

Санкт-Петербургский государственный университет

Прикладная математика и информатика

Исследование операций и принятие решений в задачах оптимизации,
управления и экономики

Яковенко Дарья Максимовна

Построение плана ремонта городских дорог

Бакалаврская работа

Научный руководитель:

д. ф.-м. н., профессор Романовский И.В.

Рецензент:

к. ф.-м. н., доцент Григорьева Н.С.

Санкт-Петербург

2016

SAINT-PETERSBURG STATE UNIVERSITY

Applied Mathematics and Computer Science

Operation Research and Decision Making in Optimisation, Control and Economics
Problems

Iakovenko Daria

Construction of a plan of city road reconstructions

Bachelor's Thesis

Scientific supervisor:
professor Joseph Romanovsky

Reviewer:
docent Natalia Grigorieva

Saint-Petersburg
2016

Оглавление

1.	Введение	4
2.	Транспортная задача и некоторые способы нахождения ее решения	4
3.	Применение метода Брэгмана к транспортной задаче в сетевой постановке	7
4.	Модификация метода Брэгмана	11
5.	Моделирование ремонта дорог на примере г. Кронштадт	12
5.1.	Ремонт Цитадельской дороги	14
5.2.	Ремонт ул. Мартынова	16
5.3.	Ремонт Макаровской ул.	18
5.4.	Ремонт Красной ул.	20
5.5.	Ремонт Советской ул.	22
6.	Выводы и рекомендации по построению плана ремонта дорог города Кронштадт	24
7.	Приложение	29

1. Введение

В декабре 2015 года Комитет по развитию транспортной инфраструктуры Санкт-Петербурга утвердил план ремонта городских дорог на 2016 год. По мнению, высказанному в газете [5], сроком начала работ будет лето, а самый разгар работ придется на осень, во время завершения поры отпусков. Из-за этого ситуация на дорогах осложнится, могут возникнуть пробки. Целью этой работы является составление плана ремонта городских дорог, такого, который позволит предотвратить, насколько это возможно, затруднения движения, возникающие на дорогах города при проведении ремонта.

Для составления плана ремонта городских дорог необходимо знать или уметь моделировать дорожную ситуацию, а также уметь моделировать изменение дорожной ситуации в случае, если та или иная дорога будет перекрыта на ремонт. В данной работе моделирование производится при помощи метода Брэгмана, применяемого к транспортной задаче в сетевой постановке.

Мы будем рассматривать два способа закрыть дорогу на ремонт: первый — дорога перекрывается полностью, движение по ней останавливается; второй — дорога перекрывается частично (например, две из четырех полос четырехполосной дороги), движение по ней продолжается, но ограничивается. Для того, чтобы моделировать второй случай, была придумана модификация известного в транспортном моделировании метода Брэгмана, которая учитывает ограничения на пропускную способность дороги.

В этой работе описывается метод сравнения вариантов расписания в ограниченной части города, который описывается с использованием дорожной сети города Кронштадт. Данные взяты из Интернета.

2. Транспортная задача и некоторые способы нахождения ее решения

Транспортная задача — это задача о перевозке грузов из пунктов производства в пункты потребления с минимальными затратами на перевозку. Задача рассматривается в матричной и в сетевой постановке.

При матричной постановке транспортной задачи даны положительные числа N и M — количество пунктов производства и потребления соответственно, а

также положительные векторы $A = (a_i)_I$, $B = (b_j)_J$, где $I = 1 : N$, $J = 1 : M$.

Каждый элемент a_i вектора A соответствует количеству груза, отправляемого из i -го пункта производства, а элемент b_j вектора B соответствует количеству груза, который требуется j -му пункту потребления. Часто накладывается дополнительное требование того, что сумма всего производимого груза равна сумме всего потребляемого груза: $\sum_{i \in I} a_i = \sum_{j \in J} b_j$. Если транспортная задача удовлетворяет этому требованию, ее называют *замкнутой*.

Кроме того задана положительная матрица $C = \|c_{ij}\|_{I,J}$, элемент c_{ij} которой равен цене за перевозку единицы груза из i -го пункта производства в j -й пункт потребления.

Требуется найти матрицу перевозок $X = \|x_{ij}\|_{I,J}$, элемент x_{ij} которой равен количеству груза, перевозимого из i -го пункта производства в j -й пункт потребления. Ясно, что нужно получить такую матрицу X , что

$$\sum_{j \in J} x_{ij} = a_i, \quad \sum_{i \in I} x_{ij} = b_j,$$

то есть сумма грузов, вывозимых из i -го пункта производства равна количеству груза, произведенному в этом пункте, и сумма грузов, ввозимых в j -й пункт потребления равна потребности этого пункта.

Сетевая постановка отличается от матричной тем, что в ней может не быть дороги между некоторыми пунктами потребления и некоторыми пунктами производства. Кроме того, могут присутствовать промежуточные пункты, которые ничего не производят и ничего не потребляют. В сетевой постановке транспортной задачи дан граф $G = \langle V, E \rangle$ с M дугами и N вершинами, дуги которого отображают дороги. Для каждой дуги известна цена перевозки по ней единицы груза. Также задан вектор $b = (b_i)_I$, $I = 1 : N$, который отображает потребности вершин: если $b_i < 0$, то i -я вершина — производитель, который производит $-b_i$ единиц груза, если $b_i > 0$, то i -я вершина — потребитель, которому необходимо b_i единиц груза, если $b_i = 0$, то i -я вершина является промежуточной: не является ни производителем, ни потребителем. Требование того, чтобы сумма производимых грузов была равна сумме потребляемых, для этой задачи выглядит так: $\sum_{i \in I} b_i = 0$.

Требуется найти вектор $x = (x_j)_J$, $J = 1 : M$ потоков на дугах, элемент x_j которого равен количеству груза, перевозимого по j -й дуге. Этот вектор

должен удовлетворять требованию того, чтобы сумма ввозимых и вывозимых из вершины i грузов была равна b_i . То есть $\sum_{k \in J} a_{ik} x_k = b_i$ для любого $i \in I$, где $A = \|a_{ik}\|_{I,J}$ — матрица инцидентий графа G .

Один из методов, которым можно решать транспортную задачу в матричной постановке, был предложен ленинградским архитектором Г.В. Шелейховским. Г.В. Шелейховский рассматривал замкнутую задачу, в которой вместо матрицы цен перевозки грузов по дорогам $C = \|c_{ij}\|_{I,J}$ задана матрица $W = \|w_{ij}\|_{I,J}$ “весов” дорог. Решение, полученное при помощи метода Шелейховского, не минимизирует цену перевозки грузов по дорогам, но обладает другим интересным свойством.

Алгоритм метода Шелейховского выглядит так:

- $X := W$;
- Для каждого $i \in I$ нормируем i -ю строку матрицы X :
находим $S := \sum_{j \in J} x_{ij}/a_i$, назначаем $x_{ij} = x_{ij}/S, j \in J$.
- Для каждого $j \in J$ нормируем j -й столбец матрицы X :
находим $S := \sum_{i \in I} x_{ij}/b_j$, назначаем $x_{ij} = x_{ij}/S, i \in I$.
- Если $S \neq 1$ для какого-то j , возвращаемся к шагу 2.

И. В. Романовский заметил, что решение, полученное при помощи метода Шелейховского, максимизирует взвешенную энтропию, а сходимость этого алгоритма была доказана Львом Мееровичем Брэгманом в 1966 году, тогда аспирантом матмеха ЛГУ.

Теорема 1 *Если мы решаем задачу максимизации взвешенной энтропии с транспортными ограничениями:*

$$\sum_{i,j} x_{ij} \times \ln(w_{ij}/x_{ij}) \rightarrow \max$$

$$\sum_{i \in I} a_i = \sum_{j \in J} b_j$$

и метод Шелейховского сходится, он сходится к оптимальному решению.

Доказательство этой теоремы можно найти в [3].

Чуть позже Брэгманом было представлено обобщение метода Шелейховского для произвольных линейных ограничений.

Теорема 2 (Брэгман, 1966) *Если мы решаем задачу максимизации взвешенной энтропии с линейными ограничениями:*

$$\sum_i x_i \times \ln(w_i/x_i) \rightarrow \max$$

$$\sum_{k \in J} a_{ik} x_k = b_i \quad (1)$$

где новое приближение вычисляется по формуле

$$x_k^* = \exp(\lambda \times a_{ik}) x_k, \quad (2)$$

λ — единственное решение уравнения

$$\sum_{k \in J} a_{ik} \exp(\lambda \times a_{ik}) x_k = b_i, \quad (3)$$

и алгоритм сходится, то он сходится к оптимальному решению.

Подробно об этом написано в [2].

За прошедшее время метод Брэгмана стал всемирно признан. В рамках этой работы рассматривается применение метода Брэгмана к транспортной задаче в сетевой постановке.

3. Применение метода Брэгмана к транспортной задаче в сетевой постановке

Применяя метод Брэгмана к транспортной задаче в сетевой постановке, будем моделировать движение по дорогам города.

Считаем, что задан граф $G = \langle V, E \rangle$ с M дугами и N вершинами, дуги которого отображают дороги. Задан вектор $b = (b_i)_I$ потребностей вершин, где $b_i < 0$, если из i -й вершины выезжает $-b_i$ автомобилей, $b_i > 0$, если в i -ю вершину въезжает b_i автомобилей и $b_i = 0$ для промежуточных вершин. Задан вектор $w = (w_j)_J$ положительных “весов” дуг. Пусть $A = \|a_{ij}\|_{I,J}$ — матрица

инциденций графа G . Требуется найти вектор $x = (x_j)_J$ потоков машин на дугах. Линейные ограничения, которым должны удовлетворять элементы вектора x , выглядят так:

$$\sum_{k \in J} a_{ik} x_k = b_i \quad \text{для любого } i \in I$$

и новое приближение вычисляется по формулам (2), (3).

Рассмотрим подробнее уравнение (3). Поскольку матрица A — это матрица инциденций графа, ее элементы a_{ik} могут принимать только значения 0, 1, и -1 . А значит, уравнение можно записать в виде

$$\sum_{k \in K_1} e^\lambda x_k - \sum_{k \in K_{-1}} e^{-\lambda} x_k = b_i, \quad (4)$$

где K_1 — множество таких k , что $a_{ik} = 1$, K_{-1} — множество таких k , что $a_{ik} = -1$.

Вынесем e^λ и $e^{-\lambda}$ за знаки суммирования:

$$e^\lambda \sum_{k \in K_1} x_k - e^{-\lambda} \sum_{k \in K_{-1}} x_k = b_i.$$

Обозначим $\sum_{k \in K_1} x_k$ за S^+ и $\sum_{k \in K_{-1}} x_k$ за S^- , получим:

$$e^\lambda S^+ - e^{-\lambda} S^- = b_i.$$

Домножив на e^λ , получим

$$e^{2\lambda} S^+ - S^- = e^\lambda b_i.$$

Перенесем $e^\lambda b_i$ в левую часть неравенства и произведем замену переменной $e^\lambda = t$, получим уравнение относительно переменной t

$$t^2 S^+ - t b_i - S^- = 0. \quad (5)$$

Предположим, что

$$S^+ \neq 0, \quad S^- \neq 0.$$

Тогда корнями этого уравнения являются

$$t_1 = \frac{b_i + \sqrt{b_i^2 + 4S^+S^-}}{2S^+}, \quad t_2 = \frac{b_i - \sqrt{b_i^2 + 4S^+S^-}}{2S^+}.$$

Корень t_2 нас не интересует, поскольку e^λ не может принимать неположительные значения. Проведем обратную замену и получим ответ

$$e^\lambda = \frac{b_i + \sqrt{b_i^2 + 4S^+S^-}}{2S^+}.$$

Новое приближение вычисляется по формуле (2). Поскольку a_{ik} могут принимать только значения 1, -1 и 0, элемент x_k^* нового приближения будет равен x_k , полученному на предыдущем шаге, если $a_{ik} = 0$, будет равен $e^\lambda x_k$, если $a_{ik} = 1$ и, наконец, будет равен $e^{-\lambda} x_k$, если $a_{ik} = -1$.

Уравнение решалось в предположении, что $S^+ \neq 0$ и $S^- \neq 0$ В случае

$$S^+ = 0, \quad S^- \neq 0$$

уравнение 5 принимает вид

$$-tb_i - S^- = 0.$$

При $b_i \neq 0$ его решением является

$$t = -\frac{S^-}{b_i}.$$

Проверим, будет ли оно положительным. Условие $S^+ = 0$ означает, что суммарный поток по всем входящим в i -ю вершину дугам равен нулю. Поскольку все потоки положительны, это означает также, что поток по каждой входящей дуге равен нулю. По предположению $S^- \neq 0$. Поскольку потоки на дугах удовлетворяют равенству (1) для каждого $i \in I$, а S^- — это сумма потоков на таких дугах, что $a_{ik} = -1$, то равенство (1) принимает вид $\sum_{k \in K_{-1}} -x_k = b_i$. Поскольку потоки на дугах положительны, $b_i < 0$.

С точки зрения модели условия $S^- \neq 0$ и $S^+ = 0$ означают соответственно, что в i -ю вершину не въезжает никаких машин, но какие-то выезжают. Значит, машины, которые движутся по исходящим дугам, выезжают на дороги непосредственно из вершины i , то есть вершина i в терминах транспортной задачи является производителем.

Полученное решение положительно, можем провести обратную замену.

$$e^\lambda = -\frac{S^-}{b_i}.$$

Осталось разобраться со случаем $S^+ \neq 0$ и $S^- = 0$ (случай $S^+ = 0$ и $S^- = 0$

является вырожденным и соответствует изолированной вершине, его рассматривать не будем). Уравнение (5) принимает вид

$$t^2 S^+ - t b_i = 0.$$

Его корни

$$t_1 = \frac{b_i}{S^+}, \quad t_2 = 0.$$

Корень $t_2 = 0$ нам не подходит. Подходит ли корень t_1 ?

Условие $S^- = 0$ означает, что сумма потоков по исходящим дугам равна нулю, т. е. все потоки по исходящим дугам равны нулю, а условие $S^+ \neq 0$ означает, что сумма потоков по входящим дугам положительна. Поскольку потоки на дугах удовлетворяют равенству (1) для каждого $i \in I$, $S^+ = \sum_{k \in K_1} x_k$, а K_1 — это множество таких индексов, что $a_{ik} = 1$, то равенство (1) принимает вид $\sum_{k \in K_1} x_k = b_i$. Потоки на дугах положительны, значит, $b_i > 0$.

С точки зрения модели из i -й вершины не выезжает никаких машин, но некоторые въезжают. Въезжающие машины не двигаются дальше, а остаются в вершине i . В терминах транспортной задачи она потребитель.

Полученное решение положительно, значит, можем провести обратную замену:

$$e^\lambda = \frac{b_i}{S^+}.$$

Таким образом, решением уравнения (4) является

$$e^\lambda = \begin{cases} \frac{b_i + \sqrt{b_i^2 + 4S^+ \cdot S^-}}{2S^+}, & \text{если } S^+ \neq 0 \text{ и } S^- \neq 0; \\ -\frac{S^-}{b_i}, & \text{если } S^+ = 0 \text{ и } S^- \neq 0; \\ \frac{b_i}{S^+}, & \text{если } S^+ \neq 0 \text{ и } S^- = 0. \end{cases} \quad (6)$$

где $S^+ = \sum_{k \in K_1} x_k$ и $S^- = \sum_{k \in K_{-1}} x_k$, K_1 — множество таких k , что $a_{ik} = 1$, K_{-1} — множество таких k , что $a_{ik} = -1$.

Алгоритм, которым будем решать транспортную задачу в сетевой постановке, можно описать так:

1. $x := w$;
2. Для каждого $i \in I$ находим новое приближение x по формуле (2), где λ вычисляется по (6);

3. Повторять шаг 2, пока значения элементов вектора x не перестанут меняться.

Итак, метод Брэгмана можно успешно применять для моделирования движения на дорогах. Необходимо понять, можно ли его применить для моделирования ситуации ремонта дорог.

Существует два варианта закрытия дороги на ремонт:

- Дорога полностью закрывается на ремонт. В этом случае мы просто выкидываем соответствующие ей дуги из графа.
- Дорога закрывается на ремонт частично (например, на четырехполосной дороге перекрывается движение по двум из четырех полос). В этом случае соответственно уменьшается пропускная способность дороги.

В первом случае по данному графу G можно построить граф G' , в котором отсутствуют дуги, соответствующие закрытым на ремонт дорогам, и решить задачу на полученном графе G' методом Брэгмана. Для второго случая необходимо модифицировать метод Брэгмана так, чтобы потоки на дугах не превышали заданной пропускной способности.

4. Модификация метода Брэгмана

Задан граф $G = \langle V, E \rangle$ с M дугами и N вершинами, $A = \|a_{ij}\|_{I,J}$ — его матрица инциденций, заданы вектор $b = (b_i)_I$ потребностей вершин, вектор $w = (w_j)_J$ положительных “весов” дуг, вектор $r = (r_j)_J$ ограничений на дугах. Требуется найти вектор $x = (x_j)_J$, удовлетворяющий линейным ограничениям (1) и условию $x_j \leq r_j$ для любого $j \in J$.

На первом шаге, как ранее, положим $x := w$. Далее будем пересчитывать значения вектора x по (2), где λ вычисляется по (6), с одним отличием: как только поток на дуге превышает пропускную способность, назначим его в точности равным пропускной способности, а “излишек” распределим по дугам, инцидентным рассматриваемой вершине и идущим в ту же сторону, что и ограниченная дуга.

Более формально это можно записать так. Пусть при рассмотрении вершины i_0 поток на дуге j_0 превысил ее пропускную способность: $x_{j_0} > r_{j_0}$. Найдем

\hat{x}_{j_0} — величину, на которую поток превышает пропускную способность: $\hat{x}_{j_0} = x_{j_0} - r_{j_0}$. Назначим поток на дуге равным пропускной способности: $x_{j_0} := r_{j_0}$. Величину \hat{x}_{j_0} распределим по дугам, инцидентным вершине i_0 , и идущим в том же направлении, что дуга j_0 : если i_0 — начало дуги j_0 , распределим \hat{x}_{j_0} по дугам с началом в i_0 . Если i_0 — конец дуги j_0 , распределим \hat{x}_{j_0} по дугам, концом которых является i_0 (чтобы найти такие дуги, рассмотрим строку i_0 матрицы инцидентностей A и выберем все такие $j \in J$, что $a_{i_0j} = a_{i_0j_0}$).

Распределяя \hat{x}_{j_0} , необходимо помнить, что на других дугах тоже важно не превысить пропускную способность. Пусть есть s дуг j_{l_1}, \dots, j_{l_s} , инцидентных вершине i_0 и направленных в ту же сторону, что и дуга j_0 . Выберем среди них s_1 дуг $j_{l_1}, \dots, j_{l_{s_1}}$ таких, что поток на них строго меньше пропускной способности. Попробуем поровну распределить \hat{x}_{j_0} между ними: найдем величину $\frac{\hat{x}_{j_0}}{s_1}$ и попробуем прибавить ее к потоку на каждой из дуг $j_{l_1}, \dots, j_{l_{s_1}}$. Если полученный поток на каждой из дуг $j_{l_1}, \dots, j_{l_{s_1}}$ не превышает пропускную способность, назначаем поток на этих дугах равным сумме исходного и величины $\frac{\hat{x}_{j_0}}{s_1}$, считаем, что дуга j_0 обработана и переходим к рассмотрению следующей дуги. Если на каких-то дугах полученный поток превышает пропускную способность, выберем среди дуг $j_{l_1}, \dots, j_{l_{s_1}}$ s_2 таких, на которых не превышает. Попробуем распределить \hat{x}_{j_0} между ними: найдем величину $\frac{\hat{x}_{j_0}}{s_2}$ и попробуем прибавить ее к потоку на каждой из дуг $j_{l_1}, \dots, j_{l_{s_2}}$ и так далее, пока на итерации p не найдем подмножество из s_p дуг, прибавление к потоку на которых значения $\frac{\hat{x}_{j_0}}{s_p}$ сделает поток не большим, чем пропускная способность этих дуг. Если найти непустое множество из s_p дуг не удастся, назначим поток на дуге j_0 равным $r_{j_0} + \hat{x}_{j_0}$, то есть большим пропускной способности.

5. Моделирование ремонта дорог на примере г. Кронштадт

Была написана программа на языке C#, реализующая метод Брэгмана и его модификацию. Моделирование дорожной ситуации производится на примере города Кронштадт.

Дороги Кронштадта отображены графом со 113 вершинами и 330 дугами, дуги которого — это дороги, а вершины — перекрестки.

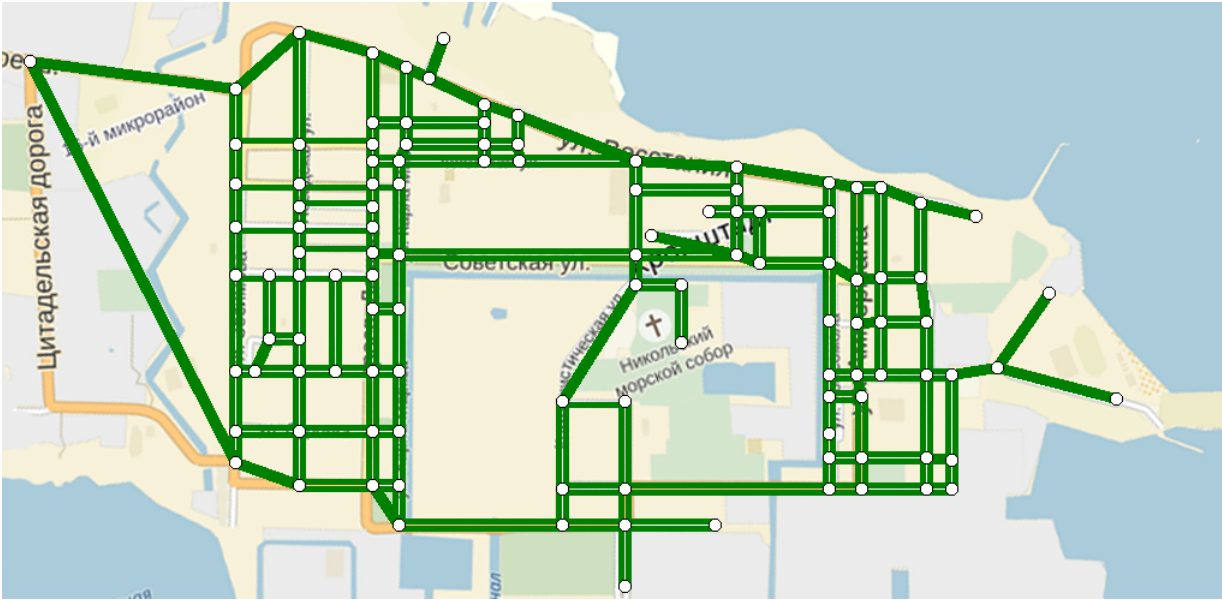


Рис. 1: Граф, отображающий дороги Кронштадта

Вектор потребностей вершин b задавался произвольно, все элементы вектора “весов” дорог w равны между собой, все элементы вектора ограничений r изначально равны очень большому числу M , которого потоки на дугах гарантированно не достигнут.

Понятно, что для более реальных результатов необходимы данные о настоящих дорогах Кронштадта: вектор w должен зависеть от длины, ширины дороги, дорожного покрытия и прочих параметров. Значение w_j должно быть тем больше, чем дорога j в некотором смысле “популярнее”. Вектор b должен отображать реальные перемещения машин внутри города. Вектор r должен зависеть от количества полос на дороге, дорожного покрытия, погодных условий и прочего.

В данной работе рассмотрены два варианта ремонта дороги: первый — дорога полностью перекрывается на ремонт (интересно, что одинаковые результаты получаются при применении метода Брэгмана для графа, из которого удалены дуги, соответствующие дороге, и при применении модифицированного метода Брэгмана, если дугам, соответствующим дороге, назначена пропускная способность 0). Второй вариант — дорога перекрывается на ремонт частично. Для того, чтобы реализовать этот вариант, в данной работе по дороге “пропускается” в два раза меньше машин, чем едет обычно: была запущена программа для исходного графа, посчитан поток на каждой из дуг. В случае, когда дорога перекрывается на ремонт, ее пропускная способность назначается равной

половине от потока в исходном графе.

Для того, чтобы отобразить изменение дорожной ситуации при закрытии на ремонт той или иной дороги, дуги графа в иллюстрациях раскрашиваются в разные цвета в зависимости от изменения потока на них: темно-зеленый цвет соответствует случаю, когда новый поток по дуге меньше либо равен потоку в исходном графе; светло-зеленый — поток увеличился, но не более, чем на 20 %; желтый — поток увеличился более, чем на 20, но не более, чем на 40 %; оранжевый — более 40, но не более 50; и, наконец, красный — поток увеличился более, чем на 50 %.

По данным Комитета по развитию транспортной инфраструктуры Санкт-Петербурга от 18.12.2015 известно, что в 2016 году семь улиц Кронштадтского района Санкт-Петербурга подлежат ремонту:

- Макаровская ул. от Синего моста до Докового моста
- Советская ул. от ул. Рошаля до ул. Комсомола
- Красная ул. от Макаровской ул. до Манежного пер.
- Кронштадтское шоссе от д. 28 до ул. Адмирала Грейга
- ул. Мартынова от ул. Карла Маркса до ул. Зосимова
- ул. Литке от Кронштадтского шоссе до Цитадельского шоссе
- Цитадельская дорога от Кронштадтского шоссе до Цитадельского шоссе

Пять из них, а именно Макаровская ул., Советская ул., Красная ул., ул. Мартынова и Цитадельская дорога рассматриваются в этой работе.

5.1. Ремонт Цитадельской дороги

Рассмотрим Цитадельскую дорогу (на Рис. 2 она отмечена синим цветом). Если полностью перекрыть дорогу на ремонт, дорожная ситуация изменится следующим образом:

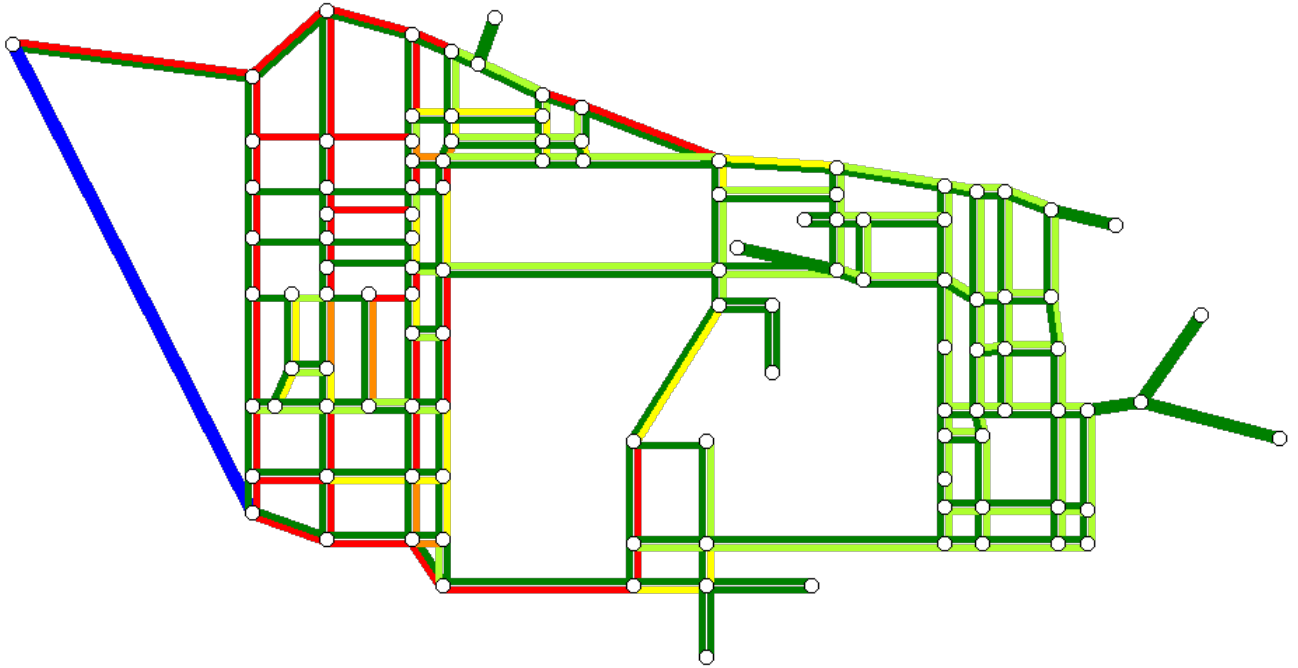


Рис. 2: Цитадельская дорога полностью перекрыта на ремонт

В случае, если дорога перекрыта на ремонт не полностью (перекрытая не полностью Цитадельская дорога отмечена бирюзовым цветом), дорожная ситуация изменится так:

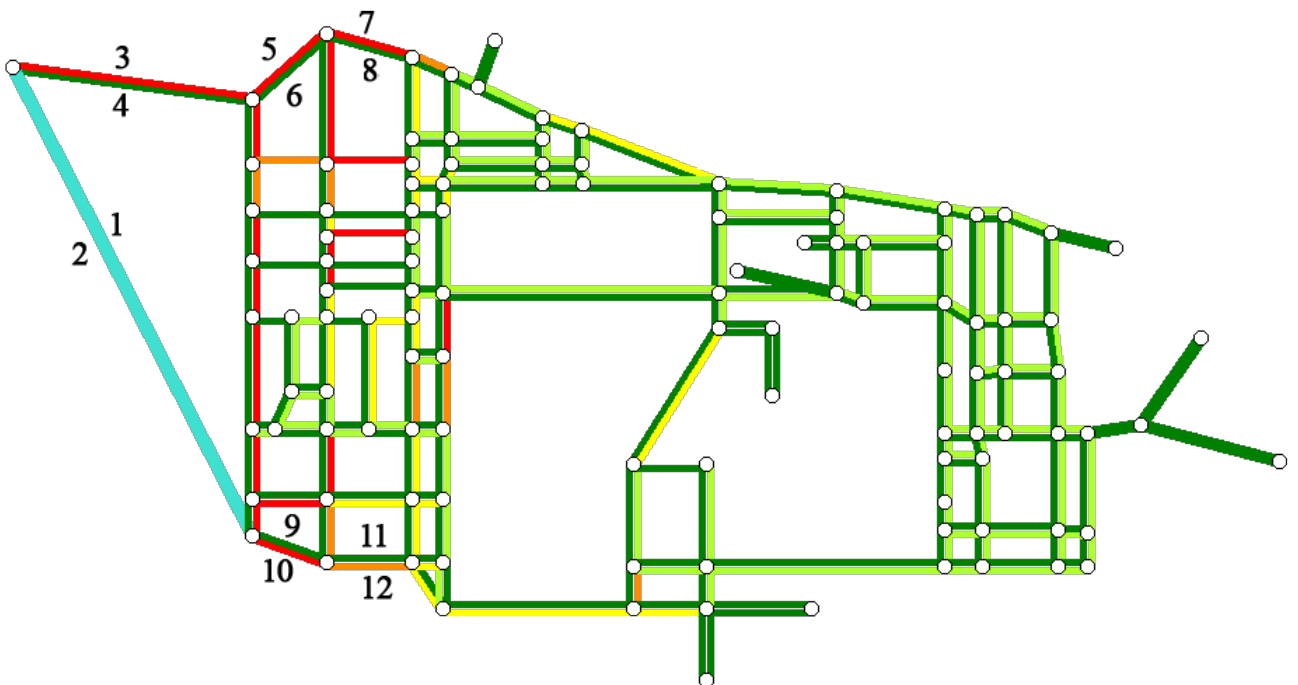


Рис. 3: Движение по Цитадельской дороге ограничено

Числовые данные, соответствующие потокам на некоторых дугах (отмечен-

ных на картинке числами), приведены в таблице ниже:

№	Название	Исходный поток	Поток при перекрытой дороге	Поток при огранич. движении
1	Цитадельская д.	0.083	0	0
2		11.995	0	5.000
3	Кронштадтское ш.	2.340	0.100	0.193
4		0.427	10.099	5.193
5	Ул. Зосимова	0.167	0.087	0.108
6		5.973	11.523	9.301
7	Ул. Восстания	1.013	0.300	0.441
8		0.987	3.328	2.268
9	Ул. Мартынова	0.599	0.147	0.220
10		1.670	6.795	4.539
11		0.186	0.105	0.129
12		5.390	9.507	7.773

Видно, что любой ремонт Цитадельской дороги существенно осложняет дорожную ситуацию во всей западной части города, но случай, когда дорога перекрыта наполовину, осложняет движение в меньшей степени.

5.2. Ремонт ул. Мартынова

Изменение дорожной ситуации при полном закрытии ул. Мартынова (на Рис. 4 отмечена синим) на ремонт выглядит следующим образом:

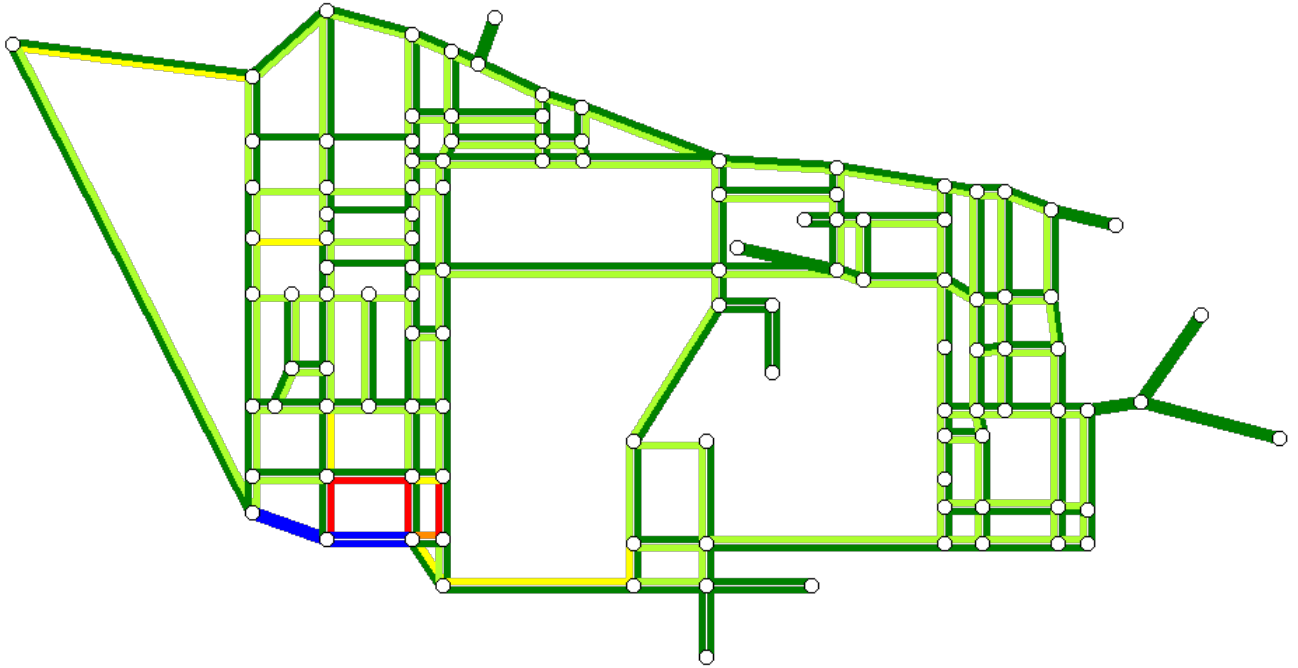


Рис. 4: Ул. Мартынова полностью перекрыта на ремонт

В случае, если ул. Мартынова закрыта на ремонт наполовину (отмечена бирюзовым на Рис. 5), изменение дорожной ситуации получилось таким же.



Рис. 5: Движение по ул. Мартынова ограничено

Ремонт ул. Мартынова существенно затрудняет движение на юго-западе и на отдельных участках западной части города. Учитывая схожесть результа-

тов при полностью закрытой на ремонт дороге и при закрытой наполовину, нет разницы, как перекрывать дорогу. Если ремонт всей дороги одновременно производится быстрее, можно рекомендовать его.

5.3. Ремонт Макаровской ул.

При полном закрытии на ремонт Макаровской ул. (отмечена синим на Рис. 6) получается следующая ситуация:

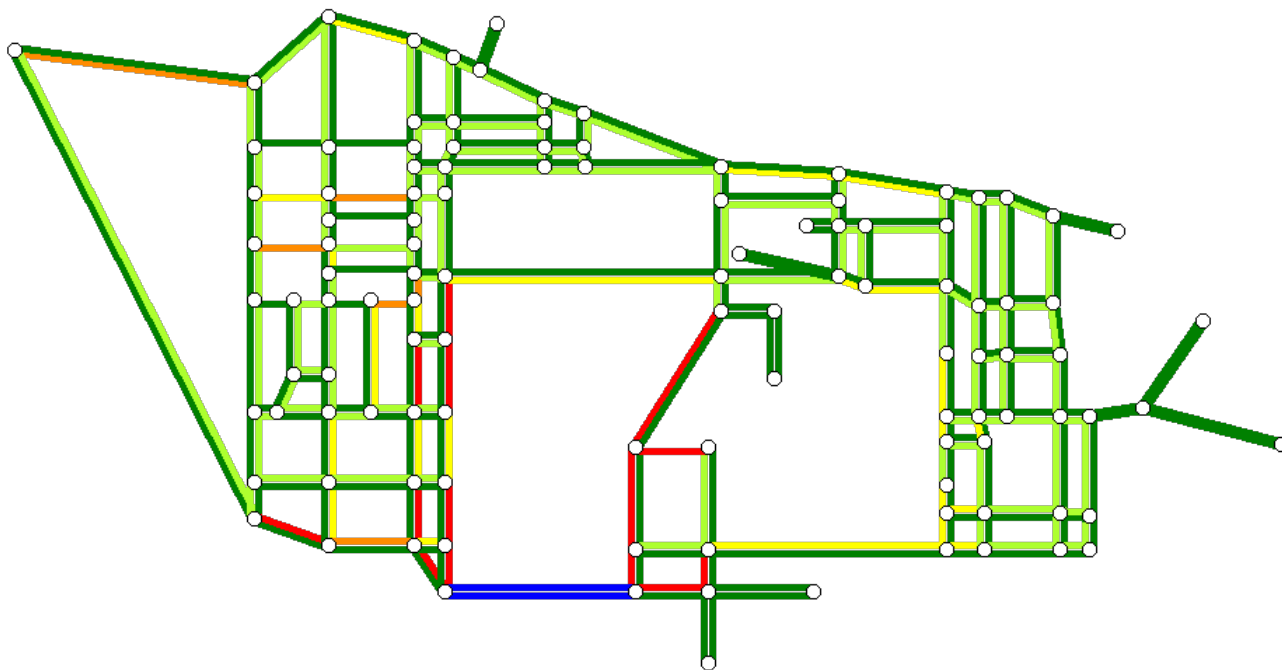


Рис. 6: Макаровская ул. полностью перекрыта на ремонт

В случае, если Макаровская ул. (отмечена бирюзовым на Рис. 7) перекрыта частично, результаты таковы:.

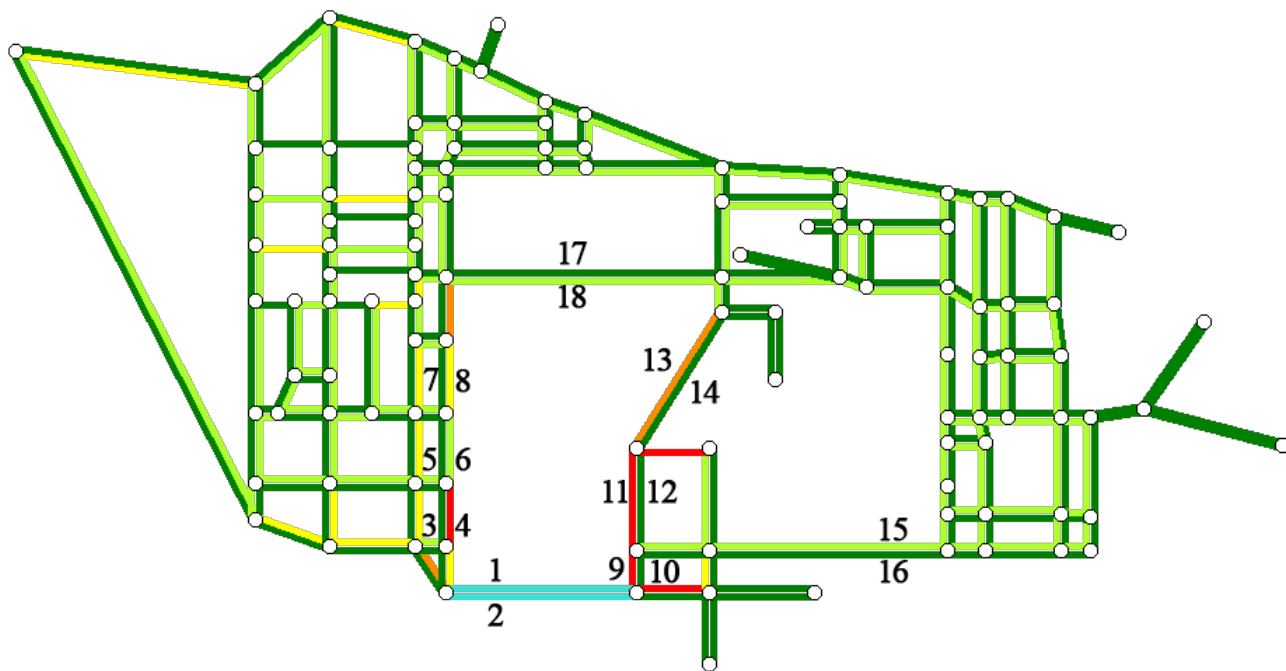


Рис. 7: Движение по Макаровской ул. ограничено

Числовые данные, отображающие поток на дорогах, отмеченных на картинке, выглядят так:

№	Название	Исходный поток	Поток при перекрытой дороге	Поток при огранич. движении
1	Макаровская ул.	4.690	0	2
2		0.213	0	0.004
3	Пр. Ленина	1.174	0.559	0.778
4		0.851	1.788	1.284
5		0.167	0.220	0.188
6		4.535	5.963	5.302
7		1.017	0.575	0.742
8		0.982	1.737	1.347
9	Коммунистич. ул.	0.399	1.437	0.798
10		2.502	0.695	1.251
11		0.283	0.784	0.474
12		3.526	1.274	2.106
13		0.187	0.381	0.267
14		5.329	2.618	3.736
15	Петровская ул.	5.836	7.372	6.620
16		0.171	0.135	0.151
17	Советская ул.	0.069	0.055	0.061
18		14.370	18.004	16.380

Полное закрытие этой улицы существенно влияет на дорожную ситуацию практически во всем городе (кроме самой восточной его части), в то время как закрытие наполовину осложняет движение главным образом в центральной части, особенно на юге. Более предпочтительным кажется второй вариант.

5.4. Ремонт Красной ул.

При закрытии на ремонт Красной ул. (отмечена синим на Рис.‘8) затрудняется движение на юге города в непосредственной близости к улице:

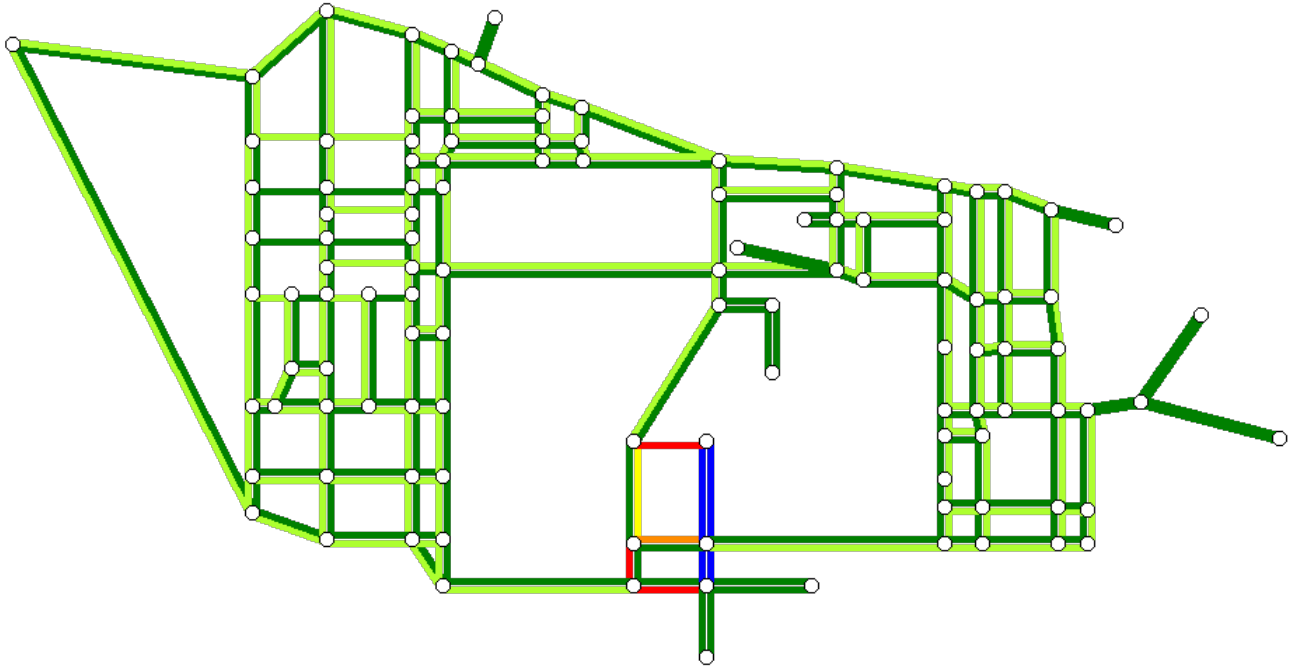


Рис. 8: Красная ул. полностью перекрыта на ремонт

Если Красная ул. (отмечена бирюзовым на Рис. 9) перекрыта на ремонт частично, результаты выглядят так:

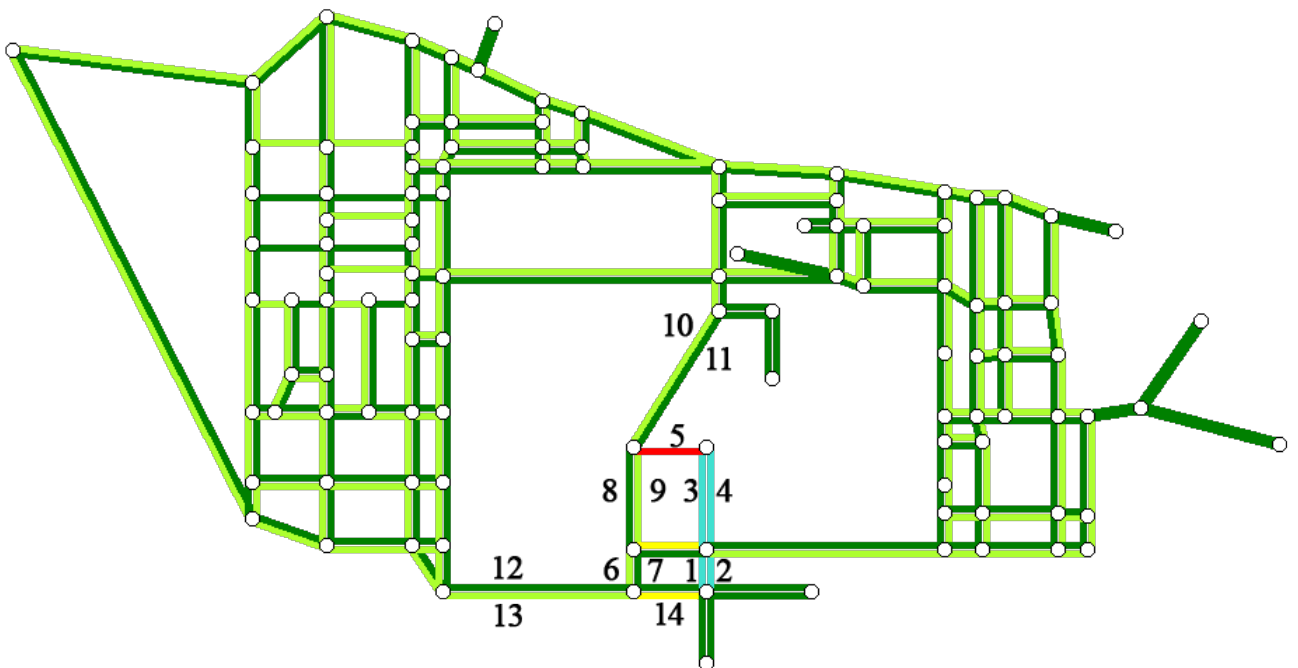


Рис. 9: Движение по Красной ул. ограничено

А так выглядят числовые данные для отмеченных на рисунке дорог:

№	Название	Исходный поток	Поток при перекрытой дороге	Поток при огранич. движении
1	Красная ул.	0.475	0	0.067
2		2.101	0	1.3
3		1.546	0	0.149
4		0.646	0	0.051
5	Манежный пер.	0.100	1	0.901
6	Коммунистич. ул.	0.399	0.659	0.454
7		2.502	1.515	2.200
8		0.283	0.232	0.244
9		3.526	4.304	4.092
10		0.187	0.190	0.194
11		5.329	5.262	5.141
12	Макаровская ул.	0.213	0.197	0.211
13		4.690	5.053	4.725
14		1.204	2.414	1.454

Второй вариант, когда на ремонт закрыта только половина дороги, кажется более предпочтительным.

5.5. Ремонт Советской ул.

Если полностью перекрыть на ремонт Советскую ул. (на Рис. 10 отмечена синим), получим такие результаты:

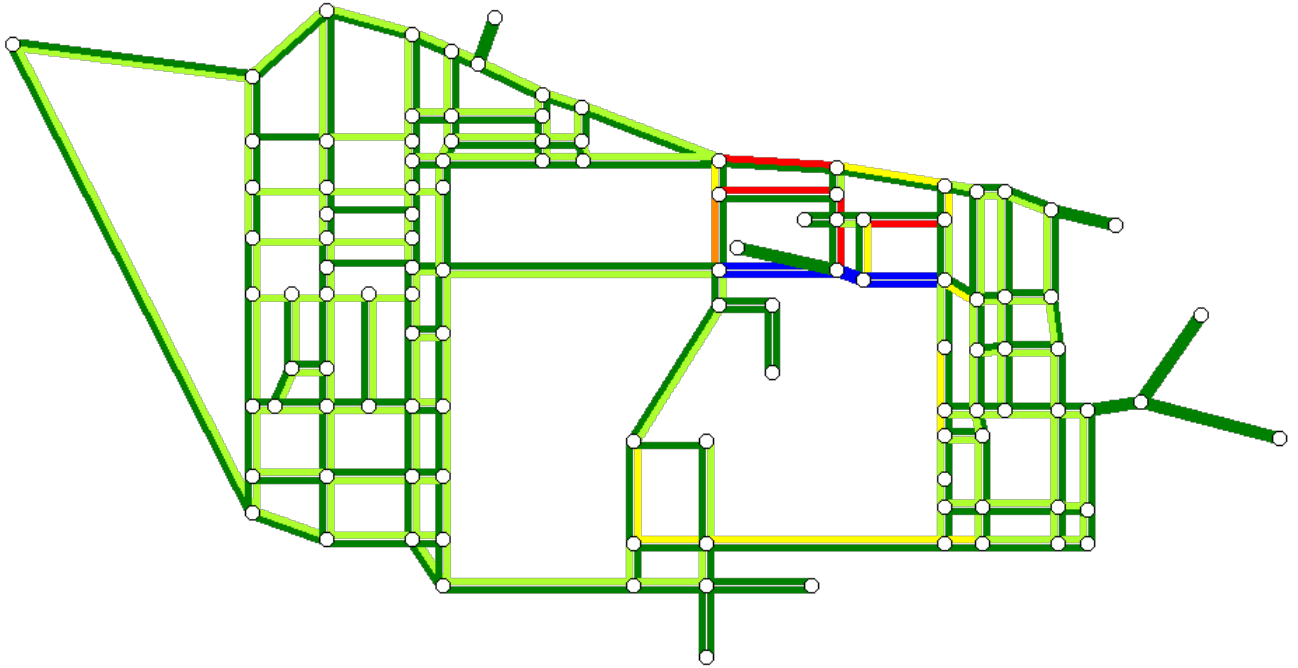


Рис. 10: Советская ул. полностью перекрыта на ремонт

Следующие результаты получаются в случае, если Советская ул. перекрыта наполовину (на Рис. 11 отмечена бирюзовым):

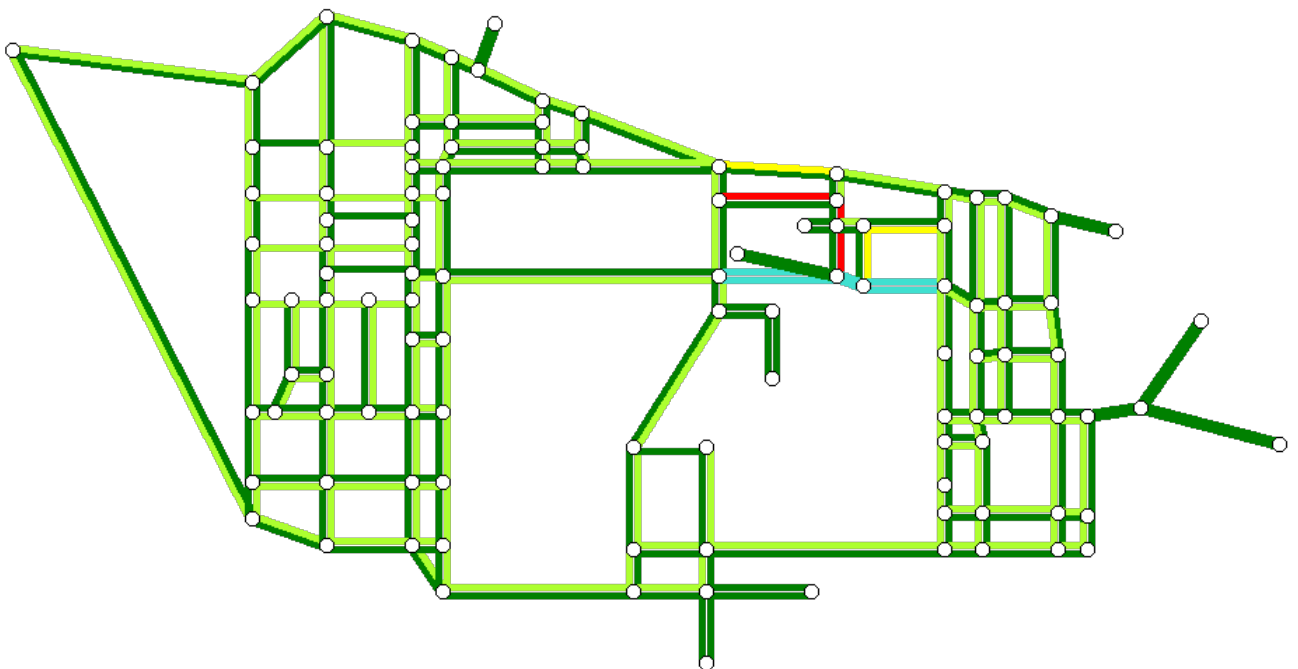


Рис. 11: Движение по Советской ул. ограничено

Таким образом, ремонт Советской ул. локально усложняет ситуацию в непосредственной близости к этой улице и совсем немного влияет на движение по

остальным дорогам города.

6. Выводы и рекомендации по построению плана ремонта дорог города Кронштадт

Как видно, практически во всех случаях (кроме ул. Мартынова) закрытие на ремонт половины дороги оказывается более предпочтительным, чем перекрытие дороги полностью.

Поскольку ремонт Цитадельской дороги затрудняет движение на западе и в центральной части города, не рекомендуется проводить ремонт Цитадельской дороги одновременно с ремонтом какой-либо еще улицы, а особенно — Макаровской, которая заметно влияет на движение на юго-западе.

Не рекомендуется одновременно ремонтировать ул. Мартынова и Макаровскую ул. (результат можно видеть на Рис. 12) из-за большого увеличения потоков на дорогах юго-запада и центральной части города.

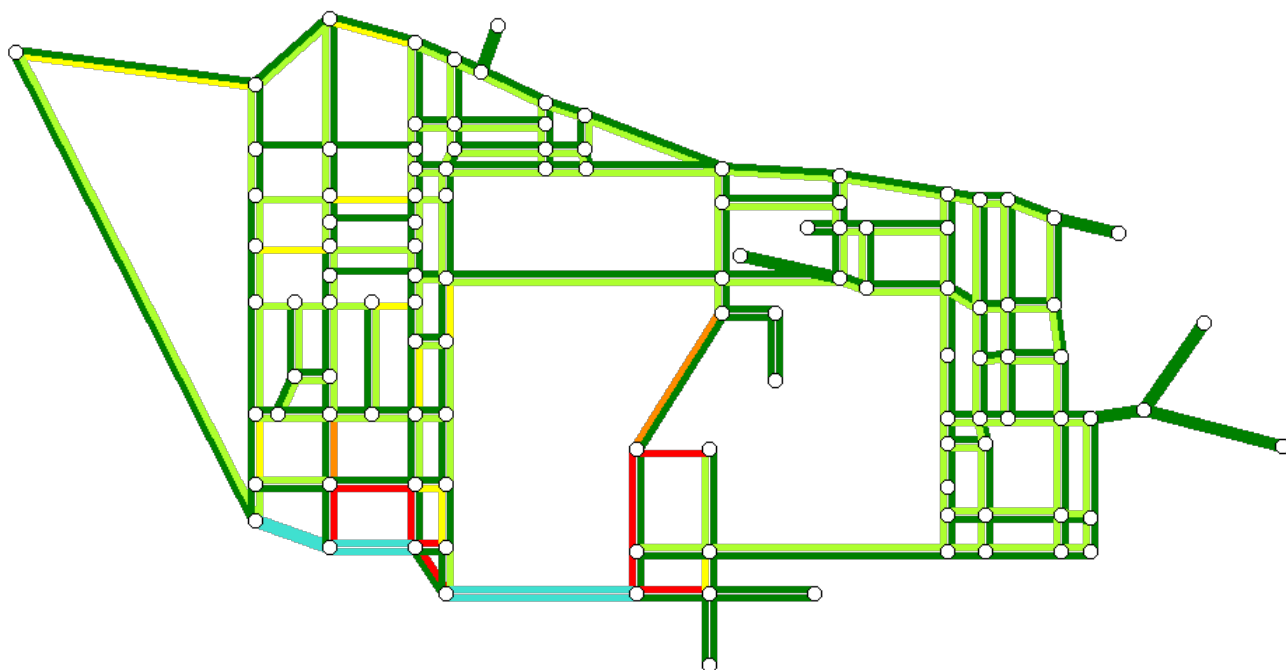


Рис. 12: Ремонт ул. Мартынова и Макаровской ул.

Не рекомендуется также одновременно ремонтировать Советскую ул. и Макаровскую ул. (Рис. 13), поскольку повышается нагрузка на многих участках, особенно в центре города. Кроме того, при проведении такого ремонта становится сложно попасть с севера города на юг: на любом пути с севера на юг

кроме одного есть дорога, нагрузка на которую увеличена более чем на 20 %, либо дорога, перекрытая на ремонт.

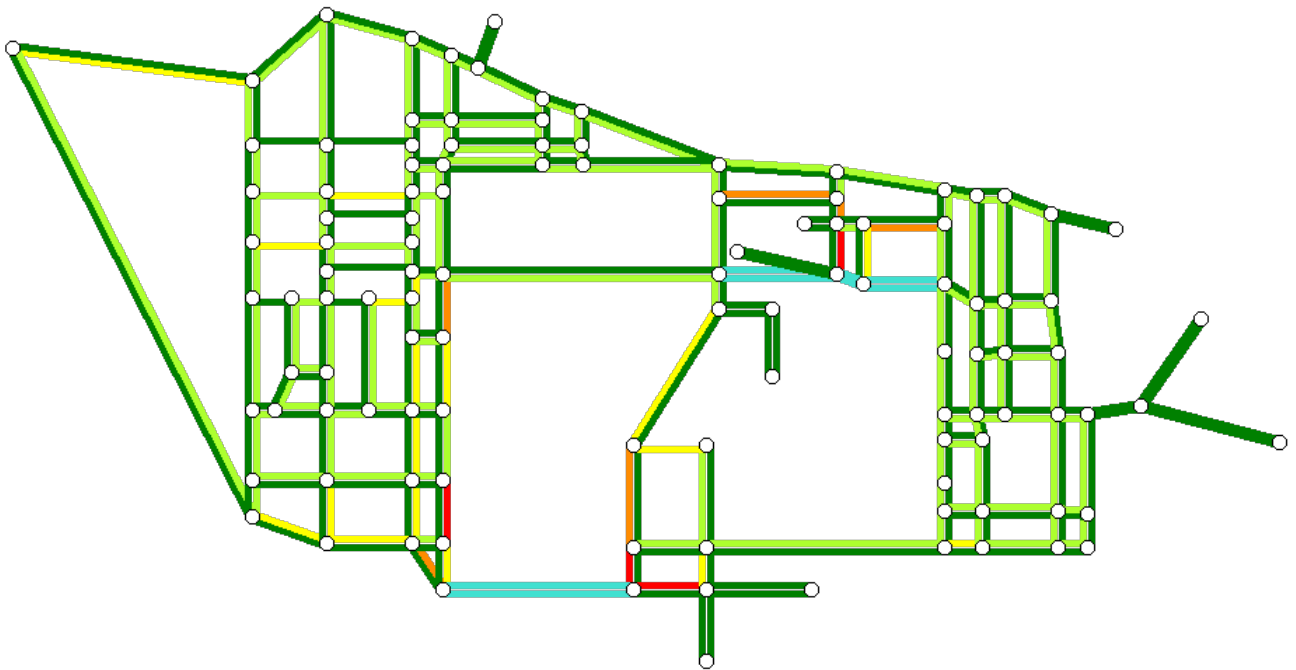


Рис. 13: Ремонт Советской ул. и Макаровской ул.

Более предпочтительным кажется вариант одновременного ремонта Советской ул. и ул. Мартынова (результат представлен на Рис. 14).

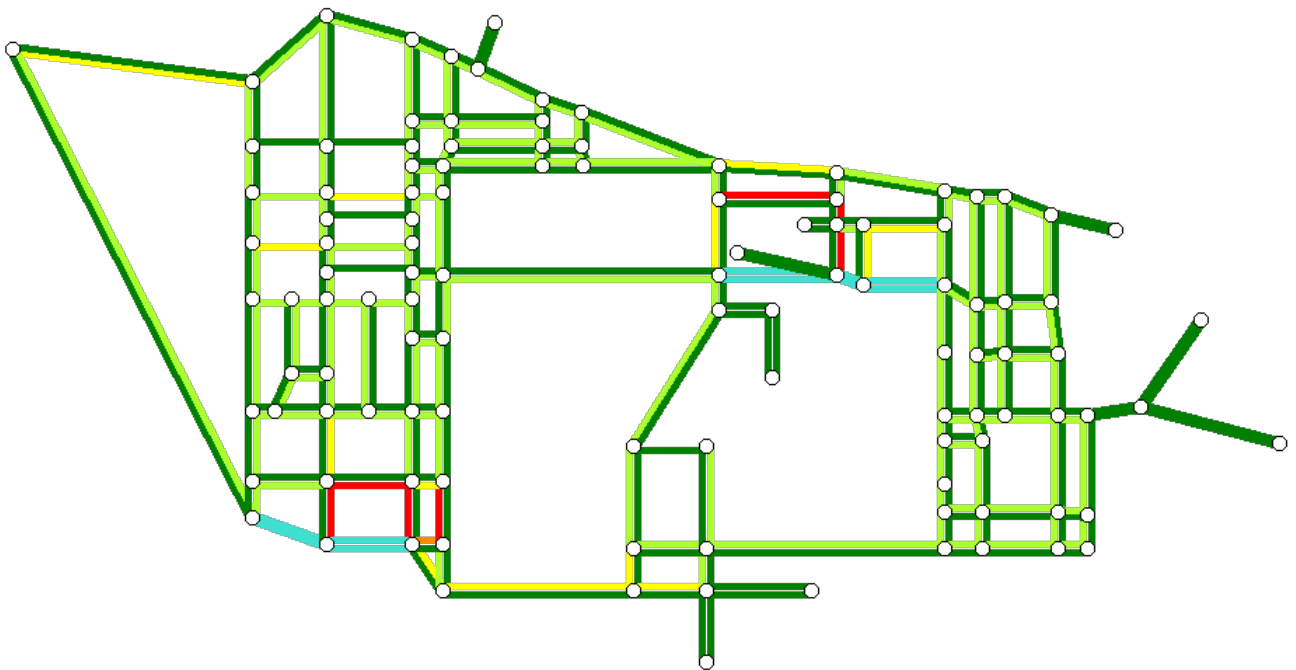


Рис. 14: Ремонт Советской ул. и ул. Мартынова

Также стоит отметить вариант одновременного ремонта Советской ул. и Красной ул. (Рис. 15): центр оказывается более загруженным, зато нет большого влияния на ситуацию на западе и востоке.

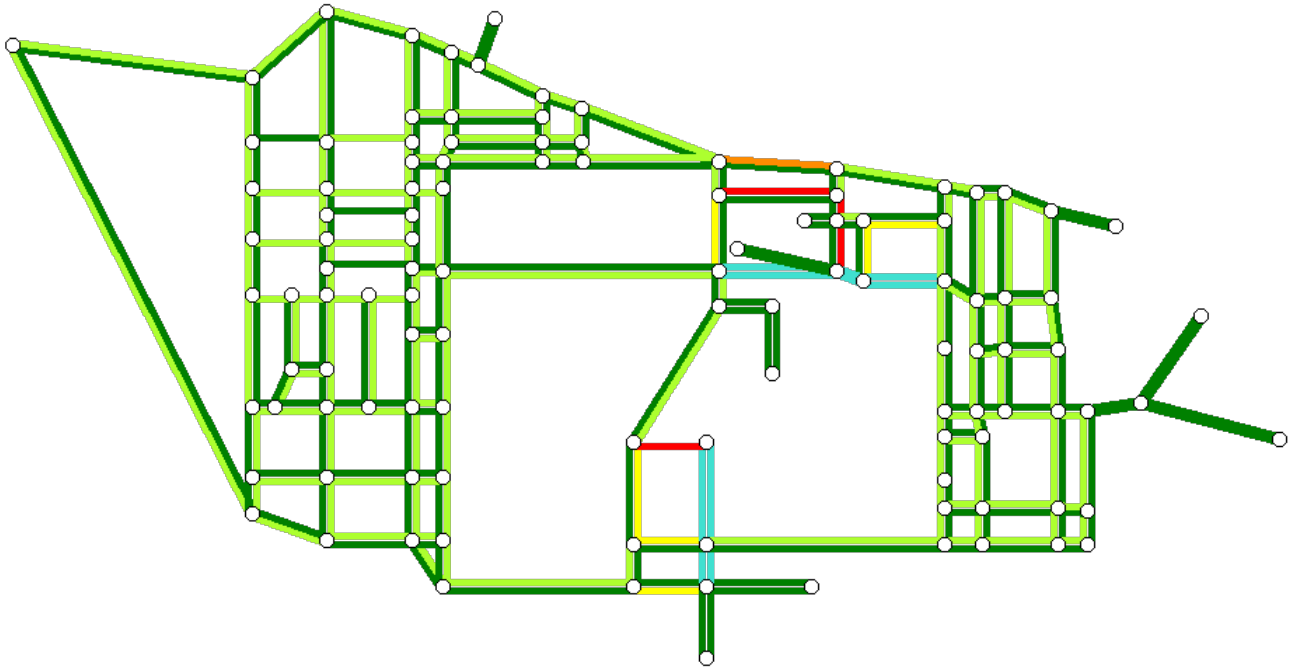


Рис. 15: Ремонт Советской ул. и Красной ул.

Ремонт Красной ул. главным образом сказывается на очень близких к ней дорогах, поэтому кажется возможным ремонтировать одновременно с ней и ул. Мартынова (Рис. 16):



Рис. 16: Ремонт Красной ул. и ул. Мартынова

И Макаровскую ул. (Рис. 17):



Рис. 17: Ремонт Красной ул. и Макаровской ул.

Хотя в обоих случаях возникают участки, поток машин по которым существенно увеличивается.

Исходя из увеличения потоков на дорогах, возникающего при перекрытии

на ремонт уже двух дорог, вероятно, не стоит одновременно ремонтировать три и более дороги. Единственный вариант ремонта трех дорог одновременно, который кажется приемлемым — ремонт Красной ул., Советской ул. и ул. Мартынова (Рис. 18).

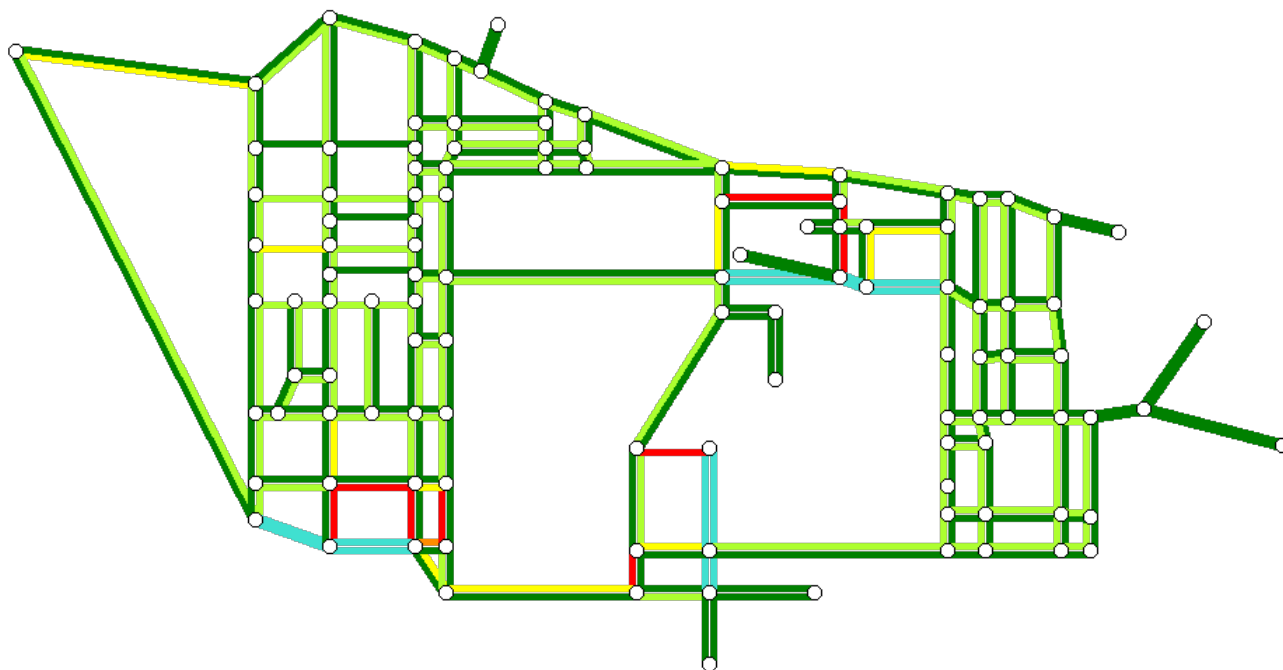


Рис. 18: Ремонт Красной ул., Советской ул. и ул. Мартынова

Алгоритм Брэгмана сходится достаточно быстро: на обработку каждого из рассмотренных выше случаев ушло от 8 до 14 секунд. При этом большинство случаев обрабатывалось не более, чем за 10 секунд, а самое большое время достигалось в тех ситуациях, где перекрывалась на ремонт Макаровская ул. (11 секунд для полностью перекрытой Макаровской ул. и 14 секунд для перекрытых ул. Мартынова и Макаровской ул.). Параметр, управляющий точностью, имел высокое значение — нам было интересно поведение программы. Если потребовать меньшую точность вычислений, можно добиться более быстрой работы программы.

Метод Брэгмана можно успешно применять для моделирования движения на дорогах города. Его существенным достоинством является быстрота, поэтому его можно применять как для небольших, так и для больших наборов данных: начиная с участков, состоящих из нескольких улиц, и заканчивая целыми районами города или городами. Конечно, для того, чтобы надеяться на правдоподобный результат необходимо иметь более достоверные данные, описыва-

ющие дороги и потребности вершин графа.

7. Приложение

Так может выглядеть подпрограмма на языке C#, реализующая модифицированный метода Брэгмана. По заданным вектору “весов” w , вектору ограничений r , вектору потребностей вершин b , матрице инциденций графа a и целым числам n, m — количеству вершин и дуг соответственно, программа генерирует вектор x потоков на дугах.

```
static void Bregman(int[,] a, double[] w, int n, int m,
double[] b, double[] r, out double[] x) {
    double[] xk = new double[m];
    double[] xp = new double[m];
    double[] xadd = new double[m];
    double eps = 0.0000000001;

    for (int i = 0; i < m; i++) {
        xk[i] = w[i];
        xadd[i] = 0;
    }

    double S = 0;
    double S1 = 0;
    double S2 = 0;
    int counter = 0;

    while (counter != m) {
        for (int i = 0; i < n; i++) {
            S = 0;
            S1 = 0;
            S2 = 0;

            for (int l = 0; l < m; l++) {
                xp[l] = xk[l];
```

```

    xadd[l] = 0;
}

for (int j = 0; j < m; j++) {
    if (a[i, j] == 1)
        S1 = S1 + xk[j];
    else if (a[i, j] == -1)
        S2 = S2 + xk[j];
}

if ((S1 == 0) || (S2 == 0)) {
    if (b[i] != 0) {
        double ttemp = 0;
        if ((S1 == 0) & (S2 != 0)) ttemp = -S2 / b[i];
        else if ((S1 != 0) & (S2 == 0)) ttemp = b[i] / S1;

        for (int j = 0; j < m; j++) {
            if (a[i, j] != 0) {
                double temp = ttemp;
                if (a[i, j] == -1) temp = 1 / ttemp;

                if (xk[j] * temp <= r[j])
                    xk[j] = xk[j] * temp;

                else if (xk[j] * temp > r[j]) {
                    int k = 0;
                    for (int l = 0; l < m; l++) {
                        if ((a[i, l] == a[i, j]) &&
                            (xk[l] < r[l]) && (l != j)) k++;
                    }
                    if (k == 0)
                        xk[j] = xk[j] * temp;
                }
            }
        }
    }
}

```

```

else if (k > 0) {
    bool p = false;
    double z;
    int kk;
    while (p) {
        z = (xk[j] * temp - r[j]) / k;
        kk = 0;
        for (int l = 0; l < m; l++) {
            if ((a[i, j] == a[i, l]) && (j != l)
                && (xk[l] + z + xadd[l] < r[l]))
                kk++;
        }
        if (kk == k) {
            for (int l = 0; l < m; l++) {
                if ((a[i, j] == a[i, l]) && (j != l)
                    && (xk[l] + z + xadd[l] < r[l])) {
                    xadd[l] = xadd[l] + z;
                    p = true;
                }
            }
        }
        else {
            k = kk;
        }
    }
    xk[j] = r[j];
}

}

}

for (int l = 0; l < m; l++) {
    xk[l] = xk[l] + xadd[l];
}

```

```

        xadd[l] = 0;
    }

}

}

else {

    S = (b[i] + (Math.Sqrt(b[i] * b[i] +
    4 * S1 * S2))) / (2 * S1);

    for (int j = 0; j < m; j++) {
        if ((a[i, j] == 1) && (xk[j] * S <= r[j]))
            xk[j] = xk[j] * S;
        else if ((a[i, j] == -1) && (xk[j] / S <= r[j]))
            xk[j] = xk[j] / S;
        else if ((a[i, j] == 1) && (xk[j] * S > r[j])) {
            int k = 0;
            for (int l = 0; l < m; l++) {
                if ((a[i, l] == a[i, j]) &&
                    (xk[l] < r[l]) && (l != j)) k++;
            }
            if (k == 0) xk[j] = xk[j] * S;
            else if (k > 0) {
                bool p = false;
                double z;
                int kk;
                while (p) {
                    z = (xk[j] * S - r[j]) / k;
                    kk = 0;
                    for (int l = 0; l < m; l++) {
                        if ((a[i, j] == a[i, l]) && (j != l)
                            && (xk[l] + z + xadd[l] < r[l]))

```



```

        kk++;
    }
    if (kk == k) {
        for (int l = 0; l < m; l++) {
            if ((a[i, j] == a[i, l]) && (j != l)
                && (xk[l] + z + xadd[l] < r[l])) {
                xadd[l] = xadd[l] + z;

                p = true;
            }
        }
    }
    else {
        k = kk;
    }
}
xk[j] = r[j];
}

}

else if ((a[i, j] == -1) && (xk[j] / S > r[j])) {
    int k = 0;
    for (int l = 0; l < m; l++) {
        if ((a[i, l] == a[i, j]) &&
            (xk[l] < r[l]) && (l != j)) k++;
    }
    if (k == 0) xk[j] = xk[j] / S;
    else if (k > 0) {
        bool p = false;
        double z;
        int kk;
        while (p) {

```

```

z = (xk[j] / S - r[j]) / k;
kk = 0;
for (int l = 0; l < m; l++) {
    if ((a[i, j] == a[i, l]) && (j != l)
        && (xk[l] + z + xadd[l] < r[l]))
        kk++;
    //xadd[l] = xadd[l] + z;
}
if (kk == k) {
    for (int l = 0; l < m; l++) {
        if ((a[i, j] == a[i, l]) && (j != l)
            && (xk[l] + z + xadd[l] < r[l])) {
            xadd[l] = xadd[l] + z;

            p = true;
        }
    }
}
else {
    k = kk;
}
}
xk[j] = r[j];
}
}
for (int l = 0; l < m; l++) {
    xk[l] = xk[l] + xadd[l];
    xadd[l] = 0;
}

}
}
counter = 0;

```

```
    for (int l = 0; l < m; l++) {
        if ((xk[l] >= xp[l] - eps) & (xk[l] <= xp[l] + eps))
            counter++;
    }
}
x = xk;
}
```

Литература

- [1] Шелейховский Г. В. Транспортные основания композиции городского плана. Л.: Гипрогор, 1936.
- [2] Брэгман Л. М. Нахождение общей точки выпуклых множеств методом последовательного проектирования. Докл. АН СССР, 1965, 162, №3, с. 487–490.
- [3] Брэгман Л. М. Доказательство сходимости метода Г. В. Шелейховского для задачи с транспортными ограничениями. Журн. вычислит. мат. и мат. физики, 1967, т.7, №1, с. 147–156.
- [4] Брэгман Л. М. Релаксационный метод нахождения общей точки выпуклых множеств и его применение для решения задач выпуклого программирования. Журн. вычислит. мат. и мат. физики, 1967, т.7, №3, с. 620–631.
- [5] Тищенко И. Закатали в асфальт: как отремонтируют петербургские дороги летом [Электронный ресурс, Дата обращения: 18.05.2016]
URL: <http://www.spbdnevnik.ru/news/2016-02-29/zakatali-v-asfalt-kak-otremontiruyut-peterburgskiey-dorogi-letom>