ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М. В. ЛОМОНОСОВА»

МЕХАНИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
специалиста

ОПТИМИЗАЦИЯ ТРАНСПОРТНОГО ПОТОКА ПРИ ЗАДАННЫХ ПУНКТАХ ОТПРАВЛЕНИЯ И НАЗНАЧЕНИЯ ВСЕХ УЧАСТНИКОВ ДВИЖЕНИЯ

Выполнил студент 610 группы
Пехтерев Станислав Игоревич

____
подпись студента

Научный руководитель:
доктор физико-математических наук
Васенин Валерий Александрович

____
подпись научного руководителя

Москва
2022

Содержание

Тема

Введение

1 Постановка задачи

Для начала поставим общую задачу оптимизации транспортного потока.

1.1 Общая постановка задачи

Пусть задан ориентированный граф $G = (V, E)$. Предположим, что имеется n участников с заданными точками отправления $A_i \in V$ и прибытия $B_i \in V$. Пусть множество P_i есть множество всех простых путей из A_i в B_i. Элемент декартового произведения $P = \prod\limits_{i=1}^{n} P_i$ назовем *комбинацией путей.* Пусть известно, что при комбинации путей участников $\mathbf{p} = (p_1, \ldots, p_n) \in P$ i-ый участник затрачивает $T_i(\mathbf{p}) \in \mathbb{R}_{\geq 0}$ времени на свой путь. Функции T_i назовем *функциями временных затрат* участника i. *Некооперативным прокладыванием пути* в ориентированном графе G назовем пятерку $F = (n, G, \{A_i\}_{i=1}^n, \{B_i\}_{i=1}^n, \{T_i\}_{i=1}^n)$. Некооперативное прокладывание пути предполагает, что каждый участник стремится сократить собственные временные затраты выбором пути p_i, несмотря на временные затраты других участников. Для того, чтобы скооперировать участников, введем некоторую функцию $\Phi(\mathbf{p}) = \phi(T_1(\mathbf{p}), \ldots, T_n(\mathbf{p}))$, определенную на множестве всех возможных комбинаций путей P и отображающую его во множество действительных чисел. С помощью нее участники могут отслеживать, как влияет изменение их пути на общую картину движения. Такую функцию назовем *функцией стоимости.*

Для заданных некооперативного прокладывания пути F и функции стоимости Φ необходимо найти комбинацию путей $\mathbf{p}^*$ такую, что функция стоимости на ней минимальна, то есть

$$\Phi(\mathbf{p}^*) = \min_{\mathbf{p} \in P} \Phi(\mathbf{p}). \tag{1}$$

Комбинацию путей $\mathbf{p}^*$ будем называть *оптимальной*, а стоимость $\Phi(\mathbf{p}^*)$ — *оптимальной стоимостью.*

Далее будем считать, что каждый участник имеет одинаковый приоритет в вопросе изменения своих временных затрат, то есть

$$\frac{\partial \phi}{\partial T_i} \equiv 1, \ i = 1, \ldots, n,$$

или

$$\phi(T_1, \ldots, T_n) = \sum_{i=1}^{n} T_i.$$

Приведем ряд ограничений на функции $T_i(\mathbf{p})$, которые позволят задать движения всех участников во времени при комбинации путей $\mathbf{p}$.

1.2 Постановка задачи в терминах модели движения

Будем считать, что временные затраты участника на выбранном пути состоят из временных затрат на каждом ребре этого пути:

$$T_i(\mathbf{p}) = \sum_{e \in p_i} \overline{\tau}_{e,i}(\mathbf{p}),$$

где функции $\overline{\tau}_{e,i}(\mathbf{p})$ — временные затраты i-ого участника на ребре e при комбинации путей $\mathbf{p}$.

Для того, чтобы задать движение участника на пути, введем функции присутствия участника на ребре в момент времени t:

$$\theta_{e,i}(\mathbf{p}, t) = \begin{cases} 1, & \text{если } i\text{-ый участник движется по ребру } e \text{ в момент времени } t, \\ 0, & \text{иначе,} \end{cases}$$

где $\sum\limits_{e \in E} \theta_{e,i}(\mathbf{p}, t)$ принимает значение 1, пока i -ый участник не посетит свою точку назначения B_i. Пусть достижение конца пути p_i наступает в момент $T_i(\mathbf{p})$, после чего $\sum\limits_{e \in E} \theta_{e,i}(\mathbf{p}, t)$ принимает значение 0. Получаем, что

$$T_i(\mathbf{p}) = \sum_{e \in p_i} \int\limits_0^{T_i(\mathbf{p})+\Delta t} \theta_{e,i}(\mathbf{p}, t) dt, \ \forall \Delta t > 0. \tag{2}$$

Будем считать, что движение каждого участника является непрерывным и однонаправленным в графе G. Другими словами, участник не может резко повляться и исчезать на несмежных ребрах, а также находиться на уже пройденных ребрах. Таким образом, функции $\theta_{e,i}(\mathbf{p}, t)$ являются индикаторами некоторых временных отрезков $[t_{e,i}^{in}(\mathbf{p}), t_{e,i}^{out}(\mathbf{p})]$, которые описывают однонаправленное движение:

$$\begin{cases} t_{e,i}^{in}(\mathbf{p}), t_{e,i}^{out}(\mathbf{p}) \in \mathbb{R}_+, & i = 1, \ldots, n, \ e \in E, \\ t_{e,i}^{in}(\mathbf{p}) \le t_{e,i}^{out}(\mathbf{p}), & i = 1, \ldots, n, \ e \in p_i, \\ t_{e,i}^{in}(\mathbf{p}) = t_{e,i}^{out}(\mathbf{p}) = 0, & i = 1, \ldots, n, \ e \notin p_i, \\ t_{e_1,i}^{in}(\mathbf{p}) = t_{e_2,i}^{out}(\mathbf{p}), & i = 1, \ldots, n, \ e_1, e_2 \in p_i, \exists A, B, C \in V : e_1 = (A, B), e_2 = (B, C) \\ t_{e,i}^{in}(\mathbf{p}) = 0, & i = 1, \ldots, n, \ e = (A_i, X), X \in V. \end{cases} \tag{3}$$

Заметим, что выбор таких отрезков пока неоднозначен. Далее считаем, что для каждого ребра e, участника i и комбинации путей $\mathbf{p}$ каким-то образом выбраны некоторые величины $t_{e,i}^{in}(\mathbf{p}), t_{e,i}^{out}(\mathbf{p})$, удовлетворяющие ограничениям (3). Тогда функция временных затрат (2) i-ого участника примет вид

$$T_i(\mathbf{p}) = \sum_{e \in E} t_{e,i}^{out}(\mathbf{p}) - t_{e,i}^{in}(\mathbf{p}). \tag{4}$$

Функция стоимости в этом случае равна

$$\Phi(\mathbf{p}) = \sum_{i=1}^{n} \sum_{e \in E} t_{e,i}^{out}(\mathbf{p}) - t_{e,i}^{in}(\mathbf{p}). \tag{5}$$

Считаем, что временные затраты участника i на ребре e ограничены некоторыми положительными константами $\overline{\tau}_{e,i}^{min}, \overline{\tau}_{e,i}^{max}$:

$$0 < \overline{\tau}_{e,i}^{min} \leq t_{e,i}^{out}(\mathbf{p}) - t_{e,i}^{in}(\mathbf{p}) \leq \overline{\tau}_{e,i}^{max}, e \in p_i,\ i = 1, \ldots, n. \tag{6}$$

Заметим, что задача оптимизации целевой функции (5) с ограничениями (3), (6) ставится в терминах задачи смешанного целочисленного линейного программирования с булевыми переменными $I_{e,i}$ и вещественными переменными $t_{e,i}^{in}, t_{e,i}^{out}$. Для участника i первые отвечают факту проезда по ребру e, вторые — моментам прохождения этого ребра. Однако в данных ограничениях решение уже есть — участник i передвигается по кратчайшему пути в графе G с весами $\overline{\tau}_{e,i}^{min}$. Тривиальность решения связана с тем, что в данной задаче оптимизации отсутствуют влияния участников друг на друга. Для того, чтобы учесть это влияние, для каждого участника i введем микроскопическую характеристику движения $v_i(\mathbf{p}, t)$ — положительную, ограниченную функцию, описывающую скорость участника. Также будем считать, что для каждого ребра $e \in E$ определена его длина $l_e > 0$.

Тогда имеет место следующее ограничение

$$\int\limits_0^{T_i(\mathbf{p})} \theta_{e,i}(\mathbf{p}, t) v_i(\mathbf{p}, t) dt = l_e, e \in p_i, i = 1, \ldots, n, \tag{7}$$

или

$$\int\limits_{t_{e,i}^{in}(\mathbf{p})}^{t_{e,i}^{out}(\mathbf{p})} v_i(\mathbf{p}, t) dt = l_e, e \in p_i, i = 1, \ldots, n. \tag{8}$$

Будем говорить, что уравнения (8) задают *движения участников*, а функции $v_i(\mathbf{p}, t)$ назовем *моделью движения*. Без ограничения общности считаем, что $\overline{\tau}_{e,i}^{min}, \overline{\tau}_{e,i}^{max}$ вычисляются в самом быстром и самом медленном вариантах передвижения по ребру e участником i, а именно

$$\overline{\tau}_{e,i}^{min} = \frac{l_e}{\max\limits_{\mathbf{p} \in P, t \in \mathbb{R}} (v_i(\mathbf{p}, t))},\ \overline{\tau}_{e,i}^{max} = \frac{l_e}{\min\limits_{\mathbf{p} \in P, t \in \mathbb{R}} (v_i(\mathbf{p}, t))}. \tag{9}$$

Заметим, что величины $t_{e,i}^{in}(\mathbf{p}), t_{e,i}^{out}(\mathbf{p}) \in \mathbb{R}_+$ — произвольные вещественные величины, которые удовлетворяют ограничениям (3), (6), (8), (9).

Утверждение 1.1. *Пусть задан ориентированный граф G с положительными длинами $\{l_e\}_{e \in E}$, модель движения $v_i(\boldsymbol{p}, t)$, и для каждого ребра e, участника i и комбинации путей $\boldsymbol{p}$ задано множество величин $t_{e,i}^{in}(\boldsymbol{p}), t_{e,i}^{out}(\boldsymbol{p}) \in \mathbb{R}_+$, для которых выполняются ограничения* (3), (6), (8), (9). *Тогда $t_{e,i}^{out}(\boldsymbol{p})$ и $t_{e,i}^{in}(\boldsymbol{p})$, $e \in p_i$ есть функции от комбинации путей $\boldsymbol{p} \in P$.*

Доказательство. Зафиксируем некоторую комбинацию путей $\mathbf{p}$. Опишем алгоритм поиска значений $t_{e,i}^{out}(\mathbf{p})$ и $t_{e,i}^{in}(\mathbf{p})$ и покажем его корректность.

Algorithm 1 Моделирование движения участников

Input: количество участников n, ориентированный граф G, комбинация путей $\mathbf{p}$ графа G

Output: $t^{out}_{e,i}(\mathbf{p})$, $t^{in}_{e,i}(\mathbf{p})$, $e \in p_i, i = 1, \dots, n$

Data: текущее время t, текущее ребро e_i и часть пройденного ребра x_i участника i

1: $t = 0$
2: **for** $i = 1, \dots, n$ **do**
3: $\quad e_i \leftarrow$ первое ребро пути p_i
4: $\quad x_i \leftarrow 0$
5: $\quad t^{in}_{e_i,i}(\mathbf{p}) \leftarrow 0$
6: **end for**
7: **while** $\exists i : i$ — не доехал **do**
8: $\quad \tau^* \leftarrow \operatorname{argmin}\{\tau \in \mathbb{R} : \tau > t, \int\limits_t^\tau v_i(\mathbf{p}, t)dt = (1 - x_i)l_{e_i}, i$— не доехал$\}_{i=1}^n$
9: $\quad$ **for** $i = 1, \dots, n$ **and** $i -$ не доехал **do**
10: $\qquad x_i \leftarrow x_i + \frac{1}{l_{e_i}} \int\limits_t^{\tau^*} v_i(\mathbf{p}, t)dt$
11: $\qquad$ **if** $x_i = 1$ **and** e_i - не последнее ребро пути p_i **then**
12: $\qquad\quad x_i \leftarrow 0$
13: $\qquad\quad t^{out}_{e_i,i}(\mathbf{p}) \leftarrow \tau^*$
14: $\qquad\quad e_i \leftarrow$ следующее ребро за e_i в пути p_i
15: $\qquad\quad t^{in}_{e_i,i}(\mathbf{p}) \leftarrow \tau^*$
16: $\qquad$ **end if**
17: $\quad$ **end for**
18: $\quad t \leftarrow \tau^*$
19: **end while**

Описанный алгоритм называется *моделированием движения.*

Корректность. Для доказательства корректности алгоритма достаточно доказать корректность шага 8 и достижимость шага 11. Это следует из того, что функция скорости ограничена снизу (см. ограничения (6), (9)). Алгоритм сойдется, поскольку пути p_i конечны. □

Используя это утверждение, можем ввести следующее понятие:

Некооперативным передвижением по графу G с положительными длинами $\{l_e\}_{e \in E}$ в модели движения $v_i(\mathbf{p}, t)$ назовем такое некооперативное прокладывание пути $F = \left(n, G, \{A_i\}_{i=1}^n, \{B_i\}_{i=1}^n, \left\{\sum\limits_{e \in p_i} t^{out}_{e,i}(\mathbf{p}) - t^{in}_{e,i}(\mathbf{p})\right\}_{i=1}^n\right)$, где функции $t^{in}_{e,i}(\mathbf{p}), t^{out}_{e,i}(\mathbf{p})$ получены путем моделирования движения с моделью движения $v_i(\mathbf{p}, t)$ и длинами ребер $\{l_e\}_{e \in E}$ графа G. Значит, постановка задачи в терминах модели движения следующая:

Пусть задано некооперативное передвижение по графу G с положительными длинами $\{l_e\}_{e \in E}$ в модели движения $v_i(\mathbf{p}, t)$. Требуется найти комбинацию путей $\mathbf{p}$ такую, что функция

$$\Phi(\mathbf{p}) = \sum_{i=1}^{n} \sum_{e \in E} t_{e,i}^{out}(\mathbf{p}) - t_{e,i}^{in}(\mathbf{p}) \tag{10}$$

минимальна.

Оказывается, что для любого некооперативного прокладывания пути при любых положительных длинах $\{l_e\}_{e \in E}$ существует эквивалентное ему некооперативное передвижение в графе с этими длинами в некоторой модели движения $v_i(\mathbf{p}, t)$. Другими словами, любое некооперативное прокладывание пути можно промоделировать.

2 Модели движения

Утверждение 2.1. *Пусть заданы некоторое некооперативное прокладывание пути F и положительные длины ребер $\{l_e\}_{e \in E}$ графа G. Тогда можно задать такую модель движения $v_i(\boldsymbol{p}, t)$, что затраченное время на передвижение i-ым участником при комбинации путей $\boldsymbol{p}$ совпадает с его временными затратами, то есть выполняется* (4).

Доказательство. Рассмотрим модель движения с постоянными скоростями

$$v_i(\mathbf{p}, t) = \overline{v}_i(\mathbf{p}) = \frac{T_i(\mathbf{p})}{\sum\limits_{e \in p_i} l_e}.$$

Промоделировав движение с такими скоростями получим (4). □

Таким образом можно сказать, что без ограничения общности считать, что каждый выбор комбинации путей $\mathbf{p}$ можно промоделировать.

Очевидно, что решение задачи перебором не является практичным — оно сводится к перебору всех комбинаций путей $\mathbf{p} \in P$. Так, например, количество таких комбинаций в полном графе есть $2^{n(|V|-1)}$, перебрать которые в условиях реальных данных вычислительно сложно. Однако, в случае, когда условие (8) можно описать в терминах задачи удовлетворения ограничений, задача оптимизации (10) может быть описана в терминах смешанного целочисленного линейного программирование и, как следствие, может быть решена стандартным решателем. Оказывается можно выделить целый класс таких моделей движения, для которых это возможно.

2.1 Макроскопические модели

Предположим, что скорость участника зависит от некоторой общей для участников величины. Например, от функции загруженности ребра

$$n_e(\mathbf{p}, t) = \sum_{i=1}^{n} \theta_{e,i}(\mathbf{p}, t),$$

значение которой в момент времени t соответствует количеству участников на ребре e в этот момент при комбинации путей $\mathbf{p}$. Предположим скорость участника зависит только от загруженности ребра, на котором он находится:

$$v_i(\mathbf{p}, t) = \sum_{e \in E} \theta_{e,i}(\mathbf{p}, t) v(n_e(\mathbf{p}, t)), i = 1, \dots, n \tag{11}$$

Такую модель движения в дальнейшем будем называть *макроскопической*. Например, естествено расмотреть модель $v(n_e(\mathbf{p}, t)) = \frac{v_{max}}{n_e(\mathbf{p}, t)}$. В общем случае такая модель задается последовательностью значений $\{v(k)\}_{k=1}^{n}$.

Лемма 2.1. *Пусть даны вещественные переменные a, b целочисленного программирования и известно, что существует константа $M > 0 : |a| < M$, $|b| < M$. Тогда можно добавить новую целочисленную переменную $\mathbf{1}(\{a < b\}) \in \{0, 1\}$ такую, что*

$$\boldsymbol{1}(\{a < b\}) = \begin{cases} 1, \; a < b, \\ 0, \; a \geq b. \end{cases}$$

Доказательство. Добавим в нашу задачу два неравенства:

$$2M(\mathbf{1}(\{a < b\}) - 1) < b - a \leq 2M\mathbf{1}(\{a < b\})$$

Очевидная проверка показывает, что неравенство выполняется для любых a, b.

□

Утверждение 2.2. *Пусть модель движения* $v_i(\boldsymbol{p}, t)$ *макроскопическая. Тогда задача* (10) *есть задача смешанного целочисленного линейного программирования.*

Доказательство. Докажем для случая $n = 2$. Для случаев $n \geq 2$ доказательство аналогичное.

Пусть имеется задача смешанного целочисленного линейного программирования (3) с переменными $t_{e,i}^{in}, t_{e,i}^{out}, I_{e,i}, e \in E, i = 1, 2$. Преобразуем условие (8) к каноническому виду задачи удовлетворения ограничений. Для удобства обозначим обоих участников индексами $i, j \in \{1, 2\}$.

$$\int\limits_0^\infty \theta_{e,i}(\mathbf{p}, t) v_i(\mathbf{p}, t) dt = \int\limits_0^\infty \theta_{e,i}(\mathbf{p}, t) \sum_{e^1 \in E} \theta_{e^1,i}(\mathbf{p}, t) v(n_{e^1}(\mathbf{p}, t)) dt =$$

$$\int\limits_0^\infty \theta_{e,i}(\mathbf{p}, t) v(n_e(\mathbf{p}, t)) dt = \int\limits_{n_e(\mathbf{p},t)=1} \theta_{e,i}(\mathbf{p}, t) v(n_e(\mathbf{p}, t)) dt + \int\limits_{n_e(\mathbf{p},t)=2} \theta_{e,i}(\mathbf{p}, t) v(n_e(\mathbf{p}, t)) dt =$$

$$\int\limits_{\theta_{e,j}(\mathbf{p},t)=0} \theta_{e,i}(\mathbf{p}, t) v(1) dt + \int\limits_{\theta_{e,j}(\mathbf{p},t)=1} \theta_{e,i}(\mathbf{p}, t) v(2) dt =$$

$$\int\limits_0^\infty \theta_{e,i}(\mathbf{p}, t) v(1) dt - \int\limits_{\theta_{e,j}(\mathbf{p},t)=1} \theta_{e,i}(\mathbf{p}, t) v(1) dt + \int\limits_{\theta_{e,j}(\mathbf{p},t)=1} \theta_{e,i}(\mathbf{p}, t) v(2) dt =$$

$$v(1) \int\limits_0^\infty \theta_{e,i}(\mathbf{p}, t) dt + (v(2) - v(1)) \int\limits_0^\infty \theta_{e,i}(\mathbf{p}, t) \theta_{e,j}(\mathbf{p}, t) dt =$$

$$v(1) \overline{\tau}_{e,i}(\mathbf{p}) + (v(2) - v(1)) \int\limits_0^\infty \theta_{e,i}(\mathbf{p}, t) \theta_{e,j}(\mathbf{p}, t) dt = l_e, e \in p_i$$

Неизвестный интеграл - время совместного проезда участников на ребре e.

В переменных задачи смешанного целочисленного программирования получим:

$$v(1)(t_{e,i}^{out} - t_{e,i}^{in}) + (v(2) - v(1))(t_{e,ij}^{out} - t_{e,ij}^{in}) = l_e I_{e,i},$$

где новые перемнные $t_{e,ij}^{in}$, $t_{e,ij}^{out}$ отвечают за начало и конец совместного проезда участников. Другими словами $[t_{e,ij}^{in}, t_{e,ij}^{out}] = [t_{e,i}^{in}, t_{e,i}^{out}] \cap [t_{e,j}^{in}, t_{e,j}^{out}]$. Просуммировав по всем ребрам $e \in E$, получим

$$v(1) \sum_{e\in E} (t_{e,i}^{out} - t_{e,i}^{in}) = \sum_{e\in E} l_e I_{e,i} - (v(2) - v(1)) \sum_{e\in E} (t_{e,ij}^{out} - t_{e,ij}^{in}).$$

Заметим, что левая часть есть временные затраты участника i с коэффициентом $v(1)$, поэтому задачу оптимизации можно переписать в виде

$$\frac{1}{v(1)} \sum_{i=1}^{n} \sum_{e\in E} l_e I_{e,i} + \frac{v(1) - v(2)}{v(1)} \sum_{i=1}^{n} \sum_{e\in E} (t_{e,ij}^{out} - t_{e,ij}^{in}) \to \min .$$

Для завершения доказательства необходимо показать, что переменные $t_{e,ij}^{in}, t_{e,ij}^{out}$ описываются линейными ограничениями. Обозначим $\Delta t = t_{e,ij}^{out} - t_{e,ij}^{in}, \Delta t_1 = t_{e,i}^{out} - t_{e,i}^{in}, \Delta t_2 = t_{e,j}^{out} - t_{e,j}^{in}, \Delta t_3 = t_{e,i}^{out} - t_{e,j}^{in}, \Delta t_4 = t_{e,j}^{out} - t_{e,i}^{in}$

Используя лемму 2.1, при $M = \max\limits_{e\in E, k=i,j} \overline{\tau}_{e,k}^{max}$, добавим в задачу новые переменные $\mathbf{1}(\{\Delta t_k > \Delta t_l\}), k \neq l, k, l \in 1, 2, 3, 4$. Рассмотрим величину $T_{max} = |E|M$. Добавим в случае $v(1) \geq v(2)$ нашу задачу следующие неравенства:

$$\Delta t \geq 0,$$

$$\Delta t \geq \Delta t_k - T_{max} \sum_{l\neq k} \mathbf{1}(\{\Delta t_k > \Delta t_l\}), k = 1, 2, 3, 4.$$

В случае $v(1) < v(2)$ добавим те же ограничения с другим знаком неравенства. Тогда с учетом оптимизации переменная Δt есть длина отрезка $[t_{e,ij}^{in}, t_{e,ij}^{out}]$.

$\square$

Следствие 2.1. *Пусть модель движения $v_i(\boldsymbol{p}, t) = \sum\limits_{e\in E} \theta_{e,i}(\boldsymbol{p}, t) v(n_e(\boldsymbol{p}, t))$ макроскопическая и последовательность $v(n) > 0, \forall n \in \mathbb{Z}_+$ убывает. Предположим, что оптимальное время движения в модели с постоянными скоростями $v(1)$ есть $\widetilde{T}$. Тогда имеет место*

$$\widetilde{T} \leq T \leq \frac{v(1)}{v(n)} \widetilde{T}.$$

Доказательство. Докажем каждое неравенство в отдельности

1. В модели, где все участники едут с постоянными скоростям движение происходит по кратчайшим путям. Тогда временные затраты есть $\widetilde{T} = \frac{1}{v(1)} \sum\limits_{i=1}^{n} \sum\limits_{e\in p_i} l_e$, где p_i - кратчайшие пути. На тех же путях задается самый худший случай макроскопической модели - все едут с минимальной скоростью, то есть $T = \frac{1}{v(n)} \sum\limits_{i=1}^{n} \sum\limits_{e\in p_i} l_e$. Тогда получим

$$T \leq \frac{1}{v(n)} \sum_{i=1}^{n} \sum_{e\in p_i} l_e = \frac{v(1)}{v(n)} \widetilde{T}.$$

2. Проделывая аналогичне выкладки что и в доказательстве 2.2, можно получить, что функция оптимизация есть

$$\frac{1}{v(1)}\sum_{i=1}^{n}\sum_{e\in E} l_e I_{e,i} + \sum_{k=2}^{n}\frac{v(1)-v(k)}{v(1)}\sum_{i=1}^{n}\sum_{e\in E}\sum_{\substack{s_k\in 2^n\\ |s_k|=k}}\Delta t_{e,s_k} \to \min,$$

где переменные $\Delta t_{e,s_k}$ отвечают за время совместного движения участников (и только их) s_k по ребру e.

Тогда получим

$$T \geq \min\left(\frac{1}{v(1)}\sum_{i=1}^{n}\sum_{e\in E} l_e I_{e,i}\right) + \min\left(\sum_{k=2}^{n}\frac{v(1)-v(k)}{v(1)}\sum_{i=1}^{n}\sum_{e\in E}\sum_{\substack{s_k\in 2^n\\ |s_k|=k}}\Delta t_{e,s_k}\right) \geq \widetilde{T}.$$

□

Таким образом, мы получили класс моделей движения, для которых задача оптимизации транспортного потока может быть поставлена в терминах смешанного целочисленного линейного программирования. Однако такой класс моделей движения плохо описывает реальное движение участников. Так, например, модель не учитывает расстояние между участниками и их порядок на ребре.

2.2 Микроскопические модели

Микроскопическими называются модели движения, которые не являются макроскопическими, то есть не представимы в виде (11). В таких моделях явно исследуется движение каждого автомобиля. Выбор такой модели позволяет теоретически достичь более точного описания движения автомобилей по сравнению с макроскопической моделью, однако этот подход требует больших вычислительных ресурсов при практических применениях.

Для простоты рассмотрим однополосное бесконечное движение. Пусть $x_i(t) \in [0, +\infty)$ — координаты на полосе участника i. Предположим, что скорость участника ограничена некоторой общей величиной v_{max}. Пусть в момент времени $t = 0$ выполняется $x_1(0) \leq x_2(0) \leq \cdots \leq x_n(0)$.

Модель пропорциональной скорости

Рассмотрим пример, когда скорость машины пропорциональна расстоянию до следующей машины. Положим $d_i(t) = x_{i+1}(t) - x_i(t), i = 1, \ldots, n-1$. Без ограничения общности считаем, что $d_i(0) < D$, где D - характерное расстояние взаимодействия участников. Иначе рассмотрим подпоследовательности участников, для которых выполняется это условие.

Пусть модель движения есть

$$v_i(t) = \begin{cases} v_{max}, & i = n, \\ v_{max}\frac{d_i(t)}{D}, & i \neq n. \end{cases} \tag{12}$$

Для поиска функций $x_i(t)$ достаточно рассмотреть систему дифференциальных уравнений

$$\dot{d}_i(t) = v_{i+1}(t) - v_i(t).$$

Решением такой системы является

$$d_{n-k}(\tau) = \sum_{l=0}^{k-1} \left(\frac{d_{n-k+l}(0) - D}{l!} \tau^l e^{-\tau} \right) + D,$$

где $\tau = \frac{v_{max}}{D} t$. Модель обладает тем свойством, что порядок участников постоянен и участники не покидают зону взаимодействия D.

Данная модель хорошо описывает реальное движение участников, однако ее практическое применение вызывает сложности, поскольку решение уравнения время, вычисляемое на шаге 2 процесса моделирования движения может быть найдено только приближенно.

Модель снижения скорости

Предположим, что существует некоторая величина c_n, которая отвечает за последовательное снижение скорости участников относительно их порядка:

$$v_{n-k} = v_{max} - c_n k, \quad k = 0, \ldots, n-1.$$

величину c_n выберем из соображений, что $v_0 = \frac{v_{max}}{n}$. Тогда $c_n = \frac{v_{max}}{n}$. Если смоделировать данное движение на графе, то функция скоростей будут кусочно постоянными. Это связано с тем, при смене ребра некоторым участником меняется порядок и величина $n_e(\mathbf{p}, t)$. Поэтому она не лучшим образом описывает реальное движение, однако проста в использовании.

Пока для микроскопических моделей нет очевидного подхода к решению. Однако для любой модели движения можно описать алгоритмы оптимизации, которые сходится к «локальному минимуму». Рассмотрим такие алгоритмы в следующем разделе.

3 Равновесие транспортных потоков

В этом разделе мы исследуем задачу поиска равновесия транспортных потоков как возможность поиска оптимального транспортного потока.

3.1 Некооперативное и кооперативное равновесие

Некооперативной игрой в нормальной форме назовем тройку $\Gamma = (n, \{S_i\}_{i=1}^n, \{H_i\}_{i=1}^n)$, где $n \in \mathbb{N}$ - количество участников игры, S_i - множество стратегий участника $i \in 1, \dots, n$, H_i - функция выйгрыша участника i, определенная на множестве ситуаций $S = \prod\limits_{i=1}^{n} S_i$ и отображающая его в множество действительных чисел.

Равновесием Нэша некооперативной игры в нормальной форме $\Gamma = (n, \{S_i\}_{i=1}^n, \{H_i\}_{i=1}^n)$ назовем такую стратегию $\mathbf{s}^* \in S$, если изменение своей стратегии с $\mathbf{s}_i^*$ на любую $s \in S_i$ не выгодно ни одному игроку i, то есть

$$H_i(\mathbf{s}^*) \geq H_i(\big(\mathbf{s}_1^*, \dots, \mathbf{s}_{i-1}^*, s, \mathbf{s}_{i+1}^*, \dots, \mathbf{s}_n^*\big)), \forall s \in S_i,\ i = 1, \dots, n.$$

Заметим, что в общем случае ничего нельзя сказать о существовании и единственности равновесия некооперативной игры.

Введем понятие некооперативного и кооперативного равновесия, которое является классическим определением равновесия в терминах некооперативного прокладывания пути, где выйгрыш заключается в сэкономленном времени передвижения и стоимости соответственно.

Некооперативным равновесием некооперативного прокладывания пути по графу F назовем комбинацию путей $\widehat{\mathbf{p}} \in P$, которая является равновесием Нэша некооперативной игры $\widehat{\Gamma} = (n, \{P_i\}_{i=1}^n, \{-T_i\}_{i=1}^n)$. Множество всех некооперативных равновесий обозначим $\widehat{P}$.

Кооперативным равновесием некооперативного прокладывания пути по графу F и функции стоимости $\Phi(\mathbf{p})$ назовем комбинацию путей $\widetilde{\mathbf{p}} \in P$, которая является равновесием Нэша некооперативной игры $\widetilde{\Gamma} = (n, \{P_i\}_{i=1}^n, \{-\Phi\}_{i=1}^n)$. Множество всех кооперативных равновесий обозначим $\widetilde{P}$.

Заметим, что определения некооперативного прокладывания пути и некооперативной игры эквивалентны. Таким образом, любой пример игры, где равновесия Нэша не существует, можно использовать как пример некооперативного передвижения по графу, где нет кооперативного равновесия. Однако для кооперативного равновесия верно обратное:

Утверждение 3.1. *Множество кооперативных равновесий $\widetilde{P}$ не пусто, причем оптимальная комбинация путей является таким равновесием, то есть $\boldsymbol{p}^* \in \widetilde{P}$.*

Доказательство. Очевидно, что для поскольку для любого $\mathbf{p} \in P$

$$\Phi(\mathbf{p}^*) \leq \Phi(\mathbf{p}),$$

поэтому это верно и для комбинаций путей $\mathbf{p} = \left(\mathbf{p}_1^*, \dots, \mathbf{p}_{i-1}^*, p, \mathbf{p}_{i+1}^*, \dots, \mathbf{p}_n^*\right)$, $p \in P_i$, $i = 1, \dots, n$. ☐

В некотором смысле кооперативное равновесие можно назвать «локальным минимум» функции Φ.

3.2 Поиск кооперативного равновесия

Рассмотрим ряд алгоритмов, позволяющих получить некоторое кооперативное равновесие.

Общим свойством всех этих алгоритмов является предположение о том, что существует некоторый алгоритм $\alpha(\{\Phi_i\}_{i=1}^n)$, позволяющий решить задачу оптимизации некоторой функции стоимости $\Phi_i(\mathbf{p}) = \phi_i(T_1(\mathbf{p}), \ldots, T_n(\mathbf{p}))$ путем выбора пути p_i. В работе Л. Е. Разумовой [1] представлен один из таких алгоритмов, позволяющий получить оптимальный путь p_i за полиномиальное относительно входных данных время, при условии, что функция $\Phi_i(\mathbf{p})$ удовлетворяет неравенству

$$\Phi_i(\mathbf{p}_1, \ldots, \mathbf{p}_{i-1}, pe, \mathbf{p}_{i+1}, \ldots, \mathbf{p}_n) \leq \Phi_i(\mathbf{p}_1, \ldots, \mathbf{p}_{i-1}, qe, \mathbf{p}_{i+1}, \ldots, \mathbf{p}_n), \tag{13}$$

где p, q — два пути к некоторой вершине $B \in V$, ребро e выходит из этой вершины и путь p оптимальнее чем q относительно стоимости Φ_i:

$$\Phi_i(\mathbf{p}_1, \ldots, \mathbf{p}_{i-1}, p, \mathbf{p}_{i+1}, \ldots, \mathbf{p}_n) \leq \Phi_i(\mathbf{p}_1, \ldots, \mathbf{p}_{i-1}, q, \mathbf{p}_{i+1}, \ldots, \mathbf{p}_n). \tag{14}$$

Таким образом, имея некоторые функции стоимости $\Phi_i(\mathbf{p})$, удовлетворяющие условиям (13), (14) можно описать полиномиальный алгоритм β, позволяющий перейти к меньшей стоимости передвижения путем изменения некоторого пути p_i участника i. Для поиска кооперативного рановесия достаточно найти неподвижную точку алгоритма β.

Algorithm 2 Поиск неподвижной точки алгоритма β

Input: Начальная комбинация путей $\mathbf{p}_0 \in P$, алгоритм β, количество итераций $iter$

Output: кооперативное равновесие $\widetilde{\mathbf{p}} \in \widetilde{P}$

Data: $\mathbf{p}_{cur}$ - текущая комбинация путей, $\mathbf{p}_{new}$ - новая комбинация путей, i - номер итерации

$\mathbf{p}_{cur} \leftarrow \mathbf{p}_0$
$i \leftarrow 0$
while $i < iter$ **do**
    $\mathbf{p}_{new} \leftarrow \beta(\mathbf{p}_{cur})$
    $i \leftarrow i + 1$
    if $\mathbf{p}_{new} = \mathbf{p}_{cur}$ **then**
        return $\mathbf{p}_{cur}$
    end if
end while
return $\mathbf{p}_{cur}$

Данный алгоритм не дает гарантий на сходимость за число итераций, не зависящее от количества комбинаций путей. Однако на каждой итерации от позволяет получить более оптимальную комбинацию путей $\mathbf{p}$. Опишем алгоритм, который, с некоторыми допущениями, не зависит от количество комбинаций путей $|P|$ и находит оптимальную комбинацию путей $\mathbf{p}$.

Аналогично можно описать алгоритм, не зависящий от начальной комбинации путей $\mathbf{p}_0 \in P$. Для этого предположим, что функции $\Phi_i(\mathbf{p})$ даны для любого количества участников, то есть имеется набор функций $\{\{\Phi_{i,k}\}_{i=1}^{k}\}_{k=1}^{n}$, для каждого k отображающие декартово прозведение $\prod\limits_{i=1}^{k} P_i$ в множество действительных чисел $\mathbb{R}$ каждая из которых удовлетворяет условиям (13), (14). Опишем алгоритм, который добавляет каждого участника последовательно.

Algorithm 3 Последовательное добавление участников в движение

Input: алгоритм α, алгоритмы $\{\beta_k\}_{k=1}^{n}$

Output: кооперативное равновесие $\widetilde{\mathbf{p}} \in \widetilde{P}$

Data: $\mathbf{p}_k \in \prod\limits_{i=1}^{k} P_i$ - кооперативное отношение для первых k участников, $\mathbf{p}_{new}$ - новая комбинация путей, k - номер итерации

$k \leftarrow 0$
while $k <= n$ **do**
    $\mathbf{p}_{k+1} \leftarrow (\mathbf{p}_k, \alpha(\mathbf{p}_k))$
    Запустим алгоритм 2 на комбинации путей $\mathbf{p}_{k+1}$
    $k \leftarrow k + 1$
end while

При гипотизе, что добавление участника-эгоиста, который оптимизирует свои стоимости в оптимальную комбинацию путей не меняет свойства оптимальности, можно сказать, что алгоритм сходится к оптимальному пути. Для того, чтобы алгоритм сошелся за n применений алгоритма α, достаточно изменить условие оптимальности на условие кооперативного равновесия.

4 Результаты

5 Заключение

Список литературы

[1] *Л. Е. Разумова, С. А. Афонин*, “Построение оптимального маршрута при заданной модели движения других участников движения”.

[2] *А. А. Глуцюк*, “О двумерных полиномиально интегрируемых бильярдах на поверхностях постоянной кривизны”, Доклады Академии наук, **481**, 6, 2018, 594–598

[3] *M. Bialy, A. E. Mironov*, “Angular billiard and algebraic Birkhoff conjecture”, Adv. Math., **313**, 2017, 102–126