}, W)$ 
10:     $G_{new} \Rightarrow DP_{new}, G = G_{new}$ 
11: пока  $\frac{DP_{new} - DP_{old}}{DP_{old}} > \epsilon$ ;
```

На основе начального решения имеется возможность рассчитать длину взвешенного дерева Штейнера, которая определяет характерное значение константы $a \approx D$ в выражении для штрафной функции (7). Следует заметить, что при построении начального приближения положение точки Штейнера выбиралось в предположении, что она смежна с корнем дерева. Однако существует только одна такая точка $k^* : (i_0, k^*) \in E$, поток от которой поступает в точку сбора. Следовательно, необходимо переопределить эффективность точки Штейнера, чтобы привести дерево к локально оптимальному:

$$H_k(i, j, s) = w_{si}d_{si} + w_{sj}d_{sj} - w_{ki}d_{ki} - w_{kj}d_{kj} - w_{sk}d_{sk}, \quad s = \text{Child}(k), \quad \{i, j\} = \text{Parents}(k),$$

где k — точка Штейнера; s — вершина графа, являющаяся дочерней для k ; i, j — вершины графа, являющиеся родительскими вершинами для k .

На каждой итерации для всех точек Штейнера k определяются смежные вершины M_k по заданной конфигурации дерева. Далее, решая трехточечную задачу для каждого множества M_k , мы определяем новое положение точек Штейнера. Следует заметить, что на этапе локальной оптимизации изменение положений точек Штейнера должно происходить не только исходя из длины взвешенного дерева, но и с учетом штрафных функций. Таким образом, точки Штейнера будут “чувствовать” свое положение

относительно самой удаленной терминальной вершины и корня дерева. Значит, при решении трехточечной задачи для каждой точки Штейнера будет минимизироваться следующее выражение, включающее в себя штрафную функцию:

$$w_{ki}d_{ki} + w_{kj}d_{kj} + w_{sk}d_{sk} + \sum_{l=i,j} a \left(\max(0, limit - d_{kl} - d_{sk} - L^+(l) - L^-(s)) \right)^2.$$

В качестве условия останова алгоритма используется сходимость целевой функции, в которой учитывается длина взвешенного дерева и штрафная функция DP , которая определена уравнением (8). Таким образом, на этапе локальной оптимизации происходит переопределение положения дополнительных точек так, чтобы для заданной конфигурации дерева был достигнут минимум функции DP . Результат работы алгоритма продемонстрирован на рис. 3. В общем случае исчезают не все пересечения. Следовательно, условия оптимальности дерева не выполнены. Дальнейшая оптимизация возможна только при изменении конфигурации транспортной сети.

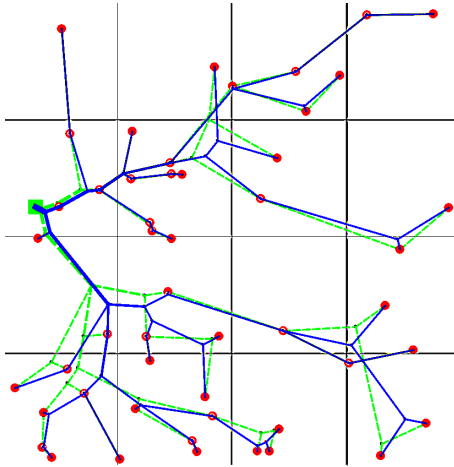


Рис. 3. Демонстрация решения после локальной оптимизации. Пунктирной линией отмечено начальное решение, сплошной линией — решение после локальной оптимизации

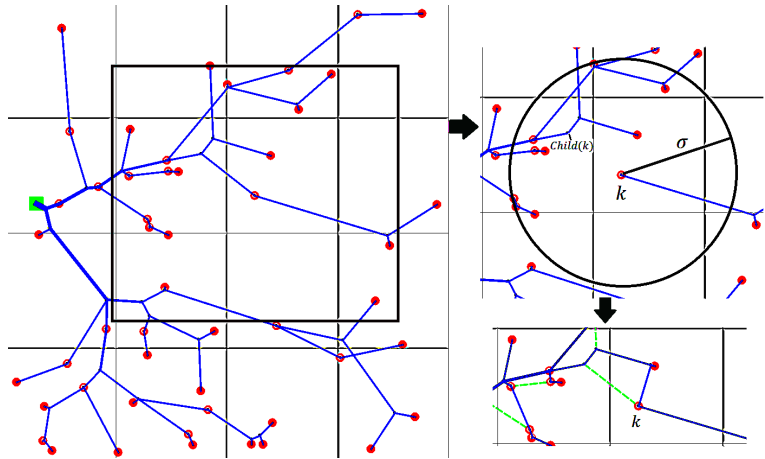


Рис. 4. Демонстрация шага на этапе глобальной оптимизации

5. Глобальная оптимизация. Результатом алгоритма 2 является локально минимальное взвешенное дерево Штейнера. На этапе глобальной оптимизации происходит изменение конфигурации дерева, заданного после начального решения. Для этих изменений зададим потенциальную функцию, определенную на множестве вершин $I = V \setminus \{i_0\}$, характеризующую стоимость транспортировки в текущей сети сбора потока от какой-либо вершины k с весом $t = w_k$ до точки сбора i_0 :

$$P_G(k, t) = P_G(Child(k), t) + d_{Child(k),k} [w_k - (w_k - t)]. \tag{11}$$

На этапе глобальной оптимизации для каждой вершины графа, принадлежащей множеству I , ищется лучший потомок. Поиск лучшего потомка посредством перебора всех вершин, принадлежащих I , является трудозатратным. Для того чтобы сузить область поиска, вводится потенциальная функция, определенная

уравнением (11) с последующим вычислением величины $\sigma = \frac{P_G(k, w_k^Y)}{w_k^Y}$, которая характеризует область

двумерного пространства, в которой следует искать лучшего потомка t для k -й вершины. Лучший потомок определяется исходя из значения функции (12). Данные пояснения продемонстрированы на рис. 4. Для того чтобы еще больше сократить область поиска, вводится множество вершин поддерева с корнем $k \in I$: $Subgraph(k) = Parents(k) \cup Subgraph(Parents(k))$, а также множество вершин, составляющих путь до корня дерева от вершины $k \in I$: $PathToRoot(k) = Child(k) \cup PathToRoot(Child(k))$.

Алгоритм 3. Глобальная оптимизация

Вход: G ;

Выход: G_{new} ;

1: $I_0 = V_G \setminus \{i_0\}$

2: **для всех** $k \in I_0$

3: $DP_{new} = \sum_{(i,j) \in E_G} w_{ij} * d_{ij} + a \left(\sum_{k \in I' \setminus \{i_0\}} \max(0, L^-(k) - limit) \right)^2$

4: $\sigma = \frac{P_G(k, w_k)}{w_k} \Rightarrow L(k) = \{v \in I_0 : d_{vk} \leq |\sigma|\}, v \notin Subgraph(k)$

5: **для всех** $ch \in PathToRoot(k)$

6: $w_{ch} = w_{ch} - w_k \Rightarrow W \rightarrow \tilde{W} (G \rightarrow \tilde{G}), E_{\tilde{G}} = E_G \setminus \{(k, Child(k))\}$

7: **для всех** $v \in L(k)$

8: $edges = find_edges(v)$

9: **для всех** $e \in edges$

10: $m = find_minimal_path(e, k) \Rightarrow c(m) = P_{\tilde{G}}(m, w_k)$

11: **если** $c(m) < w_k \sigma$ **то**

12: $E'_{\tilde{G}} = E_{\tilde{G}} \cup \{(k, m)\}, \tilde{G} \rightarrow \tilde{G}', \tilde{G}' \Rightarrow DP'_{new}$

13: **если** $DP'_{new} < DP_{new}$ **то**

14: $G_{new} = \tilde{G}'$

15: $G = G_{new}$

Далее ищется дополнение множества $Subgraph(k)$ до I . После этого из найденного дополнения выделяются вершины, удовлетворяющие условию $d_{vk} \leq \sigma, v \in I \setminus Subgraph(k)$. Для каждой вершины определяются ребра e , входящие и выходящие из нее. На каждом ребре e ищутся точки m , характеризующие кратчайшее расстояние от вершины k до данного ребра. Для каждой точки m считается функция, определяющая стоимость транспортировки потока w_k^V до корня дерева в новой конфигурации:

$$c(m) = -P_G(m, -w_k^V). \quad (12)$$

Вершина с минимальным значением функции $c(m)$ является наиболее подходящим потомком для k .

Транспортная сеть с новой конфигурацией удовлетворяет ограничениям, так как они учтены в целевой функции в виде штрафов (8). Этап глобальной оптимизации продемонстрирован на рис. 5. Из рисунка видно, что произошло изменение конфигурации дерева. Следовательно, необходимо дерево с новой конфигурацией привести снова к локально оптимальному. Таким образом, этапы локальной и глобальной оптимизации будут повторяться до сходимости значения функции DP . Конечный результат работы алгоритма (1)–(3) продемонстрирован на рис. 6 для примера, разобранный в предыдущих разделах.

6. Численные результаты. Заметим, что если на том же примере решать задачу нахождения минимального взвешенного дерева Штейнера без ограничений, то результат будет выглядеть иначе (рис. 7b).

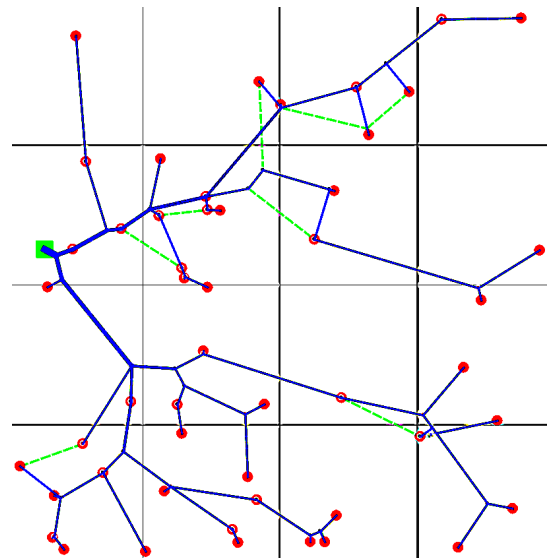


Рис. 5. Демонстрация решения после глобальной оптимизации. Пунктирной линией отмечено решение после локальной оптимизации, сплошной линией — после глобальной

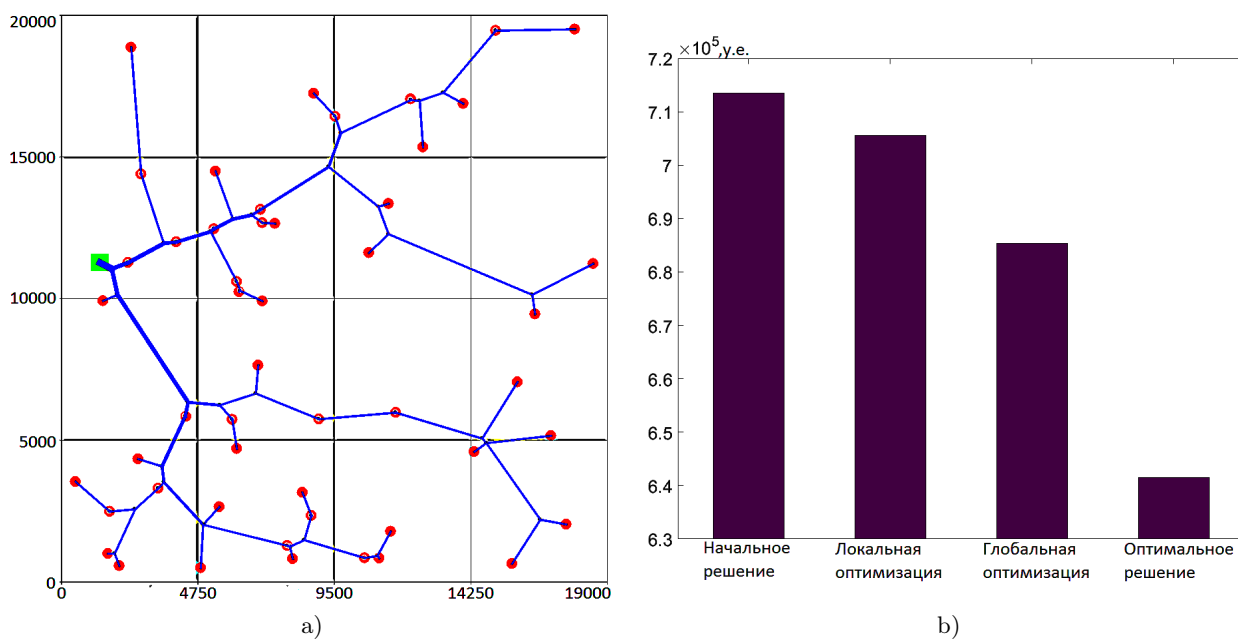


Рис. 6. Оптимальная транспортная сеть для разобранного примера с 50-ю вершинами и ограничением $limit = 27000$ м (а). График стоимости сети сбора для каждого из этапов оптимизации (b)

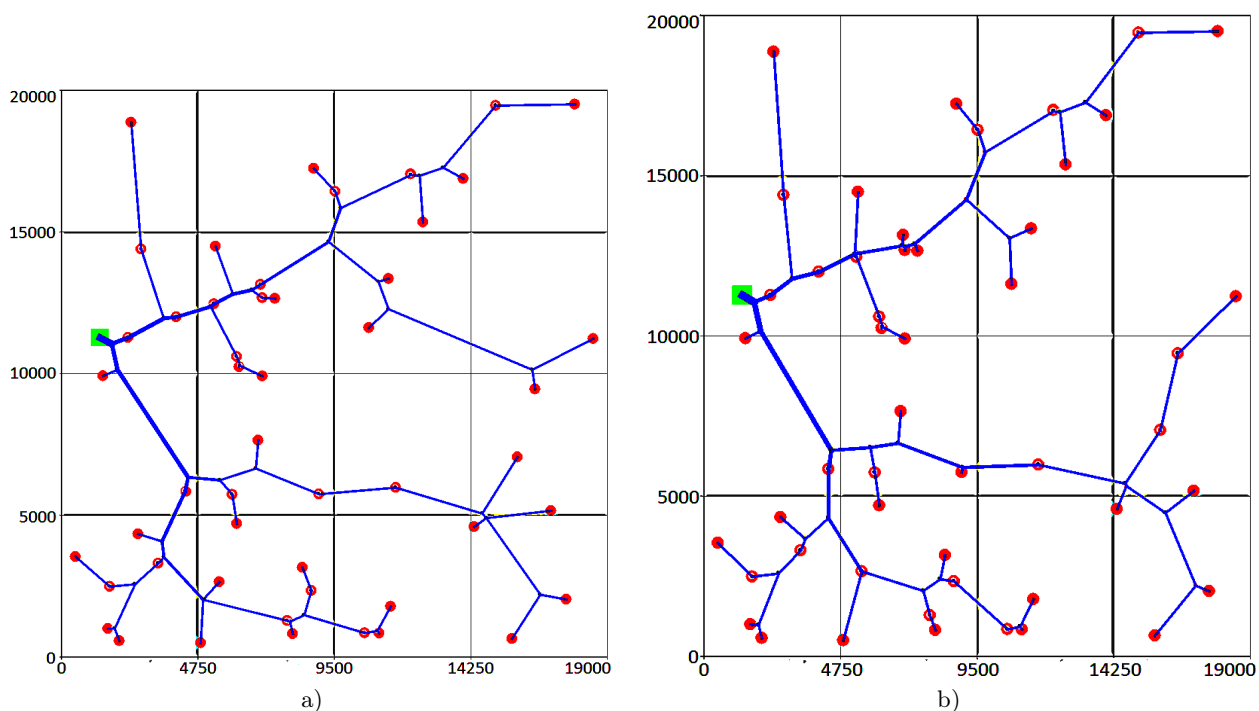


Рис. 7. Оптимальная транспортная сеть с ограничением $limit = 27000$ м (а).
Оптимальная транспортная сеть без ограничений (b)

Решение задачи без ограничений означает, что на этапах локальной и глобальной оптимизации условием остановки алгоритма является сходимость длины взвешенного дерева D , определенной уравнениями (1). Аналогичное влияние на конфигурацию дерева Штейнера имеет ограничение на максимальную суммарную длину по ребрам от терминальной вершины до корня дерева в случае значительного увеличения значения этого ограничения. В предыдущих разобранных примерах это ограничение было постоянным ($limit = 27000$). На рис. 8 продемонстрировано влияние ограничительного параметра на конфигурацию сети сбора. Видно, что с уменьшением ограничительного параметра ветви дерева Штейнера присоединяются ближе к корню, а некоторые участки коммуникации спрямляются для удовлетворения ограничения.

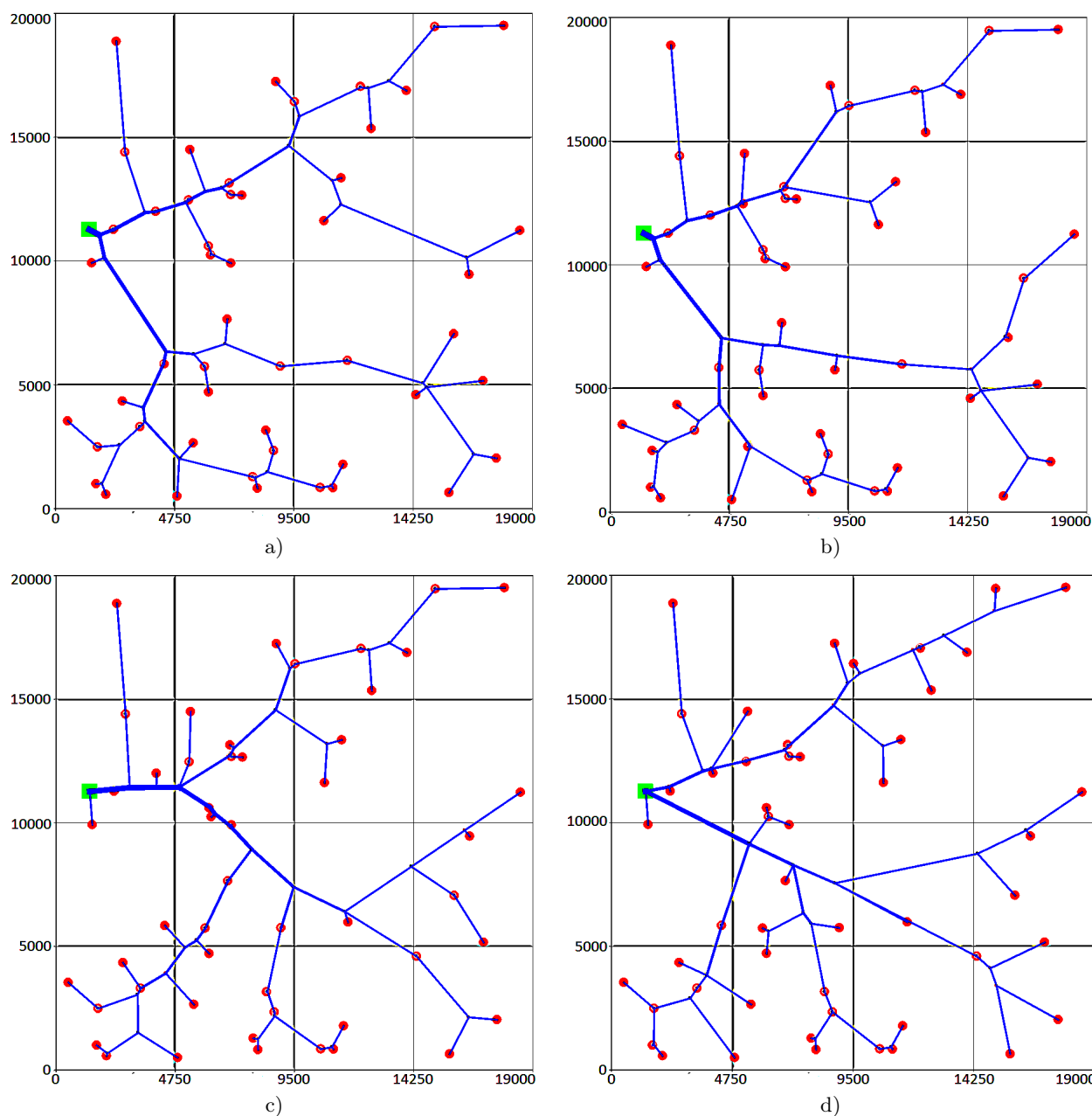


Рис. 8. $limit = 27000$ (a), $limit = 23000$ (b), $limit = 21000$ (c), $limit = 19400$ (d)

7. Заключение. Разработан эвристический алгоритм для решения задачи построения оптимального транспортного пути с ограничением для случая с множеством источников и одним стоком. Алгоритм объединяет в себе два подхода оптимизации, которые позволяют проводить локальный и глобальный поиск минимального взвешенного дерева Штейнера за полиномиальное время. Кроме того, алгоритм использует метод штрафных функций для учета ограничений на максимальную длину участка коммуникации от терминальной вершины до корня дерева.

Представлен анализ численных результатов, из которых видно, что ограничивающий параметр ($limit$) влияет на конфигурацию оптимальной транспортной сети. Данный алгоритм может быть расширен и применен для проектирования любой сети сбора (транспортная сеть сбора нефти, газа, электросетей). Одним из возможных расширений модели может быть учет неоднородности территории.

СПИСОК ЛИТЕРАТУРЫ

1. Xue G., Lillys T.P., Dougherty D.E. Computing the minimum cost pipe network interconnecting one sink and many sources // SIAM Journal on Optimization. 1999. 10, N 1. 22–42.

2. *Лотарев Д.Т., Супрун А.В., Уздемир А.П.* Локальная оптимизация в задаче Штейнера на евклидовой плоскости // Автомат. и телемех. 2004. № 7. 60–70.
3. *Xia Q.* Numerical simulation of optimal transport paths // Second International Conference on Computer Modeling and Simulation. Washington, DC: IEEE Press, 2010. doi 10.1109/ICCMS.2010.30.
4. *Costa A.M., Cordeau J.-F., Laporte G.* Fast heuristics for the Steiner tree problem with revenues, budget and hop constraints // European Journal of Operational Research. 2008. **190**, N 1. 68–78.
5. *Sinnl M., Ljubić I.* A node-based layered graph approach for the Steiner tree problem with revenues, budget and hop-constraints // Mathematical Programming Computation. 2016. **8**, N 4. 461–490.
6. *Gouveia L., Leitner M., Ljubić I.* Hop constrained Steiner trees with multiple root nodes // European Journal of Operational Research. 2014. **236**, N 1. 100–112.
7. *Трифонов А.Г.* Постановка задачи оптимизации и численные методы ее решения. http://matlab.exponenta.ru/optimiz/book_2/index.php.
8. *Лотарев Д.Т., Уздемир А.П.* Размещение транспортных сетей на неоднородной территории // Автомат. и телемех. 2002. № 7. 117–127.
9. *Лотарев Д.Т., Уздемир А.П.* Преобразование задачи Штейнера на евклидовой плоскости к задаче Штейнера на графе // Автомат. и телемех. 2005. № 10. 80–92.

Поступила в редакцию
30.03.2017

A New Algorithm for the Optimization of Transport Networks Subject to Constraints

A. A. Ananov¹, P. V. Lomovitskiy², D. V. Uzhegov³, and A. N. Khlyupin⁴

¹ *Moscow Institute of Physics and Technology Center for Engineering and Technology; ulitsa Pervomayskaya 5, Dolgoprudny, 141700, Russia; Engineer, e-mail: ananov.aa@cet-mipt.ru*

² *Moscow Institute of Physics and Technology Center for Engineering and Technology; ulitsa Pervomayskaya 5, Dolgoprudny, 141700, Russia; Engineer, e-mail: lomovitskiy.pv@cet-mipt.ru*

³ *Moscow Institute of Physics and Technology Center for Engineering and Technology; ulitsa Pervomayskaya 5, Dolgoprudny, 141700, Russia; Engineer, e-mail: uzhegov.dv@cet-mipt.ru*

⁴ *Moscow Institute of Physics and Technology Center for Engineering and Technology; ulitsa Pervomayskaya 5, Dolgoprudny, 141700, Russia; Head of Research Group, e-mail: khlyupin.an@cet-mipt.ru*

Received March 30, 2017

Abstract: A new heuristic algorithm of finding a minimum weighted Steiner tree is proposed. A transport network can be represented in the form of a directed weighted Steiner tree. Constraints are imposed on the maximal total length of communications from any terminal vertex to the root of the tree. A penalty function method is used to take the constraints into account. The effect of model parameters on the optimal network geometry is analyzed.

Keywords: transport networks, Steiner problem, graph algorithms, optimization, constrained problems.

References

1. G. Xue, T. P. Lillys, and D. E. Dougherty, “Computing the Minimum Cost Pipe Network Interconnecting One Sink and Many Sources,” *SIAM J. Optim.* **10** (1), 22–42 (1999).
2. D. T. Lotarev, A. V. Suprun, and A. P. Uzdemir, “Local Optimization in the Steiner Problem on the Euclidean Plane,” *Avtom. Telemekh.*, No. 7, 60–70 (2004) [*Autom. Rem. Contr.* **65** (7), 1089–1098 (2004)].
3. Q. Xia, “Numerical Simulation of Optimal Transport Paths,” in *Proc. Second Int. Conf. on Computer Modeling and Simulation, Sanya, China, January 22–24, 2010* (IEEE Press, Washington, DC, 2010), doi 10.1109/ICCMS.2010.30
4. A. M. Costa, J.-F. Cordeau, and G. Laporte, “Fast Heuristics for the Steiner Tree Problem with Revenues, Budget and Hop Constraints,” *Eur. J. Oper. Res.* **190** (1), 68–78 (2008).
5. M. Sinnl and I. Ljubić, “A Node-Based Layered Graph Approach for the Steiner Tree Problem with Revenues, Budget and Hop-Constraints,” *Math. Prog. Comput.* **8** (4), 461–490 (2016).

6. L. Gouveia, M. Leitner, and I. Ljubić, “Hop Constrained Steiner Trees with Multiple Root Nodes,” *Eur. J. Oper. Res.* **236** (1), 100–112 (2014).
7. A. G. Trifonov, *Formulation of the Optimization Problem and Numerical Methods of Its Solution*, http://matlab.exponenta.ru/optimiz/book_2/index.php. Cited April 24, 2017.
8. D. T. Lotarev and A. P. Uzdemir, “Location of Transport Nets on a Heterogeneous Territory,” *Avtom. Telemekh.*, No. 7, 117–127 (2002) [*Autom. Rem. Contr.* **63** (7), 1146–1154 (2002)].
9. D. T. Lotarev and A. P. Uzdemir, “Conversion of the Steiner Problem on the Euclidean Plane to the Steiner Problem on Graph,” *Avtom. Telemekh.*, No. 10, 80–92 (2005) [*Autom. Rem. Contr.* **66** (10), 1603–1613 (2005)].