Моделирование и анализ данных
2021. Том 11. № 2. С. 51–58
DOI: https://doi.org/10.17759/mda.2021110203
ISSN: 2219-3758 (печатный)
ISSN: 2311-9454 (online)

Modelling and Data Analysis
2021. Vol. 11, no. 2, pp. 51–58
DOI: https://doi.org/10.17759/mda.2021110203
ISSN: 2219-3758 (print)
ISSN: 2311-9454 (online)

◇◇◇◇◇◇◇МЕТОДЫ ОПТИМИЗАЦИИ◇◇◇◇◇◇◇

УДК 531.38;531.39

Модификация алгоритма декомпозиции путей ориентированного графа для учета расписания

Золотарев И.А.*
Московский авиационный институт (НИУ МАИ),
г. Москва, Российская Федерация
ORCID: https://orcid.org/0000-0002-6437-2212
e-mail: yngvar.antonsson@gmail.com

Рассказова В.А.**
Московский авиационный институт (НИУ МАИ),
г. Москва, Российская Федерация,
ORCID: https://orcid.org/0000-0003-4943-3133
e-mail: varvara.rasskazova@mail.ru

Работа направлена на уточнение алгоритма декомпозиции путей ориентированного графа. Рассмотрено дополнительное ограничение на параметр balance для учета расписания отправления локомотивов. Также приведено новое правило расчета параметров Ns. Приведен пример работы алгоритма декомпозиции путей ориентированного графа с учетом расписания. Научная и практическая новизна работы заключается в существенном снижении размерности исходной задачи, что особенно важно в условиях транспортных сетей сложной топологии.

Ключевые слова: теория оптимизации, оптимизация на графах, алгоритм декомпозиции путей ориентированного графа, сильно связный граф, теория расписаний.

Для цитаты:
Золотарев И.А., Рассказова В.А. Модификация алгоритма декомпозиции путей ориентированного графа для учета расписания // Моделирование и анализ данных. 2021. Том 11. № 2. С. 51–58. DOI: https://doi.org/10.17759/mda.2021110203

****Золотарев Игорь Антонович,*** студент магистратуры, Московский авиационный институт, (НИУ МАИ), г. Москва, Российская Федерация, ORCID: https://orcid.org/0000-0002-6437-2212, e-mail: yngvar.antonsson@gmail.com

*****Рассказова Варвара Андреевна,*** кандидат физико-математических наук, доцент кафедры 804 «Теория вероятностей и компьютерное моделирование», Московский авиационный институт, (НИУ МАИ), г. Москва, Российская Федерация, ORCID: https://orcid.org/0000-0003-4943-3133, e-mail: varvara.rasskazova@mail.ru

1. ВВЕДЕНИЕ

Задача составления оптимального расписания грузовых железнодорожных перевозок – сложная оптимизационная задача с большим количеством параметров и большой размерностью массива вершин (станций) и ребер (перегонов). Алгоритм декомпозиции путей ориентированного графа, изложенный в [1] и уточненный в [2], не учитывает наличие расписания отправления локомотивов, а в за время прохождения каждого из перегонов берется число 1. В данной работе будет рассмотрено дополнительное ограничение на параметр balance, играющий важную роль в исходном алгоритме, а также на расчет времени прохождения сегментов маршрута .

2. ПОСТАНОВКА ИСХОДНОЙ ЗАДАЧИ

Рассмотрим ориентированный граф $\vec{G} = (V, E)$, в котором $V = \{v_i : i = [1, n]\}$ – множество вершин и $E = \{(v_i, v_j) : i \neq j, i, j \in [1, n]\}$ – множество дуг ориентированного графа. Пусть задан набор порожденных сильно связных подграфов $\vec{G}_s = (V_s, E_s)$, $s = [1, K]$, таких что выполняются соотношения $\bigcup_{s=1}^{K} V_s = V$ и $\bigcup_{s=1}^{K} E_s = E$.

Пусть задано некоторое множество путей в ориентированном графе $\vec{G} = (V, E)$: $\mathbb{P} = \{p_i, i = [1, m]\}$.

Задача о декомпозиции заданного множества путей на заданном множестве $\vec{G}_s, s = [1, K]$ сильно связных подграфов состоит в том, чтобы разбить каждый путь из множества $\mathbb{P}$ на подпути меньшей длины

$$p_i = \left(p_{i_1}, p_{i_2}, ..., p_{i_{n(i)}}\right), i = [1, m]$$

таким образом, чтобы каждый подпуть $p_{i_j}, j = [1, n(i)]$, целиком лежал в некотором порожденном сильно связном подграфе $\vec{G}_s, s \in [1, K]$.

3. МОДИФИКАЦИЯ ЗАДАЧИ

Модифицируем постановку задачи из [1] для учета расписания (времени прохождения точек маршрута из множества путей $\mathbb{P}$). Если перенести это на физическую модель составления расписаний, то набор порожденных сильно связных подграфов $\vec{G}_s = (V_s, E_s)$, $s = [1, K]$ графа $\vec{G}$ будет представлять собой, например, «города» или участки сложных маршрутов, тогда в разных «городах» могут иметься разные дороги между двумя пунктами, то есть:

Пусть есть два подграфа $\vec{G}_i = (V_i, E_i)$ и $\vec{G}_j = (V_j, E_j)$ и две вершины $v_m \in V_i, V_j$, $v_n \in V_i, V_j$, такие, что, $(v_m, v_n) \in E_i, E_j$. В зависимости от декомпозиции пути, содержащего ребро (v_m, v_n), этот путь будет пролегать либо в подграфе $\vec{G}_i$ или $\vec{G}_j$. Вес ребра (v_m, v_n) на самом деле не обязан совпадать в подграфах $\vec{G}_i$ и $\vec{G}_j$, то есть $\exists w_i(v_m, v_n) \neq w_j(v_m, v_n)$. Физически это может быть обусловлено различным временем стоянки на маршруте, различной длиной дорог или другими внешними факторами.

Пусть имеется таблица $T_{start}\left(\tilde{p}, v_m, v_n\right)$, которое определяет время отправления из точки v_m в точку v_n для варианта прохождения $\tilde{p}$ пути p. Тогда задача о декомпозиции заданного множества путей на заданном множестве $\vec{G}_s$, $s=\left[1, K\right]$ сильно связных подграфов с учетом расписания $T_{start}\left(\tilde{p}, v_m, v_n\right)$ состоит в том, чтобы разбить каждый путь из множества $\mathbb{P}$ на подпути меньшей длины

$$p_i=\left(p_{i_1}, p_{i_2}, \ldots, p_{i_{n(i)}}\right), i=\left[1, m\right]$$

таким образом, чтобы каждый подпуть p_{i_j}, $j=\left[1, n\left(i\right)\right]$ целиком лежал в некотором порожденном сильно связном подграфе $\vec{G}_s$, $s \in\left[1, K\right]$, то есть, найти такую декомпозицию D, для которой выполняются ограничения

$$c\left(D\right) \rightarrow \min_j c\left(D\right)$$

$$balance\left(D\right) \rightarrow \max_j balance\left(D\right)$$

$$\Delta t_{max}\left(D\right)-\Delta t_{min}\left(D\right) \rightarrow \min_j\left(\Delta t_{max}\left(D_j\right)-\Delta t_{min}\left(D_j\right)\right)$$

где $\left\{D_j, j=1,2, \ldots\right\}$ – все возможные декомпозиции заданного множества путей ориентированного графа.

4. МОДИФИКАЦИЯ АЛГОРИТМА ДЕКОМПОЗИЦИИ

Пусть имеется набор времени отправления $T_{start}\left(\tilde{p}, v_m, v_n\right)$. Для упрощения задачи будем считать, что время задано в одном и том же часовом поясе, например в формате Unix Timestamp и задан мультипликатор времени (например, один час или 3600 секунд). Введем поправку на время начала работы системы, для этого найдем $T_0=\min\limits_{i=1..K} T_{start}$ и пересчитаем $T_{start}\left(\tilde{p}, v_m, v_n\right):=\left(T_{start}\left(\tilde{p}, v_m, v_n\right)-T_0\right)/3600$.

Рассмотрим процедуру расчета баланса bal_s. В терминах путей значение “1” в столбце bal_s в исходном алгоритме для некоторого $s \in\left[1, K\right]$ означает, что в пути, заданном рассматриваемой строкой, существует подпуть, взаимно обратный с некоторым подпутем в пути, заданном зафиксированной строкой. Причем эти взаимно обратные подпути при фиксированных разбиениях целиком лежат в подграфе с номером s. Также, при выборе пути для включения в декомпозицию, предпочтение отдается путям, в которых максимальное число взаимно обратных подпутей, другими словами, тех, в которых состав может быть отправлен по обратному маршруту.

При учете расписания, дополнительным критерием для расчета баланса будет учитываться присутствие в расписании обратного рейса в пределах оптимального времени ожидания T_{ε}, то есть такого, что

$$T_{start}\left(p, v_n, v_m\right)-T_{start}\left(\tilde{p}, v_m, v_n\right) \leq T_{\varepsilon}$$

Таким образом, если в таблице $M\left(\mathbb{P}\right)$ после выбора строки в декомпозицию, nom – строка таблицы, выбранная на предыдущем шаге, в столбце mk есть больше одного элемента со значением «1», то рассчитываем значение в столбцах bal_s для строк $nom_i \neq nom$ следующим образом:

Если для пары строк выполняются условия:

- $mk(nom_i) = 1$,
- $Ind(nom) \neq Ind(nom_i)$,
- $v_{k+1}(nom) = v_{k+1}(nom_i), s_k(nom) = s_k(nom_i)$, $\left|T_{start}\left(\tilde{p}(nom), v_{k+1}(nom), v_k(nom)\right) - T_{start}\left(\tilde{p}(nom_i), v_k(nom_i), v_{k+1}(nom_i)\right)\right| \leq T_\varepsilon,\ k = [b_1, b_2]: 1 \leq b_1 \leq b_2 \leq l$, $bal_s(nom) = 0, s: \exists s_k(nom_i) = s$,

то

$$bal_s(nom_i) = 1 \forall s : \exists s_k(nom_i) = s,\ k \in [b_1, b_2],\ s \in [1, K]$$

То есть, функция расчета баланса декомпозиции принимает следующий вид:

$$balance(D) = \sum_{s=1}^{K} \left|\left\{\left(p_{i_j}, p_{i_j}^*\right):\ n(i_j) = s, \quad \Delta T_{start}\left(p_{i_j}, p_{i_j}^*\right) < T_\varepsilon, i \in [1, m], j \in [1, n(i)]\right\}\right|$$

При таком порядке расчета баланса меняется также правило добавления пути в таблицу $M(\mathbb{P})$. Допустим, имеется путь $p = (v_1, v_2, v_3)$, где $(v_1, v_2), (v_2, v_3) \in \vec{G}_1, (v_2, v_3) \in \vec{G}_2$. При построении таблицы для декомпозиции в оригинальном алгоритме этот путь включался в таблицу единственным образом, как $(v_1, 1, v_2, 1, v_3)$, теперь, с учетом расписания, необходимо включить в таблицу оба варианта прохождения $(v_1, 1, v_2, 1, v_3), (v_1, 1, v_2, 2, v_3)$, так как может оказаться, что переназначение маршрута (v_2, v_3) в другой подграф может оказаться выгоднее с точки зрения затрат времени на прохождение участка маршрута.

Определим также новый порядок расчета значений в столбцах N_s, положим, что это время прохождения сегмента, тогда каждый участок (v_1, v_2) сегмента s будет проходиться за время, равное разнице между отправлением из начальной и конечной точек. Так как из последней точки маршрута нет никаких отправлений, то доопределим таблицу: $T_{start}(\tilde{p}, v_i) = T_{finish}(\tilde{p}) \forall v_i = v_{finish}(\tilde{p})$.

$$N_s(nom) = \sum_{i=1}^{l-1} \delta_s(s_i)\left(T_{start}\left(\tilde{p}(nom), v_{i+1}, v_{i+2}\right) - T_{start}\left(\tilde{p}(nom), v_i, v_{i+1}\right)\right)$$

где

$$\delta_s(s_i) = \begin{cases} 1, \text{если } s_i = s; \\ 0, \text{если } s_i \neq s. \end{cases}$$

5. ПРИМЕР РАБОТЫ АЛГОРИТМА ДЕКОМПОЗИЦИИ

Пример 1. Модифицируем пример из [1]. Пусть заданы ориентированный граф $\vec{G} = (V, E)$ и набор подграфов $\vec{G}_s, s \in [1,3]$ (на рис. 1), а также набор путей

$$\mathbb{P} = \left\{p_1 = (v_1, v_2, v_4, v_6, v_3),\ p_2 = (v_1, v_4, v_5),\ p_3 = (v_3, v_6, v_4, v_2, v_1),\ p_4 = (v_5, v_4, v_2, v_1)\right\}$$

и таблица с расписанием T_{start} (табл. 1).

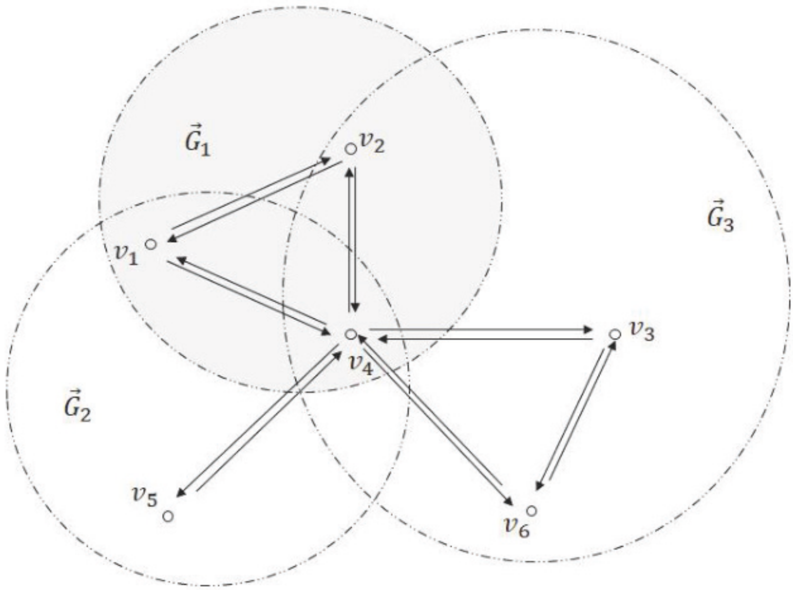

Рис. 1. Ориентированный граф

Таблица 1

Расписание отправлений

nom	*p*	v_1	$T_{start}(v_1, v_2)$	v_2	$T_{start}(v_2, v_3)$	v_3	$T_{start}(v_3, v_4)$	v_4	$T_{start}(v_4, v_5)$	v_5	T_{finish}
1	1	1	00:00	2	01:00	4	02:00	6	03:00	3	04:00
2	1	1	00:00	2	02:00	4	15:00	6	16:00	3	17:00
3	2	1	08:00	4	11:00	5	-	-	-	-	12:00
4	2	1	09:00	4	10:00	5	-	-	-	-	11:00
5	3	3	05:00	6	07:00	4	08:00	2	10:00	1	11:00
6	3	3	05:00	6	07:00	4	09:00	2	11:00	1	12:00
7	4	5	01:00	4	05:00	2	21:00	1	-	-	22:00
8	4	5	01:00	4	03:00	2	08:00	1	-	-	09:00

Построим таблицу $M(\mathbb{P})$ и пересчитаем N_s по правилу (2) (табл. 2) с учетом алгоритмов из [1, 2]. Опустим столбцы $\vec{G}_s$, т.к. они не участвуют в декомпозиции непосредственно.

Положим T_ε равным 10 часам и выполним алгоритм декомпозиции с учетом (1). На первом проходе алгоритма будет выбран путь с номером 3, так как он проходит через минимальное число подграфов ($c(\text{nom} = 3) = 1,\ c(\text{nom} \neq 3) = 2$).

Таблица 2

Таблица $M(\mathbb{P})$ с пересчитанными значениями N_s

nom	p	mk	Ind	v_1	s_1	v_2	s_2	v_3	s_3	v_4	s_4	v_5	c	bal_1	bal_2	bal_3	bal_Σ	N_1	N_2	N_3
1	1	1	0	1	1	2	1	4	3	6	3	3	2	0	0	0	0	2	0	2
2	1	1	0	1	1	2	3	4	3	6	3	3	2	0	0	0	0	2	0	15
3	2	1	0	1	2	4	2	5	0	0	0	0	1	0	0	0	0	0	4	0
4	2	1	0	1	1	4	2	5	0	0	0	0	2	0	0	0	0	1	1	0
5	3	1	1	1	1	2	1	4	3	6	3	3	2	0	0	0	0	3	0	3
6	3	1	1	1	1	2	3	4	3	6	3	3	2	0	0	0	0	2	0	5
7	4	1	1	1	1	2	1	4	2	5	0	0	2	0	0	0	0	20	1	0
8	4	1	1	1	1	2	3	4	2	5	0	0	2	0	0	0	0	2	1	5

На втором проходе будет выбран путь с номером 8, который имеет взаимно обратный подпуть с путем номер 3 с отправкой обратного поезда в течение 10 часов ($c(nom=7)=c(nom=8)$, $bal_\Sigma(nom=7)<bal_\Sigma(nom=8)$).

Таблица 3

Результат работы алгоритма

nom	p	mk	Ind	v_1	s_1	v_2	s_2	v_3	s_3	v_4	s_4	v_5	c	bal_1	bal_2	bal_3	bal_Σ	N_1	N_2	N_3
3	2	2	0	1	2	4	2	5	0	0	0	0	1	0	0	0	0	0	4	0
8	4	2	1	1	1	2	3	4	2	5	0	0	2	0	1	0	1	2	1	5
1	1	2	0	1	1	2	1	4	3	6	3	3	2	1	0	0	1	2	0	2
5	3	2	1	1	1	2	1	4	3	6	3	3	2	1	0	1	2	3	0	3
2	1	0	0	1	1	2	3	4	3	6	3	3	2	1	0	0	1	2	0	15
4	2	0	0	1	1	4	2	5	0	0	0	0	2	0	0	0	0	1	1	0
6	3	0	1	1	1	2	3	4	3	6	3	3	2	1	0	1	2	2	0	5
7	4	0	1	1	1	2	1	4	2	5	0	0	2	0	0	0	0	20	1	0

Следующим будет выбран путь с номером 1, т.к. при включении его в декомпозицию, она будет иметь наименьшее время ($c(nom=1)=c(nom=2)$, $bal_\Sigma(nom=1)=bal_\Sigma(nom=2)$, $N_\Sigma(nom=1)<N_\Sigma(nom=2)$). На последнем проходе алгоритма будет выбран путь с номером 5 ($N_\Sigma(nom=5)<N_\Sigma(nom=6)$). Результат выполнения алгоритма представлен в табл. 2.

6. ЗАКЛЮЧЕНИЕ

Алгоритм декомпозиции путей ориентированного графа расширен для учета расписания. Введено дополнительное ограничение на параметр balance, введены правила для пересчета параметров . С помощью модифицированного алгоритма декомпозиции путей ориентированного графа посчитан модельный пример.

Литература

1. *Гайнанов Д.Н., Кибзун А.И., Рассказова В.А.* Задача о декомпозиции множества путей ориентированного графа и ее приложение // Автоматика и телемеханика. 2018. № 12. С. 142–166.
2. *Золотарев И.А., Рассказова В.А.* Практическая реализация алгоритма декомпозиции путей ориентированного графа // Моделирование и анализ данных. 2020. Том 10. № 3. С. 60–68.

Modification of Algorithm of Directed Graph Paths Decomposition with Schedule

Igor A. Zolotarev*
Moscow Aviation Institute, Moscow, Russia,
ORCID: https://orcid.org/0000-0002-6437-2212
e-mail: yngvar.antonsson@gmail.com

Varvara A. Rasskazova*
Moscow Aviation Institute, Moscow, Russia,
ORCID: https://orcid.org/0000-0003-4943-3133
e-mail: varvara.rasskazova@mail.ru

This study aims to refinement of the oriented graph paths decomposition algorithm. An additional constraint on the balance parameter is considered to take into account the locomotive departure schedule. Also new rule to compute Ns parameters is given. An example of the operation of the oriented graph paths decomposition algorithm with the schedule is given. The scientific and practical novelty of the work lies in a significant reduction in the dimension of the original problem, which is especially important in the conditions of transport networks of complex topology.

Keywords: optimization theory, graph optimization, algorithm of directed graph paths decomposition, strongly connected graph, scheduling theory.

For citation:
Zolotarev I.A., Rasskazova V.A. Modification of Algorithm of Directed Graph Paths Decomposition with Schedule. *Modelirovanie i analiz dannykh = Modelling and Data Analysis*, 2021. Vol. 11, no. 2, pp. 51–58. DOI: https://doi.org/10.17759/mda.2021110203 (In Russ., abstr. in Engl.).

References

1. Gainanov D.N., Kibzun A.I., Rasskazova V.A. The Decomposition Problem for the Set of Paths in a Directed Graph and Its Application. *Automation and Remote Control*, 2018, vol. 79, no. 12, pp. 2217–2236
2. Zolotarev I.A., Rasskazova V.A. Practical Realization of Algorithm of Oriented Graph Paths Decomposition. *Modelling and Data Analysis*, 2020. Vol. 10, no. 3, pp. 60–68.

***Igor A. Zolotarev,** Master Student, Moscow Aviation Institute, Moscow, Russia, ORCID: https://orcid.org/0000-0002-6437-2212, e-mail: yngvar.antonsson@gmail.com

****Varvara A. Rasskazova,** PhD in Physics and Mathematics, Associate Professor, Moscow Aviation Institute, Moscow, Russia, ORCID: https://orcid.org/0000-0003-4943-3133, e-mail: varvara.rasskazova@mail.ru