ИНФОРМАЦИОННЫЕ И ВЫЧИСЛИТЕЛЬНЫЕ ТЕХНОЛОГИИ

УДК 519.85, 519.17

Научная статья

Кустовая оптимизация высокого ранга оптимальности потоковых сетей

М. Б. Абазоков, В. Ч. Кудяев

Институт прикладной математики и автоматизации КБНЦ РАН, 360000,
КБР, г. Нальчик, ул. Шортанова, 89 А., Россия

E-mail: ipma@niipma.ru

Решена задача построения больших потоковых сетей высокого ранга оптимальности на основе кустовой оптимизации, связывающей с каждой вершиной сети её фрагмент, имеющий определенную размерность и позволяющий на всех фрагментах достичь более высокого ранга оптимальности за заданное время решения задачи на компьютере.

Ключевые слова: терминальная потоковая сеть, оптимизация структуры, ранги экстремумов, метод ранговой оптимизации, метод кустовой оптимизации, компьютерное проектирование, сеть регионального и межрегионального водоснабжения, снижение размерности задачи.

DOI: 10.26117/2079-6641-2021-37-4-104-118

Поступила в редакцию: 10.11.2021

В окончательном варианте: 12.12.2021

Для цитирования. Абазоков М. Б., Кудяев В. Ч. Кустовая оптимизация высокого ранга оптимальности потоковых сетей // *Вестник КРАУНЦ. Физ.-мат. науки.* 2021. Т. 37. № 4. С. 104-118. DOI: 10.26117/2079-6641-2021-37-4-104-118

Контент публикуется на условиях лицензии Creative Commons Attribution 4.0 International (<https://creativecommons.org/licenses/by/4.0/deed.ru>)

© Абазоков М. Б., Кудяев В. Ч., 2021

1. Задача синтеза терминальной потоковой сети

Задача синтеза состоит в следующем

$$z(v) = \sum_{ij \in D} c_{ij}(v_{ij}) l_{ij} \rightarrow \min, \quad (1)$$

$$\sum_{i \in \Gamma_j^+} v_{ij} - \sum_{k \in \Gamma_j^-} v_{jk} = g_i, \quad \forall j \neq 1 \in B, \quad (2)$$

$$\sum_{j \in \Gamma_j^-} v_{1j} = Q, \quad (3)$$

$$v_{ij} \geq 0, \quad \forall (i, j) \in D \quad (4)$$

Финансирование. Исследование выполнялось без финансовой поддержки фондов.

где $\Gamma(B, D)$ – заданный оргграф возможных соединений вершин сети; B и D – множества его вершин и дуг; v_{ij} , c_{ij} , l_{ij} – искомое значение величины потока, заданные удельная стоимость и длина (i, j) -й дуги; Q – заданный поток в сеть; g_i – заданный расход потока в i -м узле сети. Γ_j^+ и Γ_j^- – соответственно множества дуг, входящих и исходящих. Целевая функция (1) отражает стоимость потоков сети, ограничение (2) есть уравнение неразрывности потока, соотношение (3) говорит о том, что узел 1 является источником потока. Функция $c_{ij}(v_{ij})$ непрерывна, строго вогнута и возрастает, $c_{ij}(0) = 0$. В связи с этим экстремумы могут достигаться лишь в угловых точках транспортного многогранника (2)-(4). Задача, как известно, является NP -полной существенно-многоэкстремальной задачей. Для нелокального решения задачи предложено несколько методов.

2. Существующие методы решения задачи

Рассмотрим наиболее известные методы решения задачи: метод погружения и метод ветвей и границ.

2.1. Метод погружения

Метод погружения для решения задачи минимизации вогнутой функции на выпуклом многограннике изложен в работе [1]. В работах [2], [3, с. 65-83] метод модифицирован для решения задачи на транспортном многограннике. Суть метода состоит в следующем. Из текущей угловой точки выпуклого многогранника ограничений переходим в такую смежную к ней, в которой значение целевой функции меньше, чем в исходной. При достижении угловой точки транспортного многогранника, в которой значение целевой функции меньше, чем в смежных к ней, проводим из этой угловой точки прямые, проходящие по ребрам многогранника ограничений через смежные к ней точки вне многогранника, и определяем на них точки, в которых целевая функция достигает того же значения, что и в исходной угловой точке локального экстремума. По этим точкам проводим гиперплоскость и отделяем от многогранника ограничений полученный симплекс, вершиной которого является точка локального экстремума, запоминая значение целевой функции в этой точке. Далее определяется на полученном многограннике следующая точка экстремума и т.д. до тех пор, пока на очередной итерации текущий многогранник не будет целиком погружен в отсекаемую часть. После этого определяется та из отсеченных угловых точек, в которой значение локального экстремума целевой функции минимально.

В [4, с. 131-132] отмечен следующий недостаток метода: «При $n = 51$ возникали трудности из-за медленной сходимости вычислительного процесса отсечений. ... Было предложено осуществлять сдвиг отсекающей гиперплоскости на некоторую величину h , что привело к труднорешаемой проблеме выбора данной величины: при повышенном значении h пропускаются локальные экстремумы целевой функции, а при малых h процесс оптимизации требует чрезмерного машинного времени».

2.2. Метод ветвей и границ

Метод ветвей и границ, представленный в работах [5], [6, с. 206-216], основан на последовательном делении множества допустимых решений с отбрасыванием подмножеств, которые имеют высокую нижнюю оценку минимизируемой функции. Теоретически это обеспечивает получение глобального минимума, но с заданной допустимой погрешностью. В [4] отмечено, что если эта величина мала, то процесс сходится медленно. Добавим, что целевая функция аппроксимируется вогнутой кусочно-линейной функцией с несколькими звеньями, что уже дает значительную погрешность в решении задачи минимизации вогнутой функции на транспортном многограннике. Увеличение количества звеньев приводит к резкому увеличению времени решения задачи.

3. Ранги экстремумов

Для существенно многоэкстремальных задач большой размерности локальный экстремум не информативен. Поэтому в работах [7, 8, 9, 10] введена градация экстремумов и представлены методы нелокальной оптимизации задачи (1)-(4), ориентированные на получение экстремума достижимого ранга за экономически оправданное время решение задачи на ЭВМ.

Определение 1. Точкой экстремума P -го ранга задачи (1)-(4) назовем такую угловую точку $\{v_{ij}^*\}_{ij \in D}$ транспортного многогранника (2)-(4), которая является точкой глобального минимума на выпуклой линейной комбинации вершин многогранника, имеющих смежность в промежутке $[1, P]$ к этой точке.

Любой вершине транспортного многогранника, как известно, соответствует базисное решение задачи, т.е. некоторое остовное дерево заданного орграфа сети $\Gamma(B, D)$. $[1, P]$ – это множество угловых точек многогранника достижимых из заданной угловой точки. Целью является построение остовного дерева графа $\Gamma(B, D)$, имеющего наибольший ранг.

Для того, чтобы перевести это определение на язык теории сетей было введено понятие фрагментов P -го ранга сети.

Определение 2. P -фрагментом сети назовем замыкаемые любыми P хордами графа $\Gamma(B, D)$ P контуров на текущем остовном дереве T графа $\Gamma(B, D)$.

Теорема (УСЛОВИЕ РАНГОВОЙ ОПТИМАЛЬНОСТИ ДЛЯ СЕТЕВОЙ ЗАДАЧИ).

Для того, чтобы решение $\{v_{ij}^*\}_{ij \in D}$, где $v_{ij} = 0 \ \forall (i, j) \notin T$, задачи (1)-(4) было экстремумом P -го ранга, необходимо и достаточно, чтобы оно было оптимально по всем фрагментам P -го ранга T, P на T , т.е.

$$\sum_{(i,j) \in T, P} c_{ij}(v_{ij}^*)l_{ij} \leq \sum_{(i,j) \in D} c_{ij}(v_{ij})l_{ij}$$

где $\{v_{ij}\}_{ij \in D}$ – любое допустимое решение задачи, но такое, что $v_{ij} = v_{ij}^* \ \forall (i, j) \notin T, P$.

Доказательство теоремы приведено в работе [7].

4. Суть метода ранговой оптимизации терминальной потоковой сети

Метод ранговой оптимизации представлен в работах [7, 8, 9, 10]. Его суть состоит в сведении оптимизации сети к оптимизации ее фрагментов все более высокого ранга.

При оптимизации 1-го ранга на очередной итерации выделяется очередная хорда (i, j) и соответствующий ей фрагмент 1-го ранга (контур сети) на T . Проведя оптимизацию полученного фрагмента, определяем очередную независимую переменную относительно полученного решения и переходим к оптимизации соответствующего фрагмента. Процесс оптимизации 1-го ранга системы завершается при получении решения, которое не может быть улучшено внесением в оптимизируемое текущее остовное дерево любой из его хорд и соответствующим изменением потоков по образованному при этом контуру.

Далее переходим к оптимизации 2-го ранга. Для этого выделяем на каждой очередной итерации очередную пару хорд и соответствующие им фрагменты 1-го ранга. В том случае, когда эти фрагменты пересекаются, формируем фрагмент 2-го ранга – объединение двух фрагментов 1-го ранга. Решаем задачу оптимизации фрагмента 2-го ранга и переходим к следующей итерации.

Оптимизация системы прекращается при достижении заданного оптимума P -го ранга либо заданного времени решения задачи на компьютере.

Таким образом, метод ранговой оптимизации сетевых систем состоит в сведении оптимизации сети к оптимизации ее фрагментов P -го ранга. Рассмотрение только связанных P -фрагментов позволяет существенно снизить размерность задачи.

5. Определение связности P -контуров сети

При решении задачи P -го ранга важной подзадачей является определение связности P контуров, замыкаемых P хордами, вносимыми в текущую потоковую сеть при продвижении к оптимуму P -го ранга.

Рассмотрим множества $x_1, x_2, \dots, x_n$ объединенные в одно множество X (т.е. $X = \{x_1, x_2, \dots, x_n\}$).

Определение 3. Назовем множество X несвязным, если его можно разделить на два подмножества X_1 и X_2 , так что $X_1 \cup X_2 = X$ и $X_1 \cap X_2 = \emptyset$, чтобы $\forall x_i \in X_1$ и $\forall x_j \in X_2$ выполнялось $x_i \cap x_j = \emptyset$ для. В противном случае множество X назовем связным.

В качестве элементов множеств рассмотрим P -контур сети. На рис. 1 представлен вариант связанных и несвязных контуров.

Рис. 1. Связные контуры (слева) и несвязные контуры (справа)

Алгоритм определения связности P -контуров сети состоит в следующем:

Пусть у нас есть множество $X = \{x_1, x_2, \dots, x_n\}$, где x_i – это контур (множество ветвей) соответствующий i -й хорде для текущего остовного дерева.

1. Если X состоит из 1-го элемента, то множество X связно. Конец алгоритма.
 2. Запускаем цикл проверки. Перебираем все элементы (контура) со 2-го до P -го.
 - 2.1. Если x_1 и x_i пересекаются, то $x_1 := x_1 \cup x_i$ и переходим к пункту 1. Если нет, то продолжаем перебор.
- Если перебраны все элементы и у множества x_1 не было пересечений ни с одним x_i , $i = 2, 3, \dots, P$, то множество несвязно. Конец алгоритма.

Блок-схема данного алгоритма представлена на рис. 2.

Рис. 2. Блок-схема алгоритма определения связности P -контуров сети

6. Увеличение ранга оптимальности подсетей P -оптимальной терминальной потоковой сети

6.1. Суть метода

Обозначим T_n^P время компьютерного проектирования n -вершинной терминальной потоковой сети (ТПС) P -го ранга оптимальности на заданном избыточном базовом графе (БГ) возможных соединений её вершина друг с другом.

Покажем, что T_n^P имеет экспоненциальный рост по P .

Действительно, поскольку при оптимизации осуществляется направленный перебор связных P -фрагментов сети, т.е. фрагментов, образуемых внесением в

текущую оптимизированную сеть любых P хорд БГ и удалением из сети встречных им дуг, то

$$T_n^P = \eta C_n^P, \quad (5)$$

где η – постоянный коэффициент – доля связанных P -фрагментов сети умноженная на среднее время вычисления стоимости одного P -фрагмента. Получим

$$\frac{T_n^{P+1}}{T_n^P} = \frac{C_n^{P+1}}{C_n^P} = \frac{n-P}{P+1}, \quad \frac{T_n^{P+2}}{T_n^P} = \frac{(n-P-1)(n-P-2)}{(P+1)(P+2)}.$$

В общем случае, получим

$$\frac{T_n^{P+k}}{T_n^P} = \left(\frac{n-P-k+1}{P+1} \right) \left(\frac{n-P-k+2}{P+2} \right) \dots \left(\frac{n-P-1}{P+k-1} \right) \left(\frac{n-P}{P+k} \right). \quad (6)$$

Выбрав наименьший сомножитель числителя и наибольший знаменателя, получим

$$\frac{T_n^{P+k}}{T_n^P} > \left(\frac{n-P-k+1}{P+k} \right)^k.$$

Поскольку для больших сетей $p, k \ll n$, получим

$$\frac{T_n^{P+k}}{T_n^P} \sim O \left(\left(\frac{n}{P+k} \right)^k \right). \quad (7)$$

Пусть T – заданное время решение задачи синтеза ТПС. Выделим время $T/2$ на решение задачи методом динамической декомпозиции и ранговой оптимизации всей сети (см. выше, а также [10]). Пусть за это время на компьютере построена сеть P -го ранга. Тогда в силу (7) за оставшееся время $T/2$ не может быть построена сеть $(P+1)$ -го ранга. Поэтому для дальнейшего продвижения в оптимизации сети применим подход нелокальной оптимизации аналогичный подходу в методе «бегущей волны оптимизации» для снижения размерности и решения задачи оптимального управления [11].

Для снижения размерности задачи синтеза сети будем проводить дальнейшую оптимизацию $(P+1)$ -го ранга и более высоких рангов не на всей n -вершинной сети, а на ее связанных фрагментах каждый из которых сопоставляется одной из вершин сети.

Для реализации этого подхода с каждой вершиной i следует связать фрагмент Φ_i^{P+1} сети, количество вершин которого не превышает некоторой искомой величины m , такой, что время решения задачи синтеза на нем сети $(P+1)$ -го ранга с соблюдением граничных условий с остальной сетью было в n раз меньше времени $T/2$, т.е.:

$$\frac{T_n^P}{T_m^{P+1}} = \frac{C_n^P}{C_m^{P+1}} \cong n, \quad P+1 < m < n. \quad (8)$$

Каждый фрагмент Φ_i^{P+1} пересекается при этом с m_i фрагментами связанными с входящими в Φ_i^{P+1} вершинами и, таким образом, выполняется необходимое условие оптимальности сети $(P+1)$ -го ранга: *любой фрагмент сети Φ_i^{P+1} , $i = 1, 2, \dots, n$, $(P+1)$ оптимален.*

Очевидно, что для приближения условия к достаточному следует определить предельное (наибольшее) значение m , при котором выполняется (8).

6.2. Оценка предельного количества вершин Φ_i^{P+1} -фрагментов сети

Каждый фрагмент Φ_i^{P+1} представляет собой часть остовного дерева текущей оптимизируемой сети, корневой вершиной которого является i -я вершина сети. Эта часть должна содержать максимально приближенное к значению m количество вершин для выполнения условия (8). Определим это максимальное значение:

$$\frac{T_n^P}{T_m^{P+1}} = \frac{n!(P+1)!(m-P-1)!}{P!(n-P)!m!} =$$

$$= \left(\frac{P+1}{m-P}\right) \left(\frac{n-P+1}{m-P+1}\right) \left(\frac{n-P+2}{m-P+2}\right) \dots \left(\frac{n-1}{m-1}\right) \frac{n}{m}. \quad (9)$$

Оценим эту величину снизу:

$$\frac{T_n^P}{T_m^{P+1}} > \left(\frac{P+1}{m-P}\right) \left(\frac{n}{m}\right)^P. \quad (10)$$

Тогда предельное значение m определяется системой:

$$\begin{cases} \left(\frac{P+1}{m-P}\right) \left(\frac{n}{m}\right)^P \geq n, \\ \left(\frac{P+1}{m+1-P}\right) \left(\frac{n}{m+1}\right)^P < n. \end{cases} \quad (11)$$

Итак, из системы (11) определяется предельное значение m .

6.3. Построение Φ_i^{P+1} -фрагментов сети

Далее следует построить способ выделения на дереве, корневой вершиной которого является вершина i , $i = 1, 2, \dots, n$, такой его связной части, количество вершин которой максимально приближено к m . Для этого будем использовать понятие достижимости вершин графа из заданной вершины [12, с. 29–33]. Процесс построения дерева при этом будет состоять в следующем.

Определяем 1-достижимые вершины D_i^1 из вершины i и их количество $|D_i^1|$. Если $|D_i^1| = 0$, т.е. вершина i есть тупиковая вершина текущей оптимизируемой сети, то переходим к формированию фрагмента сети для следующей вершины. В противном случае определяем 2-достижимые вершины и их количество $|D_i^2|$. Если $|D_i^1| + |D_i^2| < m$ и $|D_i^2| = 0$, то продолжаем процесс построения Φ_i . Процесс подключения очередного слоя вершины (3-достижимых и т.д.) продолжается до тех пор пока не будет выполнено условие

$$|D_i^l| \neq 0, \quad |D_i^{l+1}| = 0, \quad \sum_{j=1}^l |D_i^j| \leq m, \quad \text{либо} \quad \sum_{j=1}^l |D_i^j| \leq m \quad \text{и} \quad \sum_{j=1}^{l+1} |D_i^j| > m. \quad (12)$$

Далее осуществляется процесс оптимизации $(P+1)$ -го ранга на Φ_i . После этого переходим к формированию следующего фрагмента и так до тех пор, пока не будут оптимизированы все фрагменты. Подчеркнем еще раз, что фрагменты, построенные таким образом, пересекаются – каждый фрагмент пересекается со всеми фрагментами, корневыми вершинами которых является вершины сети,

входящие во фрагмент. Именно поэтому вариация любого фрагмента переносятся на другие фрагменты и тем самым является нелокальной, что существенно для задачи минимизации вогнутой функции на многограннике, к каковой относится задача синтеза ТПС.

Аналогичным образом могут быть оценены и времена T_n^{P+2} , T_n^{P+3} , и вообще, T_n^{P+k} . В этих случаях, как не трудно понять, Φ_i должна уменьшаться. Существует, конечно, предел уменьшению фрагментов, т.к. m должно превышать ранг.

6.4. Определение целевого ранга оптимальности на Φ_i^{P+1} -фрагментах сети

Пусть известно, что за время $T/2$ компьютерной системой построена n -вершинная потоковая сеть P -го ранга оптимальности. Следует определить, обладая только этой информацией, достижимый ранг оптимальности $\hat{P}$ для сетей с количеством вершин $\hat{n} < n$. Рассмотрим случай $\hat{n} < n$ (аналогичным образом решается и определение $\hat{P}$ для $\hat{n} > n$).

Для этого следует определить такое наибольшее $\hat{n}$ с которого на сетях с $\hat{n} < n$ будет достигнут по крайней мере $(P+1)$ -й ранг оптимальности. Для этого должны выполняться следующие условия

$$\frac{T_n^P}{T_{\hat{n}}^{P+1}} \geq 1, \quad \frac{T_n^P}{T_{\hat{n}+1}^{P+1}} < 1.$$

Тогда получаем:

$$\begin{cases} \frac{C_n^P}{C_{\hat{n}}^{P+1}} = \frac{n!(P+1)!(\hat{n}-P-1)!}{P!(n-P)!\hat{n}!} = \left(\frac{P+1}{\hat{n}-P}\right) \left(\frac{n-P+1}{\hat{n}-P+1}\right) \left(\frac{n-P+2}{\hat{n}-P+2}\right) \dots \left(\frac{n-1}{\hat{n}-1}\right) \frac{n}{\hat{n}} \geq 1, \\ \frac{C_n^P}{C_{\hat{n}+1}^{P+1}} = \frac{n!(P+1)!(\hat{n}-P)!}{P!(n-P)!(\hat{n}+1)!} = \left(\frac{P+1}{\hat{n}-P+1}\right) \left(\frac{n-P+1}{\hat{n}-P+2}\right) \left(\frac{n-P+2}{\hat{n}-P+3}\right) \dots \left(\frac{n-1}{\hat{n}}\right) \frac{n}{\hat{n}+1} < 1. \end{cases}$$

Оценивая правые части ограничения сверху, получим

$$\begin{cases} \frac{T_n^P}{T_{\hat{n}}^{P+1}} = \left(\frac{P+1}{\hat{n}-P}\right) \left(\frac{n}{\hat{n}}\right)^P \geq 1 \\ \frac{T_n^P}{T_{\hat{n}+1}^{P+1}} = \left(\frac{P+1}{\hat{n}-P+1}\right) \left(\frac{n}{\hat{n}}\right)^P < 1 \end{cases} \quad (13)$$

Например, в соответствии с (13) для $n = 100$ получим $\hat{n} = 45$.

7. Алгоритм кустовой оптимизации

Алгоритм кустовой оптимизации:

1. Задаются следующие параметры: минимальное и максимальное количество вершин во фрагменте, а также ранг оптимизации.
2. Проводится *проход кустовой оптимизации* до отсутствия улучшений на всех кустовых фрагментах графа.

Алгоритм прохода кустовой оптимизации:

1. Берутся все вершины графа по очереди.
2. Если текущая вершина является висячей, то переходим к пункту 1.
3. Из текущей вершины методом обхода в ширину (см. рис. 3) находятся вершины будущего фрагмента, т.е.
 - 3.1. Находятся и добавляются слои вершин до тех пор, пока количество всех вершин во всех слоях не станет больше заданного максимального количества вершин во фрагменте.
 - 3.2. Удаляется последний слой, т.к. количество всех вершин во всех слоях больше допустимого.
 - 3.3. Если (после удаления) количество всех вершин во всех слоях меньше заданного минимального количества вершин во фрагменте, то переходим к пункту 1; иначе выдаются все вершины во всех слоях.
4. Отдельно от базового графа формируется фрагмент (куст) из выданных (в пункте 3.3) вершин, содержащий все выданные вершины и все дуги базового графа, соединяющие эти вершины.
5. Проводим оптимизацию заданного ранга на данном фрагменте. Если оптимизация дала улучшение, то хорды фрагмента, ставшие ветвями, добавляют в текущее остовное дерево; а ветки фрагмента, ставшие хордами, удаляются из текущего остовного дерева. Переходим к пункту 1.

Фрагмент графа – это, по сути, отдельный граф, являющийся подграфом базового графа, содержащий все выданные в пункте 3.3 вершины базового графа и все дуги между ними. Для наглядности, ниже, на рис. 3, представлена схема формирования слоев фрагмента из вершин базового графа, основанная на методе обхода в ширину по ветвям графа.

Рис. 3. Схема формирования слоев фрагмента из текущей корневой вершины.

8. Результаты вычислительного эксперимента

8.1. Результаты прямой оптимизации

Проведена прямая оптимизация 3-го и 4-го рангов на ПБГ при $n = 100$. Результаты представлены в табл. 1.

Таблица 1

Оптимизация	1 оптимизация	Доведение до 2 оптимальности	Доведение до 3 оптимальности	Доведение до 4 оптимальности
Стоимость сети «до»	264,111	229,697	227,763	224,223
Суммарное количество проходов (всех рангов)	4	5	12	1
Суммарное количество улучшений (всех рангов)	88	15	41	0
Улучшение стоимости	34,414	1,934	3,54	0
Время (час:мин:сек, миллисек)	00:00:25,042	00:05:50,023 (00:06:15,065 если с начала)	17:10:26,901 (17:16:41,966 если с начала)	307:50:46,444 (325:07:28,410 если с начала)
Стоимость сети «после»	229,697	227,763	224,223	224,223

На рисунке (см. Приложение, рис. 6) приведены: начальное остовное дерево (а), деревья 1-го, 2-го, 3-го, 4-го рангов оптимальности (б, в, г, д соответственно).

Как видно из таблицы, общее время построения 3-оптимальной потоковой сети на плотном базовом графе (ПБГ) составляет около 17 часов, а 4-оптимальной потоковой сети около 325 часов.

8.2. Результаты кустовой оптимизации

С дерева 2-го ранга оптимальности проводилась кустовая оптимизация 4-го ранга оптимальности с границами количества вершин $7 \leq m \leq 40$ (см. Приложение рис. 7). Результаты представлены в табл. 2.

Таблица 2

Оптимизация	кустовая оптимизация 4 ранга ($7 \leq m \leq 40$)
Стоимость сети «до»	227,763
Суммарное количество проходов (всех рангов)	3
Улучшение стоимости	3,411
Время $\approx$ (час:мин:сек, миллисек)	11:02:48,169
Стоимость сети «после»	224,352

8.3. Коррекция кустовой оптимизации 4-го ранга прямой оптимизацией 2-го ранга

Кустовая оптимизация имеет нормированное дальноедействие (количество слоев вершин). Именно вследствие этого проводится полная оптимизация 2-го ранга на всей сети, которая не ограничена по дальноедействию. Результаты приведены в табл. 3 (см. Приложения рис. 8).

Таблица 3

Оптимизация	2 оптимизация
Стоимость сети «до»	224,352
Суммарное количество проходов (всех рангов)	3
Суммарное количество улучшений (всех рангов)	2
Улучшение стоимости	0,129
Время $\approx$ (час:мин:сек, миллисек)	00:01:54,557
Стоимость сети «после»	224,223

Таким образом, оптимизация состоит из 3 этапов:

1. Полная оптимизация 2-го ранга.
2. Кустовая оптимизация 4-го ранга, полученного на 1-ом этапе решения.
3. Полная оптимизация 2-го ранга, полученного на 2-ом этапе решения.

Оптимизация представлена цепочкой действий (см. рис. 4).

Рис. 4. Цепочка действий

Цепочка действий, изображенная на рис. 4, заняла около 11 часов и дала тот же результат, что и полная 3-оптимизация за 17 часов, и полная 4-оптимизация за 325 часов. Финальная сеть данной цепочки изображена на рис.5.

0_plot_of_graph10x10pbpg_treeOptKrazoz_for_tiger4var_and_tiger1opt_16sep2021thur_0_jump_start(tree_price=224.22)

Рис. 5. Сеть 4-го ранга оптимальности

Заключение

Разработан метод и алгоритм построения сети P -го ранга оптимальности на основе оптимизации ее связанных фрагментов $(P+1)$ -го ранга, существенно снижающих размерность задачи и время ее решения на ЭВМ.

Проведен обширный вычислительный эксперимент на большой симметричной сети со 100 вершинами, показавший эффективность метода и алгоритмов.

Разработана программа решения задачи кустовой оптимизации больших сетей.

Конкурирующие интересы. Авторы заявляют, что конфликтов интересов в отношении авторства и публикации нет.

Авторский вклад и ответственность. Все авторы участвовали в написании статьи и полностью несут ответственность за предоставление окончательной версии статьи в печать. Окончательная версия рукописи была одобрена всеми авторами.

Приложение

Рис. 6. Простая оптимизация 4-го ранга

Рис. 7. Кустовая оптимизация 4-го ранга с границами $7 \leq m \leq 40$

Рис. 8. Простая оптимизация 2-го ранга

Список литературы/References

1. Туй Х. Вогнутое программирование при линейных ограничениях // *Докл. АН СССР*, 1964. Т. 159, № 1, С. 32–35. [Tuy Kh. Vognutoe programmirovaniye pri lineynykh ogranicheniyakh // *Dokl. AN SSSR*, 1964. T. 159, № 1, С. 32–35].
2. Булатов В. П., Кассинская Л. И. Некоторые методы минимизации вогнутой функции на выпуклом многограннике // *Методы оптимизации и их приложения. Иркутск: СЭИ СО АН СССР*, 1987, С. 151–172. [Bulatov V. P., Kassinskaya L. I. Nekotorye metody minimizatsii vognutoy funktsii na vypuklom mnogogrannike // *Metody optimizatsii i ikh prilozheniya. Irkutsk: SEI SO AN SSSR*, 1987, С. 151–172 (In Russian)].
3. Анциферов Е. Г., Ащепков Л. Т., Булатов В. П. *Методы оптимизации и их приложения. Ч. 1. Математическое программирование*. Новосибирск: Наука, 1990. 158 с. [Antsiferov E. G., Ashchepkov L. T., Bulatov V. P. *Metody optimizatsii i ikh prilozheniya. Ch. 1. Matematicheskoe programmirovaniye*. Novosibirsk: Nauka, 1990. 158 с. (In Russian)].
4. Меренков А. П., Сеннова Е. В., Сумароков С. В. и др. *Математическое моделирование и оптимизация систем тепло-, водо-, нефте- и газоснабжения*. Новосибирск: Наука, 1992. 407 с. [Merenkov A. P., Sennova E. V., Sumarokov S. V. i dr. *Matematicheskoe modelirovaniye i optimizatsiya sistem teplo-, vodo-, nefte- i gazosnabzheniya*. Novosibirsk: Nauka, 1992. 407 с. (In Russian)].
5. Трубин В. А. *Свойства и методы решения задач оптимального синтеза сетей*. Киев: Об-во «Знание» УССР, 1982. 23 с. [Trubin V. A. *Svoystva i metody resheniya zadach optimal'nogo sinteza setey*. Kiev: Ob-vo «Znanie» USSR, 1982. 23 с.].
6. Трубин В. А., Михалевич В. С., Шор Н. З. *Оптимизационные задачи производственно – транспортного планирования*. Москва: Наука, 1986. 260 с. [Trubin V. A., Mikhalevich V. S., Shor N. Z. *Optimizatsionnye zadachi proizvodstvenno – transportnogo planirovaniya*. Moskva: Nauka, 1986. 260 с. (In Russian)].
7. Кудяев В. Ч. Ранги экстремумов и структурная оптимизация больших сетевых систем // *Известия КБНЦ РАН*, 2016. № 4(72), С. 15–24. [Kudaev V. Ch. Rangi ekstremumov i strukturnaya optimizatsiya bol'shikh setevykh sistem // *Izvestiya KBNTs RAN*, 2016. № 4(72), С. 15–24 (In Russian)].
8. Кудяев В. Ч., Абазоков М. Б. Ранговая оптимизация потоковых сетей // *Вестник КРАУНЦ. Физ.-мат. науки.*, 2018. № 4(24), С. 178–185. [Kudaev V. Ch., Abazokov M. B. Rangovaya optimizatsiya potokovykh setey // *Vestnik KRAUNTs. Fiz.-mat. nauki.*, 2018. № 4(24), С. 178–185 (In Russian)].
9. Кудяев В. Ч., Абазоков М. Б. Компьютерное проектирование потоковых сетей Р-го ранга оптимальности // *Известия КБНЦ РАН*, 2019. № 6(92), С. 122–131. [Kudaev V. Ch., Abazokov M. B. Komp'yuternoye proektirovaniye potokovykh setey R-go ranga optimal'nosti // *Izvestiya KBNTs RAN*, 2019. № 6(92), С. 122–131 (In Russian)].
10. Ватель И. А., Кононенко А. Ф. Об одной численной схеме решения задачи оптимального управления // *ЖВМиМФ*, 1970. № 1. [Vatel' I. A., Kononenko A. F. Ob odnoy chislennoy skheme resheniya zadachi optimal'nogo upravleniya // *ZhVMiMF*, 1970. № 1 (In Russian)].
11. Кристофидес Н. *Теория графов. Алгоритмический подход*. Москва: Мир, 1978. 432 с. [Christofides N. *Graph theory. An algorithmic approach*. London: Academic press, 1975. 432 с. (In Russian)].

Bush optimization method for high ranked stream networks

M. B. Abazokov, V. Ch. Kudaev

Institute of Applied Mathematics and Automation, 360000, Kabardino-Balkariya, Nalchik, Shortanova st., 89 a, Russia.

E-mail: ipma@niipma.ru

The problem of constructing high ranked large-scale stream networks is solved by using bush optimization technique. This technique implies connection of each network vertex with its fragment of a certain dimension. That allows reaching a higher ranked optimality for solving the problem in a given amount of time by a computer.

Keywords: terminal stream network, structure optimization, ranks of extrema, rank optimization method, bush optimization method, computer design, regional and interregional water supply network, dimensionality reduction problem.

DOI: 10.26117/2079-6641-2021-37-4-104-118

Original article submitted: 10.11.2021

Revision submitted: 12.12.2021

For citation. Abazokov M. B., Kudaev V. Ch. Bush optimization method for high ranked stream networks. *Vestnik KRAUNC. Fiz.-mat. nauki.* 2021, **37**: 4, 104-118. DOI: 10.26117/2079-6641-2021-37-4-104-118

Competing interests. The authors declare that there are no conflicts of interest regarding authorship and publication.

Contribution and Responsibility. All authors contributed to this article. Authors are solely responsible for providing the final version of the article in print. The final version of the manuscript was approved by all authors.

The content is published under the terms of the Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/deed.ru>)

© Abazokov M. B., Kudaev V. Ch., 2021

Funding. The study was carried out without financial support from foundations.