



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ: ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА: КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ

НАПРАВЛЕНИЕ ПОДГОТОВКИ: 09.03.01 Информатика и вычислительная техника

О Т Ч Е Т

по домашнему заданию

Тема: Поток в транспортной сети. Алгоритм Форда-Фалкерсона

Дисциплина: Дискретная математика

Студент

ИУ6-42Б
(Группа)

13.05.24
(Подпись, дата) О. В. Жданов
(И. О. Фамилия)

Преподаватель

13.05.24
(Подпись, дата) В. В. Гуренко
(И. О. Фамилия)

Москва, 2024

1. Задание, теоретические сведения, построение графа, определение источника и стока

1.1. Задание

Сеть в виде взвешенного орграфа задана матрицей Ω пропускных способностей ориентированных ребер. При помощи алгоритма Форда – Фалкерсона определить максимальный поток φ_{\max} , доставляемый от источника $s = x_1$ к стоку $s = x_{12}$ и указать минимальный разрез, отделяющий t от s . Оптимизационную часть алгоритма реализовать в виде коррекции потока хотя бы на одном увеличивающем маршруте.

Матрица пропускных способностей (вариант 19):

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}
x_1		13		46	26							
x_2							20					
x_3						12						
x_4						21					24	
x_5			27				15	14				
x_6									16			
x_7								8		13		31
x_8			33							22		
x_9								4		10		18
x_{10}											17	50
x_{11}								22				7
x_{12}												

1.2. Теоретические сведения для решения задания

1.2.1. Теорема 1

Если $(s, \dots, x_i, \dots, x_j, \dots, t)$ – путь от s к t и все ребра этого пути ненасыщенные, то поток на этом пути (а следовательно, и во всей сети) можно увеличить на величину $\delta^* = \min\{\delta(x_i, x_j)\}$ по всем ребрам пути.

При применении теоремы 1 хотя бы одно ребро становится насыщенным. При итерационном применении теорема 1 появляется ситуация, когда нет ни одного пути от источника к стоку, на котором все ребра были бы ненасыщенными. Эта ситуация говорит о полном потоке.

1.2.2. Теорема 2

Рассмотрим маршрут из источника к стоку. Пусть на этом маршруте прямые рёбра строго ненасыщенные, а поток на обратных рёбрах строго положителен. Тогда, если $(s, \dots, x_i, \dots, x_j, \dots, t)$ – маршрут из s к t , удовлетворяющий указанным условиям, то величину потока на этом маршруте (следовательно, и во всей сети) можно увеличить на величину $\varepsilon^* = \min\{\delta^*, \varphi^*\}$, где δ^* – минимальная остаточная пропускная способность по всем прямым ребрах маршрута, а φ^* – минимальная величина потоков по всем обратным ребрам маршрута. На всех прямых ребрах маршрута потоки увеличиваются на ε^* , на обратных – уменьшаются на ε^* . Такой маршрут называется увеличивающим.

1.2.3. Теорема 3

Поток в сети достигает максимального значения тогда и только тогда, когда в сети более невозможно построить увеличивающий маршрут.

1.2.4. Теорема 4 (Теорема Форда – Фалкерсона)

Для любой сети с одним источником и одним стоком величина максимального потока φ_{\max} , доставляемого от источника к стоку, равна пропускной способности минимального разреза.

1.2.5. Алгоритм Форда – Фалкерсона

- Задать начальный поток.
- Итерационно применяя теорему 1, получить полный поток.
- Итерационно применять теорему 2. При возникновении ситуации невозможности построить более увеличивающий маршрут на основании теоремы 3 достигаем максимальный поток.
- Определить разрез, по которому проходит максимальный поток.

1.3. Построенный граф, его источник и сток

Построим взвешенный ориентированный граф по матрице Ω пропускных способностей ориентированных ребер (рис. 1).

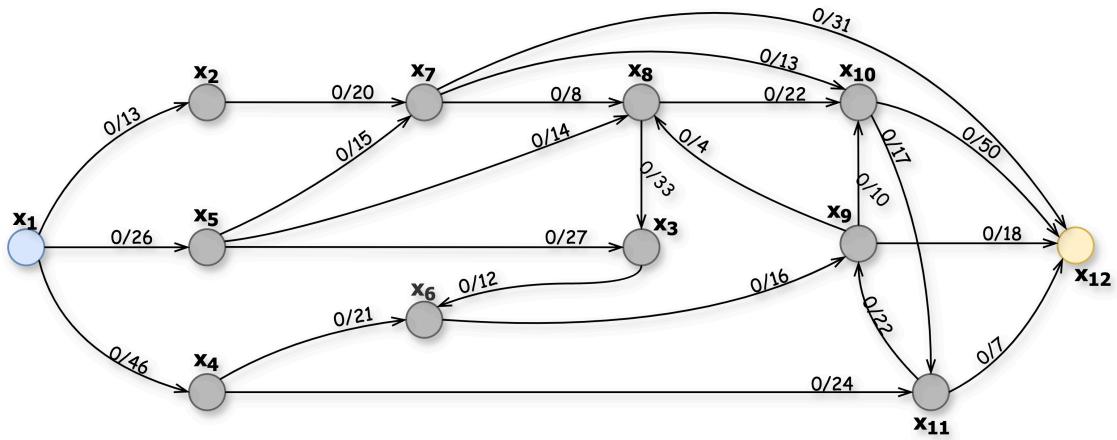


Рисунок 1 — Построенный граф

Источником будет вершина $x_1 = s$, стоком – вершина $x_2 = t$. Поток $\varphi = 0$.

2. Получение полного потока

Определим, итерационно применяя теорему 1, полный поток φ_n .

2.1. Путь $(x_1, x_2, x_7, x_8, x_{10}, x_{11}, x_{12})$

Рассмотрим путь $(x_1, x_2, x_7, x_8, x_{10}, x_{11}, x_{12})$.

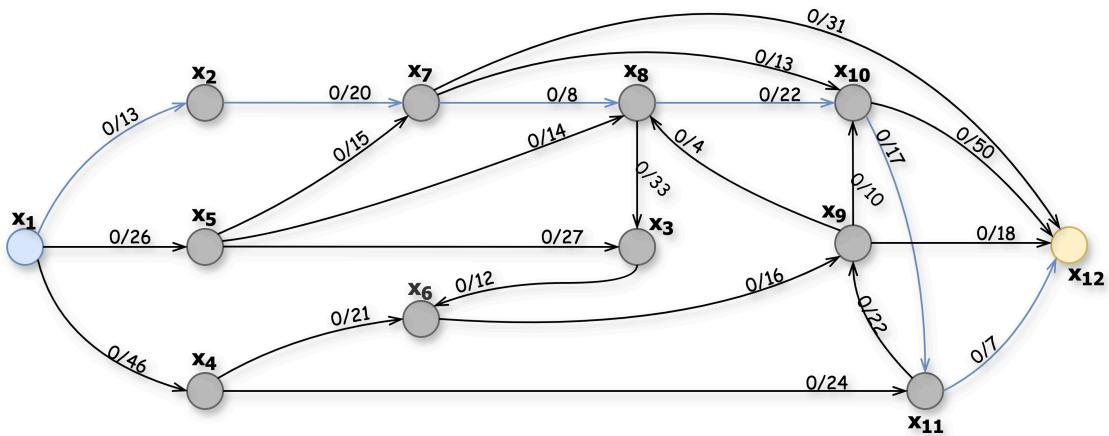


Рисунок 2 — Путь $(x_1, x_2, x_7, x_8, x_{10}, x_{11}, x_{12})$, отмеченный на графике

Рассчитаем остаточную пропускную способность пути:

$$\begin{aligned} \delta^* &= \min\{ c(x_1, x_2) - \varphi(x_1, x_2), c(x_2, x_7) - \varphi(x_2, x_7), \\ &\quad c(x_7, x_8) - \varphi(x_7, x_8), c(x_8, x_{10}) - \varphi(x_8, x_{10}), \\ &\quad c(x_{10}, x_{11}) - \varphi(x_{10}, x_{11}), c(x_{11}, x_{12}) - \varphi(x_{11}, x_{12})\} = \\ &= \min\{13 - 0, 20 - 0, 8 - 0, 22 - 0, 17 - 0, 7 - 0\} = \\ &= \min\{13, 20, 8, 22, 17, 7\} = 7 \end{aligned}$$

$c(x_i, x_j)$ - пропускная способность ребра с вершинами x_i и x_j

$\varphi(x_i, x_j)$ - поток ребра с вершинами x_i и x_j

Добавим к потоку всех ребер, входящих в путь, остаточную пропускную способность пути. Ребра, пропускная способность которых будет равна потоку через них, станут насыщенными.

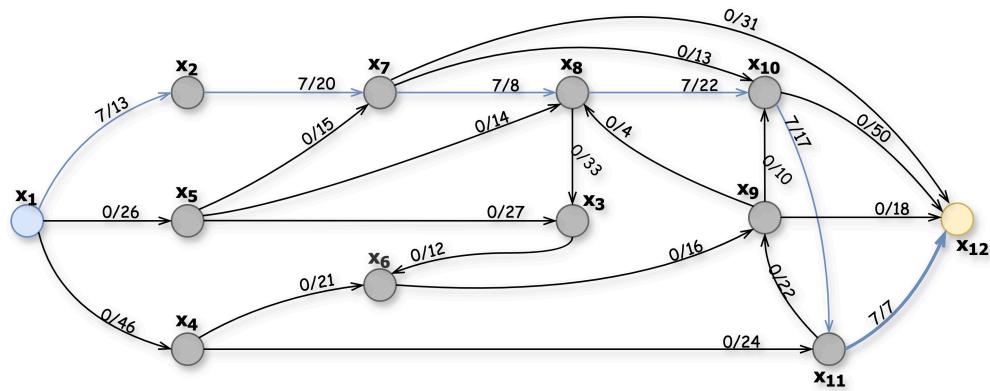


Рисунок 3 — Граф с добавленными потоками через ребра пути

В данном случае ребро (x_{11}, x_{12}) стало насыщенным.

Таким образом общий поток $\varphi_n = \varphi_n + \delta^* = 0 + \delta^* = 7$.

В результате, после выполнения всех действий получаем граф:

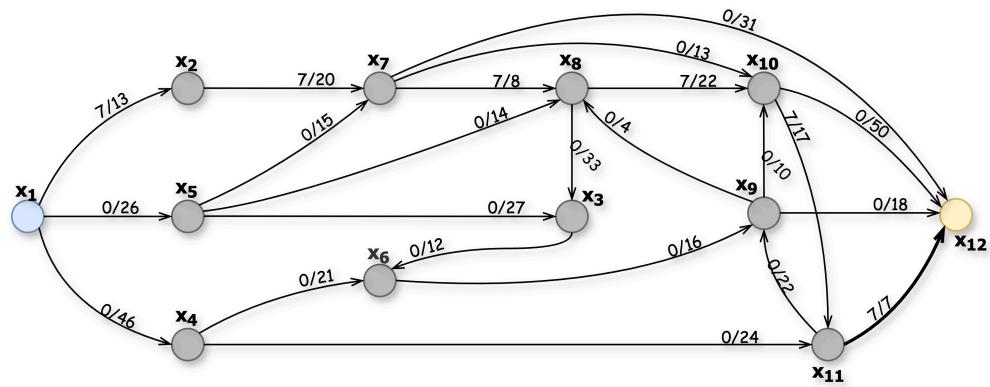


Рисунок 4 — Граф с текущими потоками

2.2. Путь $(x_1, x_2, x_7, x_{10}, x_{11}, x_9, x_{12})$

Рассмотрим путь $(x_1, x_2, x_7, x_{10}, x_{11}, x_9, x_{12})$.

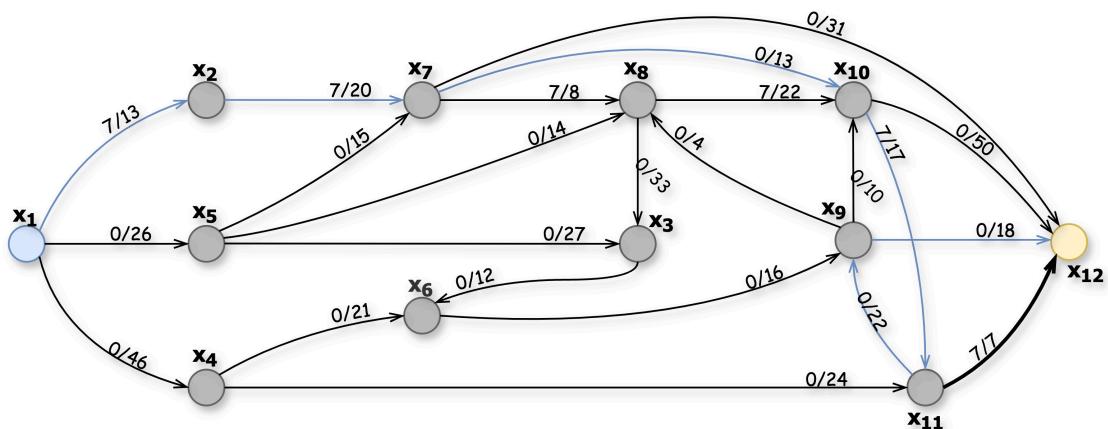


Рисунок 5 — Путь $(x_1, x_2, x_7, x_{10}, x_{11}, x_9, x_{12})$

Рассчитаем остаточную пропускную способность пути:

$$\begin{aligned}
 \delta^* &= \min\{ c(x_1, x_2) - \varphi(x_1, x_2), c(x_2, x_7) - \varphi(x_2, x_7), \\
 &\quad c(x_7, x_{10}) - \varphi(x_7, x_{10}), c(x_{10}, x_{11}) - \varphi(x_{10}, x_{11}), \\
 &\quad c(x_{11}, x_9) - \varphi(x_{11}, x_9), c(x_9, x_{12}) - \varphi(x_9, x_{12})\} = \\
 &= \min\{13 - 7, 20 - 7, 13 - 0, 17 - 7, 22 - 0, 18 - 0\} = \\
 &= \min\{6, 13, 13, 10, 22, 18\} = 6
 \end{aligned}$$

$c(x_i, x_j)$ - пропускная способность ребра с вершинами x_i и x_j

$\varphi(x_i, x_j)$ - поток ребра с вершинами x_i и x_j

Добавим к потоку всех ребер, входящих в путь, остаточную пропускную способность пути. Ребра, пропускная способность которых будет равна потоку через них, станут насыщенными.

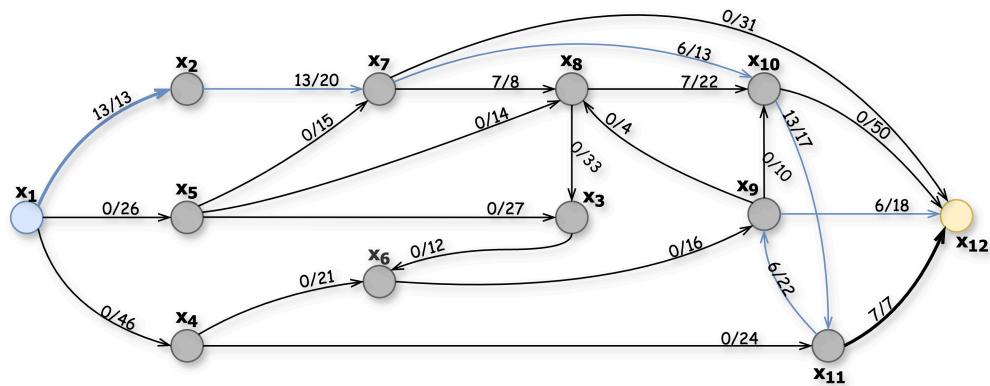


Рисунок 6 — Граф с добавленными потоками через ребра пути

В данном случае ребро (x_1, x_2) стало насыщенным.

Таким образом общий поток $\varphi_n = \varphi_n + \delta^* = 7 + 6 = 13$.

В результате, после выполнения всех действий получаем граф:

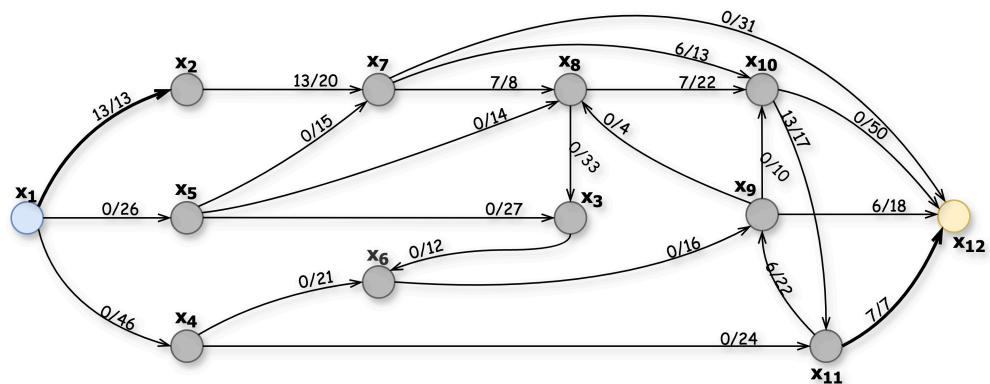


Рисунок 7 — Граф с текущими потоками

2.3. Путь (x_1, x_5, x_7, x_{12})

Рассмотрим путь (x_1, x_5, x_7, x_{12}) .

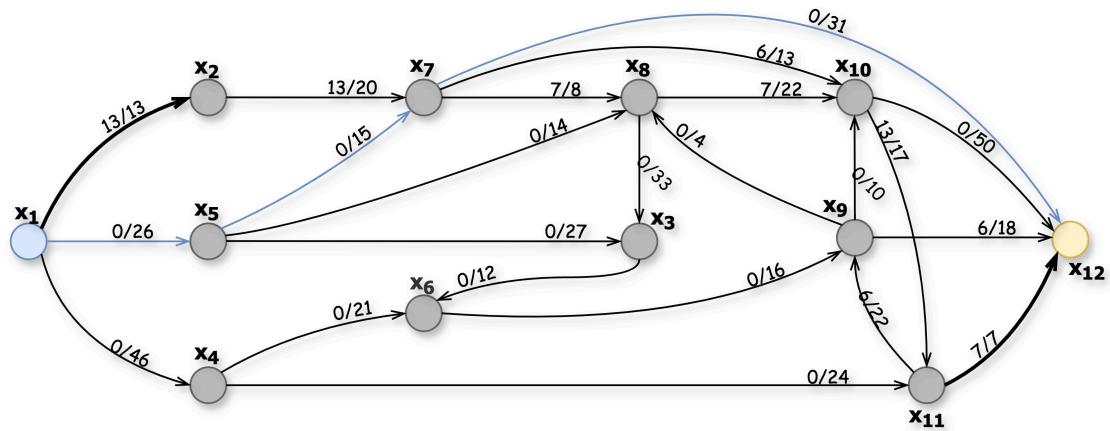


Рисунок 8 — Путь (x_1, x_5, x_7, x_{12})

Рассчитаем остаточную пропускную способность пути:

$$\begin{aligned} \delta^* &= \min\{ c(x_1, x_5) - \varphi(x_1, x_5), c(x_5, x_7) - \varphi(x_5, x_7), \\ &c(x_7, x_{12}) - \varphi(x_7, x_{12}) \} = \min\{26 - 0, 15 - 0, 31 - 0\} = \\ &= \min\{26, 15, 31\} = 15 \end{aligned}$$

$c(x_i, x_j)$ - пропускная способность ребра с вершинами x_i и x_j

$\varphi(x_i, x_j)$ - поток ребра с вершинами x_i и x_j

Добавим к потоку всех ребер, входящих в путь, остаточную пропускную способность пути. Ребра, пропускная способность которых будет равна потоку через них, станут насыщенными.

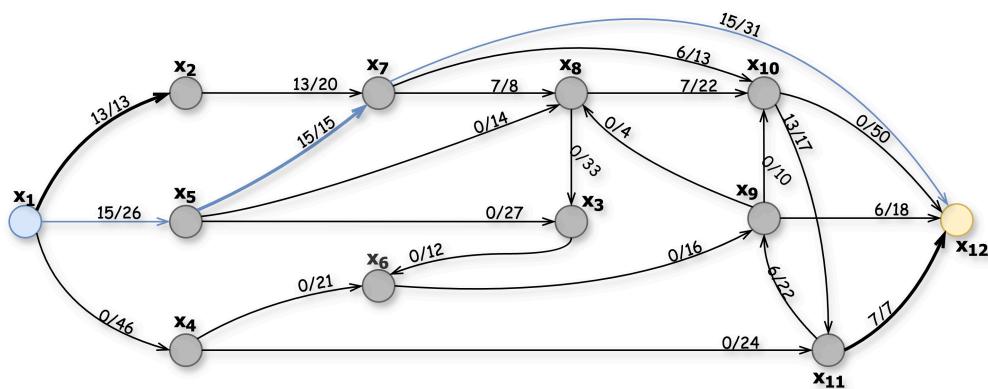


Рисунок 9 — Граф с добавленными потоками через ребра пути

В данном случае ребро (x_5, x_7) стало насыщенным.

Таким образом общий поток $\varphi_n = \varphi_n + \delta^* = 13 + \delta^* = 28$.

В результате, после выполнения всех действий получаем граф:

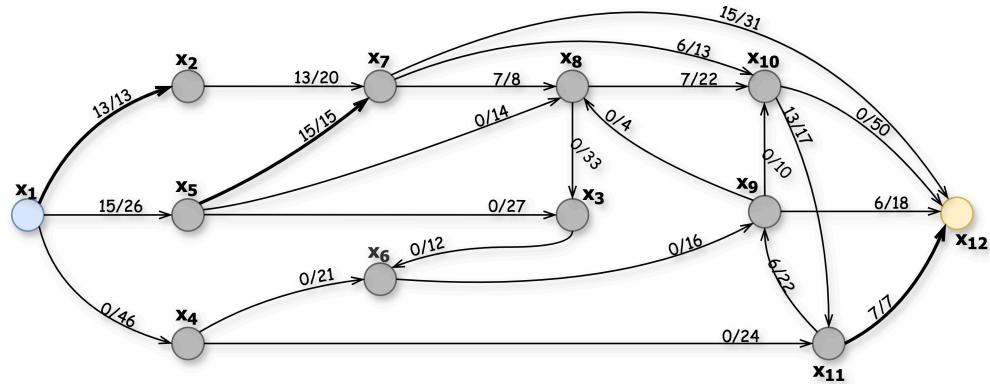


Рисунок 10 — Граф с текущими потоками

2.4. Путь $(x_1, x_5, x_8, x_{10}, x_{11}, x_9, x_{12})$

Рассмотрим путь $(x_1, x_5, x_8, x_{10}, x_{11}, x_9, x_{12})$.

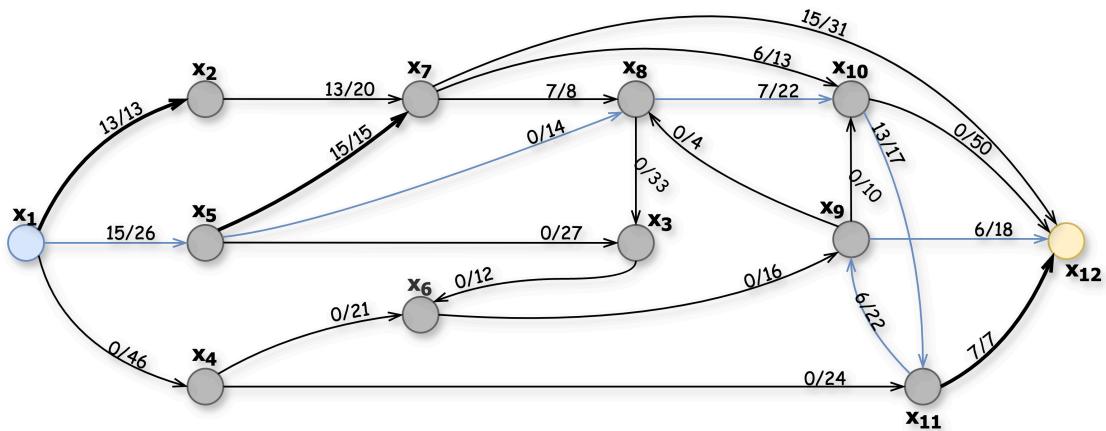


Рисунок 11 — Путь $(x_1, x_5, x_8, x_{10}, x_{11}, x_9, x_{12})$

Рассчитаем остаточную пропускную способность пути:

$$\begin{aligned}\delta^* &= \min\{ c(x_1, x_5) - \varphi(x_1, x_5), c(x_5, x_8) - \varphi(x_5, x_8), \\ &\quad c(x_8, x_{10}) - \varphi(x_8, x_{10}), c(x_{10}, x_{11}) - \varphi(x_{10}, x_{11}), \\ &\quad c(x_{11}, x_9) - \varphi(x_{11}, x_9), c(x_9, x_{12}) - \varphi(x_9, x_{12})\} \\ &= \min\{26 - 15, 14 - 0, 22 - 7, 17 - 13, 22 - 6, 18 - 6\} \\ &= \min\{11, 14, 15, 4, 16, 12\} = 4\end{aligned}$$

$c(x_i, x_j)$ - пропускная способность ребра с вершинами x_i и x_j

$\varphi(x_i, x_j)$ - поток ребра с вершинами x_i и x_j

Добавим к потоку всех ребер, входящих в путь, остаточную пропускную способность пути. Ребра, пропускная способность которых будет равна потоку через них, станут насыщенными.

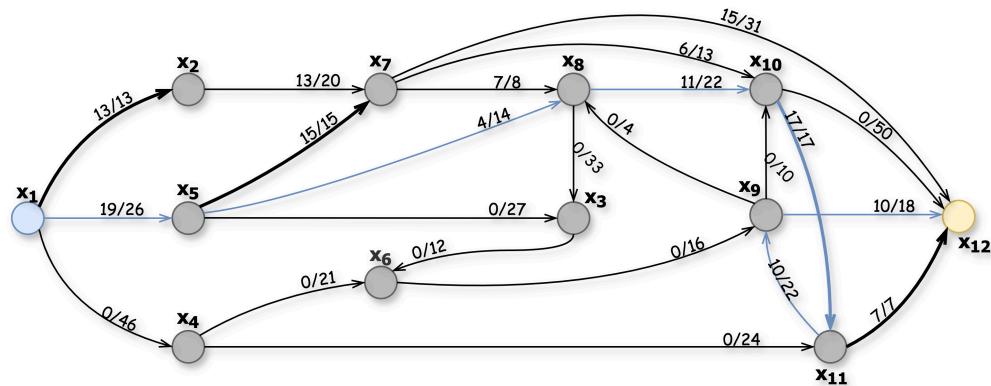


Рисунок 12 — Граф с добавленными потоками через ребра пути

В данном случае ребро (x_{10}, x_{11}) стало насыщенным.

Таким образом общий поток $\varphi_n = \varphi_n + \delta^* = 28 + \delta^* = 32$.

В результате, после выполнения всех действий получаем граф:

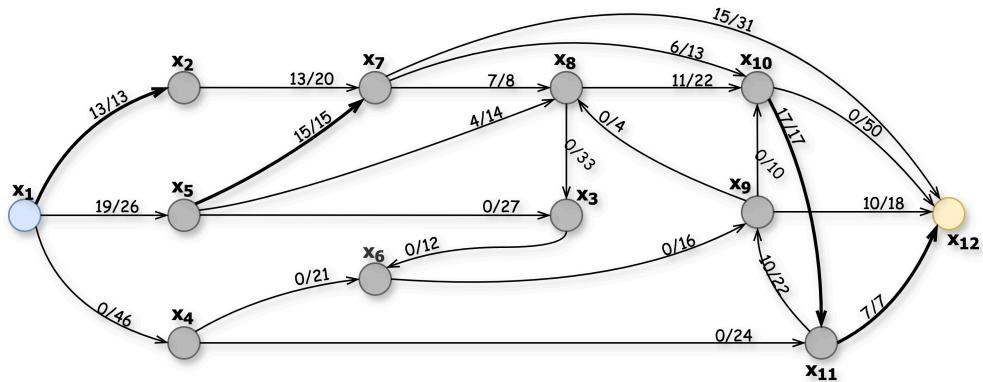


Рисунок 13 — Граф с текущими потоками

2.5. Путь $(x_1, x_5, x_8, x_{10}, x_{12})$

Рассмотрим путь $(x_1, x_5, x_8, x_{10}, x_{12})$.

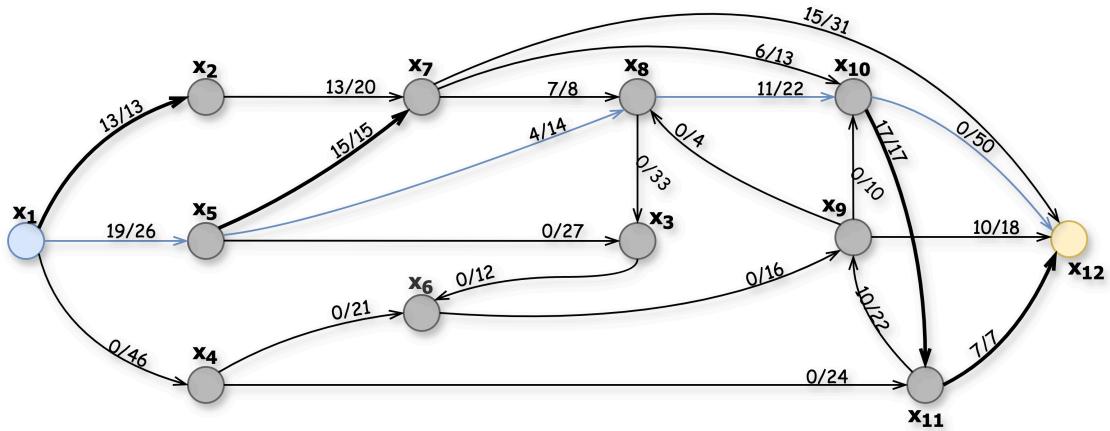


Рисунок 14 — Путь $(x_1, x_5, x_8, x_{10}, x_{12})$

Рассчитаем остаточную пропускную способность пути:

$$\begin{aligned}\delta^* &= \min\{ c(x_1, x_5) - \varphi(x_1, x_5), c(x_5, x_8) - \varphi(x_5, x_8), \\ &\quad c(x_8, x_{10}) - \varphi(x_8, x_{10}), c(x_{10}, x_{12}) - \varphi(x_{10}, x_{12})\} \\ &= \min\{26 - 19, 14 - 4, 22 - 11, 50 - 0\} \\ &= \min\{7, 10, 11, 50\} = 7\end{aligned}$$

$c(x_i, x_j)$ - пропускная способность ребра с вершинами x_i и x_j

$\varphi(x_i, x_j)$ - поток ребра с вершинами x_i и x_j

Добавим к потоку всех ребер, входящих в путь, остаточную пропускную способность пути. Ребра, пропускная способность которых будет равна потоку через них, станут насыщенными.

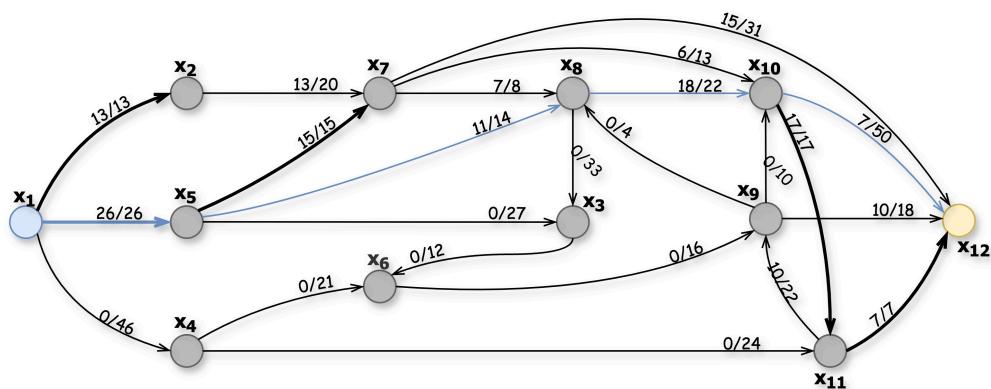


Рисунок 15 — Граф с добавленными потоками через ребра пути

В данном случае ребро (x_1, x_5) стало насыщенным.

Таким образом общий поток $\varphi_n = \varphi_n + \delta^* = 32 + \delta^* = 39$.

В результате, после выполнения всех действий получаем граф:

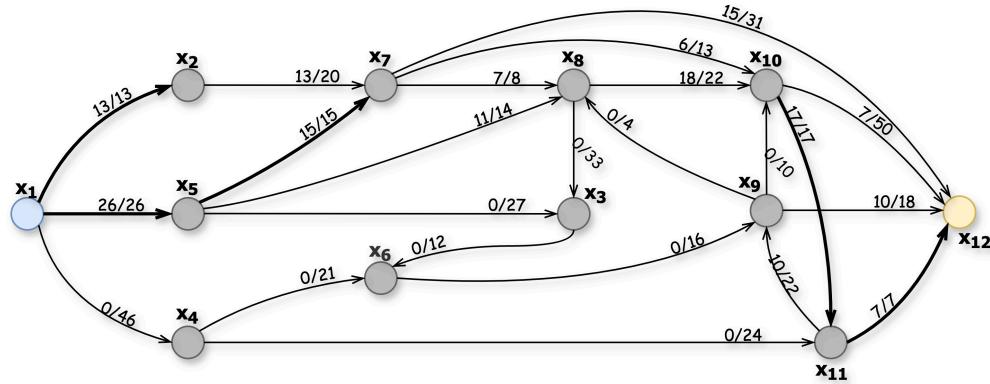


Рисунок 16 — Граф с текущими потоками

2.6. Путь $(x_1, x_4, x_6, x_9, x_8, x_{10}, x_{12})$

Рассмотрим путь $(x_1, x_4, x_6, x_9, x_8, x_{10}, x_{12})$.

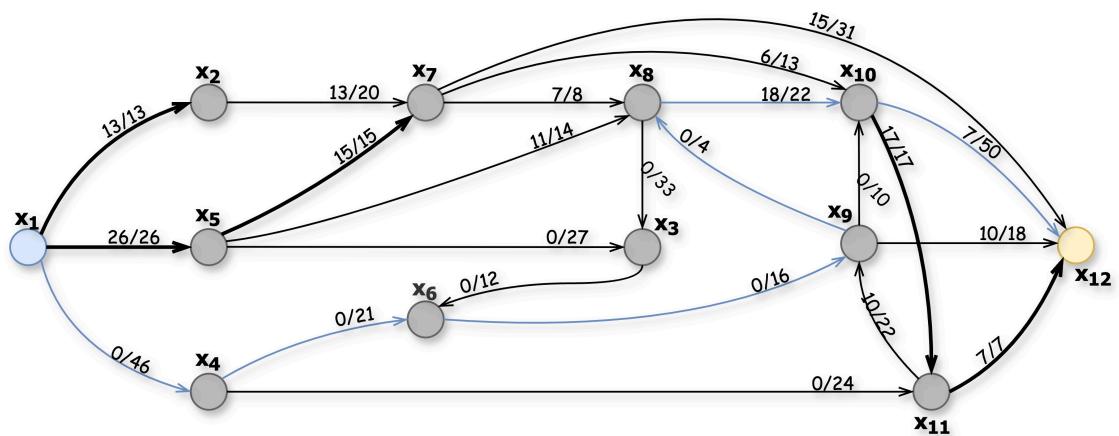


Рисунок 17 — Путь $(x_1, x_4, x_6, x_9, x_8, x_{10}, x_{12})$

Рассчитаем остаточную пропускную способность пути:

$$\begin{aligned} \delta^* &= \min\{ c(x_1, x_4) - \varphi(x_1, x_4), c(x_4, x_6) - \varphi(x_4, x_6), \\ &\quad c(x_6, x_9) - \varphi(x_6, x_9), c(x_9, x_8) - \varphi(x_9, x_8), \\ &\quad c(x_8, x_{10}) - \varphi(x_8, x_{10}), c(x_{10}, x_{12}) - \varphi(x_{10}, x_{12})\} \\ &= \min\{46 - 0, 21 - 0, 16 - 0, 4 - 0, 22 - 18, 50 - 7\} = \\ &= \min\{46, 21, 16, 4, 4, 43\} = 4 \end{aligned}$$

$c(x_i, x_j)$ - пропускная способность ребра с вершинами x_i и x_j

$\varphi(x_i, x_j)$ - поток ребра с вершинами x_i и x_j

Добавим к потоку всех ребер, входящих в путь, остаточную пропускную способность пути. Ребра, пропускная способность которых будет равна потоку через них, станут насыщенными.

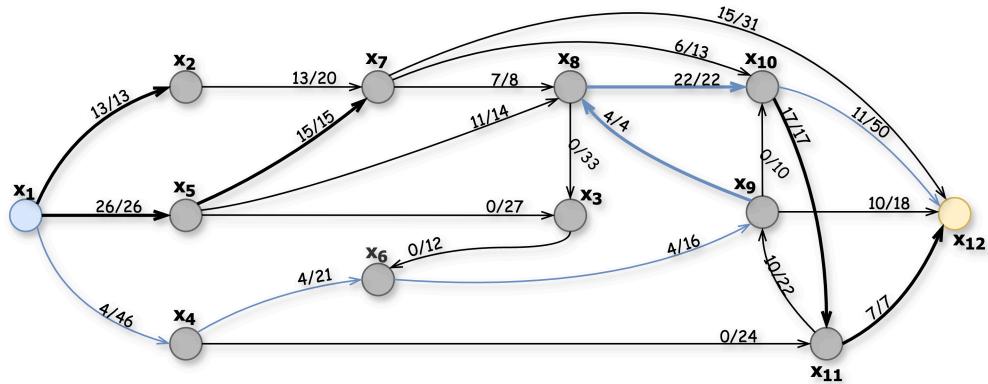


Рисунок 18 — Граф с добавленными потоками через ребра пути

В данном случае ребра (x_9, x_8) и (x_8, x_{10}) стали насыщенными.

Таким образом общий поток $\varphi_n = \varphi_n + \delta^* = 39 + \delta^* = 43$.

В результате, после выполнения всех действий получаем граф:

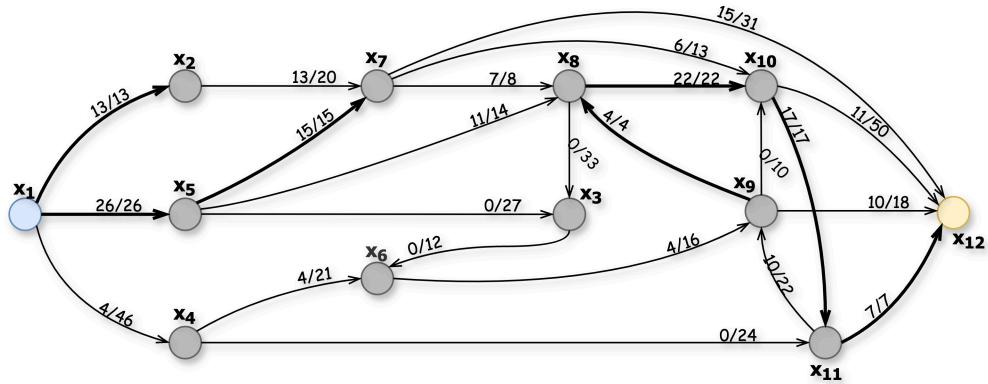


Рисунок 19 — Граф с текущими потоками

2.7. Путь $(x_1, x_4, x_6, x_9, x_{10}, x_{12})$

Рассмотрим путь $(x_1, x_4, x_6, x_9, x_{10}, x_{12})$.

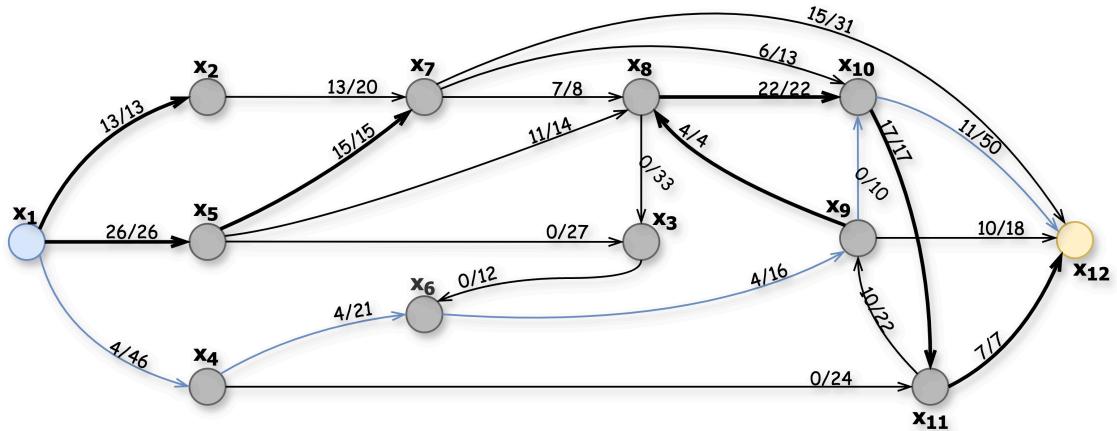


Рисунок 20 — Путь $(x_1, x_4, x_6, x_9, x_{10}, x_{12})$

Рассчитаем остаточную пропускную способность пути:

$$\begin{aligned}\delta^* &= \min\{ c(x_1, x_4) - \varphi(x_1, x_4), c(x_4, x_6) - \varphi(x_4, x_6), \\ &\quad c(x_6, x_9) - \varphi(x_6, x_9), c(x_9, x_{10}) - \varphi(x_9, x_{10}) \\ &\quad c(x_{10}, x_{12}) - \varphi(x_{10}, x_{12})\} \\ &= \min\{46 - 4, 21 - 4, 16 - 4, 10 - 0, 50 - 11\} = \\ &= \min\{42, 17, 12, 10, 39\} = 10\end{aligned}$$

$c(x_i, x_j)$ - пропускная способность ребра с вершинами x_i и x_j

$\varphi(x_i, x_j)$ - поток ребра с вершинами x_i и x_j

Добавим к потоку всех ребер, входящих в путь, остаточную пропускную способность пути. Ребра, пропускная способность которых будет равна потоку через них, станут насыщенными.

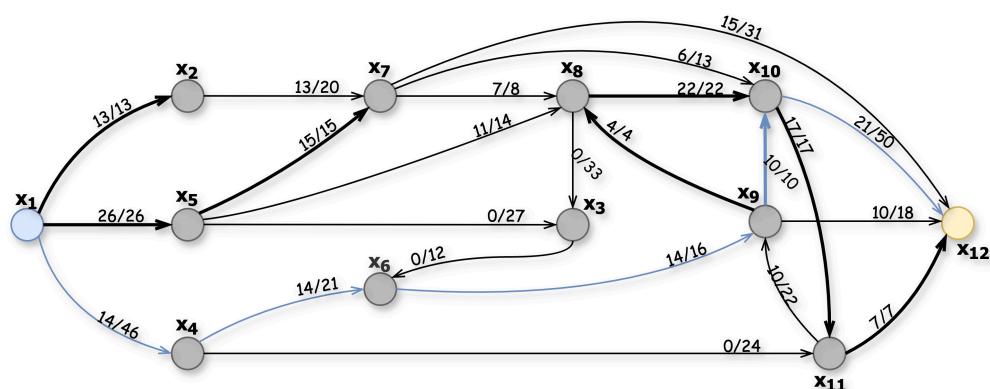


Рисунок 21 — Граф с добавленными потоками через ребра пути

В данном случае ребро (x_9, x_{10}) стало насыщенным.

Таким образом общий поток $\varphi_n = \varphi_n + \delta^* = 43 + \delta^* = 53$.

В результате, после выполнения всех действий получаем граф:

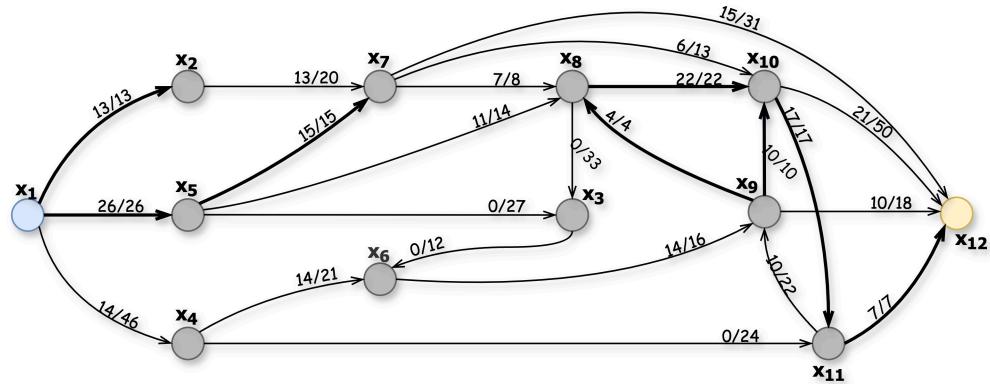


Рисунок 22 — Граф с текущими потоками

2.8. Путь $(x_1, x_4, x_6, x_9, x_{12})$

Рассмотрим путь $(x_1, x_4, x_6, x_9, x_{12})$.

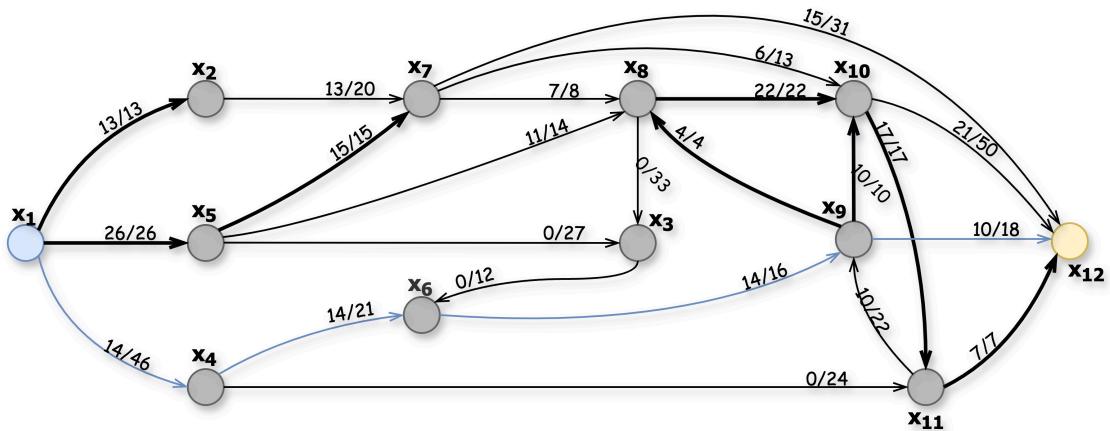


Рисунок 23 — Путь $(x_1, x_4, x_6, x_9, x_{12})$

Рассчитаем остаточную пропускную способность пути:

$$\begin{aligned}\delta^* &= \min\{ c(x_1, x_4) - \varphi(x_1, x_4), c(x_4, x_6) - \varphi(x_4, x_6), \\ &\quad c(x_6, x_9) - \varphi(x_6, x_9), c(x_9, x_{12}) - \varphi(x_9, x_{12})\} \\ &= \min\{46 - 14, 21 - 14, 16 - 14, 18 - 10\} = \\ &= \min\{32, 7, 2, 8\} = 2\end{aligned}$$

$c(x_i, x_j)$ - пропускная способность ребра с вершинами x_i и x_j

$\varphi(x_i, x_j)$ - поток ребра с вершинами x_i и x_j

Добавим к потоку всех ребер, входящих в путь, остаточную пропускную способность пути. Ребра, пропускная способность которых будет равна потоку через них, станут насыщенными.

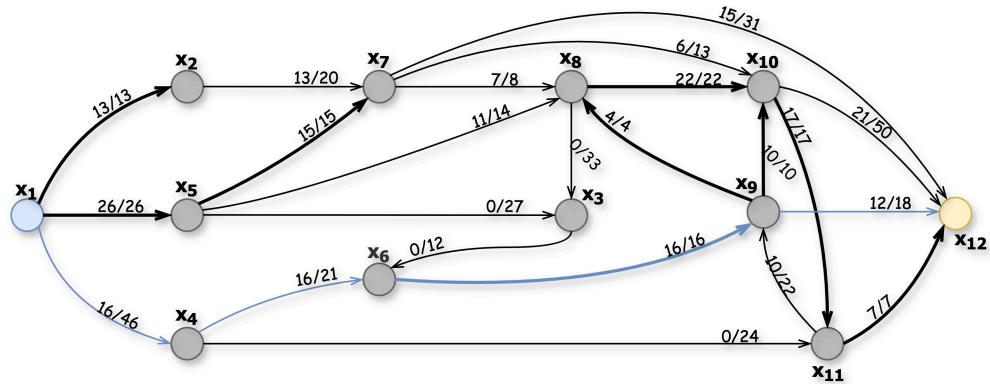


Рисунок 24 — Граф с добавленными потоками через ребра пути
В данном случае ребро (x_6, x_9) стало насыщенным.

Таким образом общий поток $\varphi_n = \varphi_n + \delta^* = 53 + \delta^* = 55$.

В результате, после выполнения всех действий получаем граф:

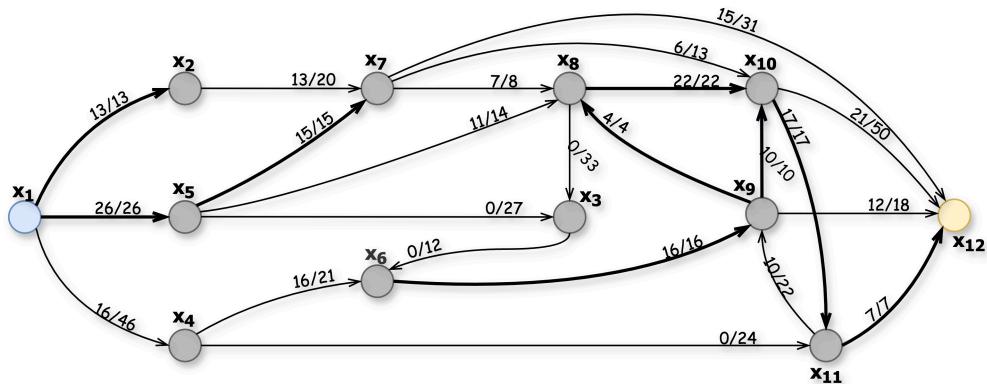


Рисунок 25 — Граф с текущими потоками

2.9. Путь $(x_1, x_4, x_{11}, x_9, x_{12})$

Рассмотрим путь $(x_1, x_4, x_{11}, x_9, x_{12})$.

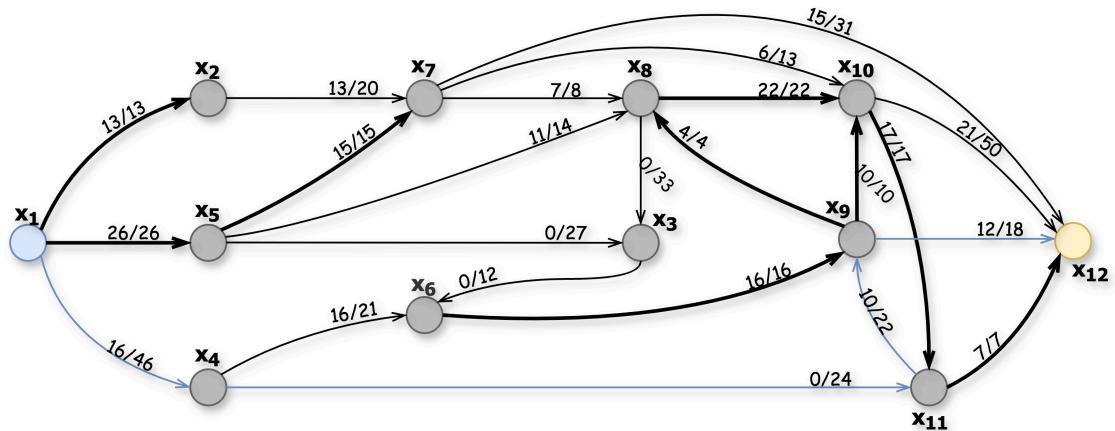


Рисунок 26 — Путь $(x_1, x_4, x_{11}, x_9, x_{12})$

Рассчитаем остаточную пропускную способность пути:

$$\begin{aligned}\delta^* &= \min\{ c(x_1, x_4) - \varphi(x_1, x_4), c(x_4, x_{11}) - \varphi(x_4, x_{11}), \\ &\quad c(x_{11}, x_9) - \varphi(x_{11}, x_9), c(x_9, x_{12}) - \varphi(x_9, x_{12})\} \\ &= \min\{46 - 16, 24 - 0, 22 - 10, 18 - 12\} = \\ &= \min\{30, 24, 12, 6\} = 6\end{aligned}$$

$c(x_i, x_j)$ - пропускная способность ребра с вершинами x_i и x_j

$\varphi(x_i, x_j)$ - поток ребра с вершинами x_i и x_j

Добавим к потоку всех ребер, входящих в путь, остаточную пропускную способность пути. Ребра, пропускная способность которых будет равна потоку через них, станут насыщенными.

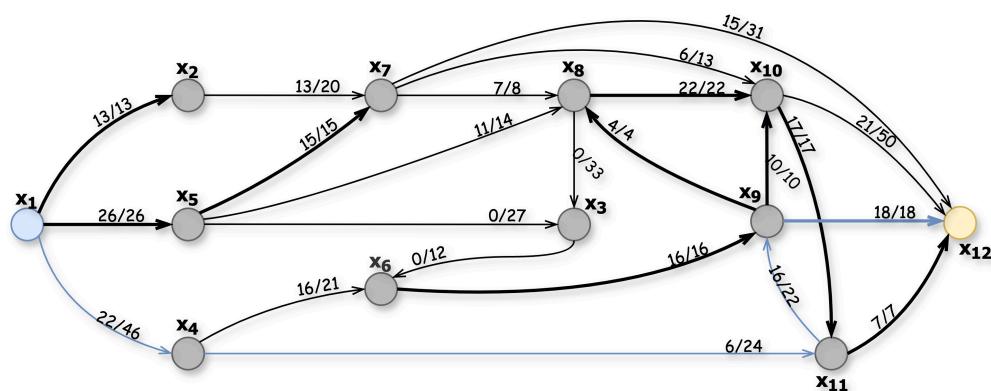


Рисунок 27 — Граф с добавленными потоками через ребра пути

В данном случае ребро (x_9, x_{12}) стало насыщенным.

Таким образом общий поток $\varphi_n = \varphi_n + \delta^* = 55 + \delta^* = 61$.

В результате, после выполнения всех действий получаем граф:

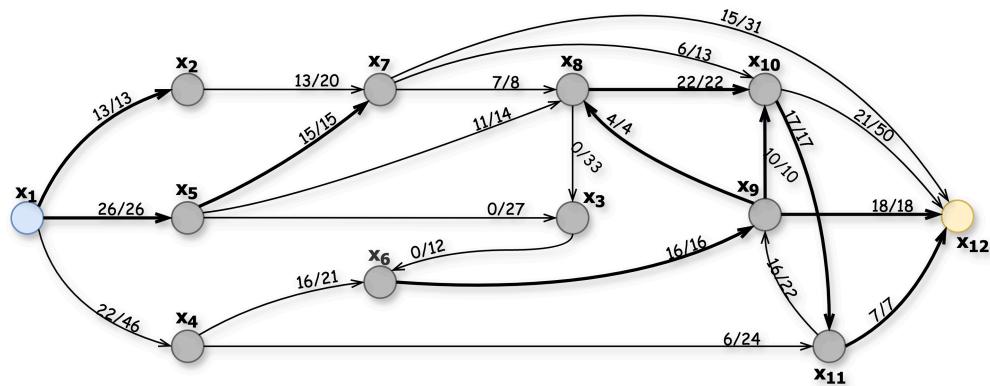


Рисунок 28 — Граф с текущими потоками

2.10. Почему больше нет путей от источника к стоку?

В процессе выполнения данного пункта мы строим пути, проходящие только через ненасыщенные ребра, таким образом из источника можно начать строить путь только через ребро (x_1, x_4) . Из вершины x_4 ведут 2 ребра. Данные ребра ненасыщены, поэтому можно продолжить построение пути по ним. Одно из ребер ведет в вершину x_6 , из которой исходит только одно ребро, но оно насыщено, поэтому по данному ребру из x_4 построить путь невозможно. Второе ребро ведет в вершину x_{11} , оно ненасыщено. Из x_{11} исходят 2 ребра, но одно из них насыщено, поэтому из вершины x_{11} путь может проходить только в вершину x_9 . Из данной вершины исходят только насыщенные ребра, поэтому дальнейшее построение пути до стока невозможно.

2.11. Полученный график и полный поток

После применения теоремы 1 получили график с насыщенными ребрами (выделены цветом на рис. 29).

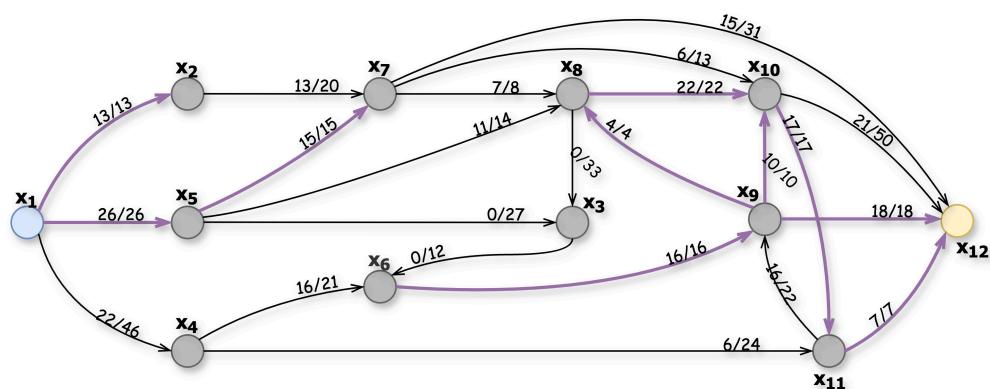


Рисунок 29 — Граф с текущими потоками

Полный поток получился равным $\varphi_n = 61$.

2.12. Проверки

Посчитаем сумму потоков ребер, исходящих из вершины x_1 .

$$\varphi_{x_1} = \varphi(x_1, x_2) + \varphi(x_1, x_4) + \varphi(x_1, x_5) = 13 + 26 + 22 = 61 = \varphi_n$$

Посчитаем сумму потоков ребер, приходящих в вершину x_{12} .

$$\begin{aligned}\varphi_{x_{12}} &= \varphi(x_7, x_{12}) + \varphi(x_9, x_{12}) + \varphi(x_{10}, x_{12}) + \varphi(x_{11}, x_{12}) = \\ &= 15 + 21 + 18 + 7 = 61 = \varphi_{x_1} = \varphi_n\end{aligned}$$

Значения исходящего из истока, входящего в сток и общего потока совпали.

Построим матрицу потоков ребер.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}
x_1		13		22	26							
x_2							13					
x_3						0						
x_4							16				6	
x_5			0				15	11				
x_6									16			
x_7								7		6		15
x_8			0							22		
x_9								4		10		18
x_{10}											17	21
x_{11}									16			7
x_{12}												

По данной таблице можно посчитать сумму входящего потока и исходящего для любой вершины. Эти потоки будут равны (за исключением вершин источника и стока). Для подсчета входящего потока вершины x_n суммируем

все значения столбца x_n таблицы, для подсчета исходящего потока суммируем все значения строки.

Вершина	Входящий поток	Исходящий поток
x_1		61
x_2	13	13
x_3	0	0
x_4	22	22
x_5	26	26
x_6	16	16
x_7	28	28
x_8	22	22
x_9	32	32
x_{10}	38	38
x_{11}	23	23
x_{12}	61	

3. Оптимизация

Итерационно используя теорему 2, увеличиваем полный поток.

3.1. Увеличивающий маршрут ($x_1, x_4, x_{11}, x_{10}, x_{12}$)

Выполним разметку вершин, необходимо пометить вершину стока. Для начала пометим знаком + источник x_1 . Каждую следующую вершину помечаем номером предыдущей со знаком + или -:

- Со знаком +, если ребро направлено из предыдущей вершины (последней помеченной) в помечаемую и оно строго ненасыщено (в данном случае ребро называется прямым),
- Со знаком -, если ребро противоположно направлено и его поток не равен 0 (само ребро называется обратным).

Таким образом получилось отметить 5 вершин

Вершина	Пометка
x_1	+
x_4	+1
x_{11}	+4
x_{10}	-11
x_{12}	+10

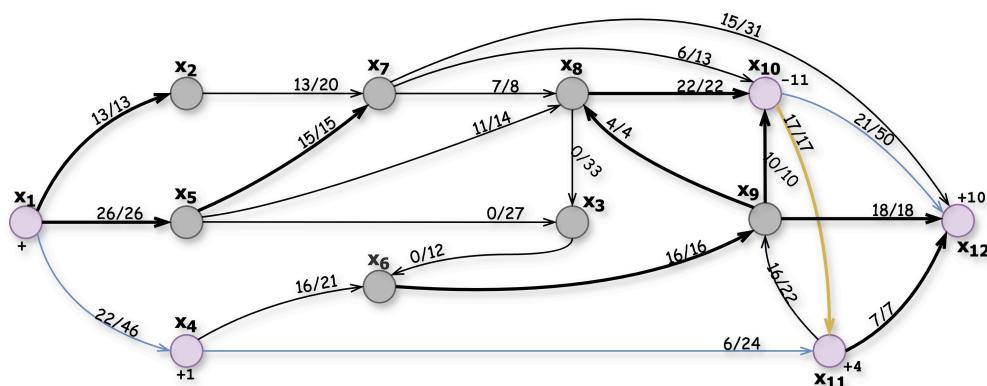


Рисунок 30 — Граф с метками

Ребра, которые направлены на вершины с метками + (прямые ребра) обозначим синим, а те, которые противоположно направлены на вершины с метками - (обратные ребра) обозначим оранжевым.

Рассчитаем остаточную пропускную способность синих (прямых) ребер.

$$\delta^* = \min\{ c(x_1, x_4) - \varphi(x_1, x_4), c(x_4, x_{11}) - \varphi(x_4, x_{11}), c(x_{10}, x_{12}) - \varphi(x_{10}, x_{12}) \} = \\ = \min\{46 - 22, 24 - 6, 50 - 21\} = \min\{24, 18, 29\} = 18$$

$c(x_i, x_j)$ - пропускная способность ребра с вершинами x_i и x_j

$\varphi(x_i, x_j)$ - поток ребра с вершинами x_i и x_j

Найдем минимальный поток оранжевых (обратных) ребер.

$$\varphi^* = \min\{\varphi(10, 11)\} = \min\{17\} = 17$$

$$\varepsilon^* = \min\{\delta^*, \varphi^*\} = \min\{18, 17\} = 17$$

Поток в сети можно увеличить на 17 единиц ($\varphi_{\max} = \varphi_n + 17 = 78$) за счет данного увеличивающего маршрута. Увеличим поток на синих (прямых) ребрах на ε^* и уменьшим на оранжевых (обратных) на то же значение.

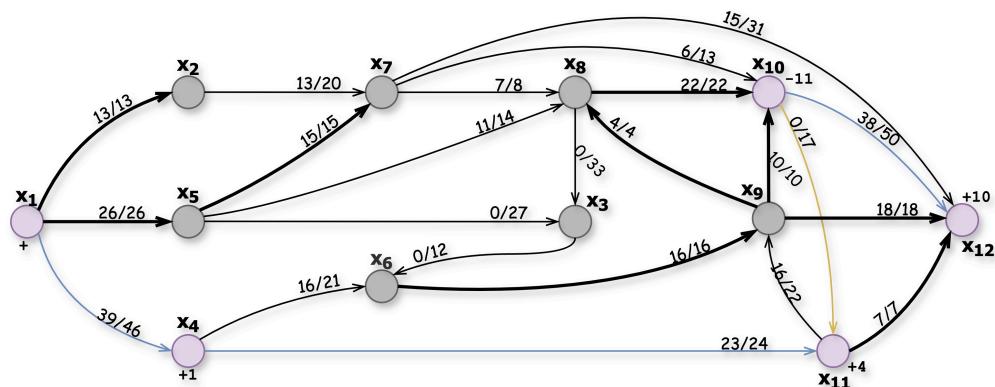


Рисунок 31 — Граф с оптимизированным распределением потоков

Так как на оранжевом (обратном) ребре поток уменьшился можно сделать вывод, что ребро (x_{10}, x_{11}) перестало быть насыщенным.

В результате, после выполнения всех действий получаем граф:

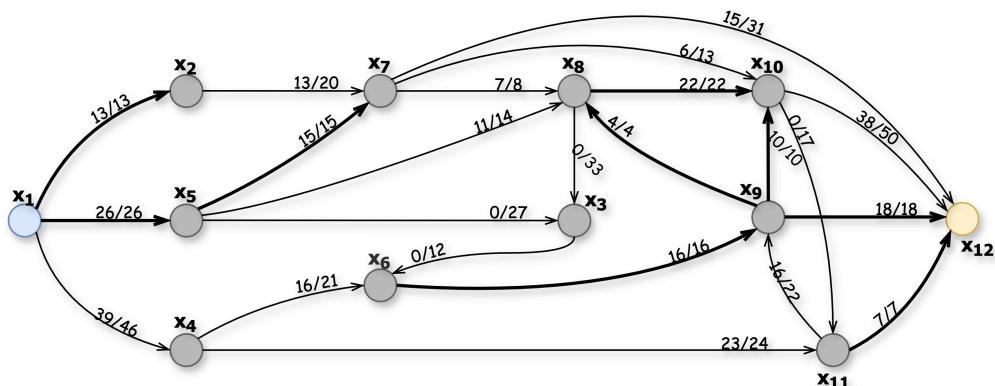


Рисунок 32 — Граф с текущими потоками

3.2. Почему больше нет увеличивающих маршрутов?

Попробуем пометить еще один увеличивающий маршрут. Пометим x_1 знаком +. Из x_1 можно попасть только в вершину x_4 , так как x_1 не является концом ребра, ни одну вершину мы не можем пометить знаком -, а знаком + можем пометить, только ту, ребро которой не является насыщенным.

Таким образом помечаем x_4 как +1. Вершина x_4 является концом ребра, начало которого в x_1 , таким образом вершина x_1 может быть помечена как -4, но так как из x_1 можно пометить только x_4 помечать x_1 второй раз не целесообразно.

Вершина x_4 является началом 2 ненасыщенных ребер, это означает, что мы можем пометить одну из двух вершин меткой +4. Если мы пометим вершину x_6 , то будет невозможно пометить x_9 , так как ребро насыщено, и x_3 , так как поток, проходящий через ребро равен 0. Таким образом из вершины x_6 можно пометить только вершину $-x_4$.

Попробуем вернуться на пару шагов назад и пометить x_{11} вместо x_6 . Так как больше ничего пометить мы не можем (остальные маршруты не могут привести к стоку) далее будем рассматривать все относительно вершины x_{11} . Помечать x_4 нет смысла. Поэтому дальше пометить мы можем только x_9 .

Меткой -9 можно пометить x_{11} , но помечать данную вершину нет смысла. Меткой +9 мы не можем пометить ни одну из вершин, так как все ребра, началом которых является x_9 , насыщены. Остается рассмотреть только возможность пометки вершины x_6 меткой -9: из нее мы можем попасть только в вершину x_4 и далее вести рассуждения так же, как и во втором абзаце.

Таким образом любые попытки пометить сток приводят нас к циклу, в котором мы помечаем одни и те же вершины разными метками, но при этом сток пометить не удается, поэтому можно утверждать, что больше нет увеличивающих маршрутов.

Поэтому по теореме 3 можно утверждать, что полный поток равен максимальному.

3.3. Полученный граф и общий поток

Получили граф с максимальным общим потоком.

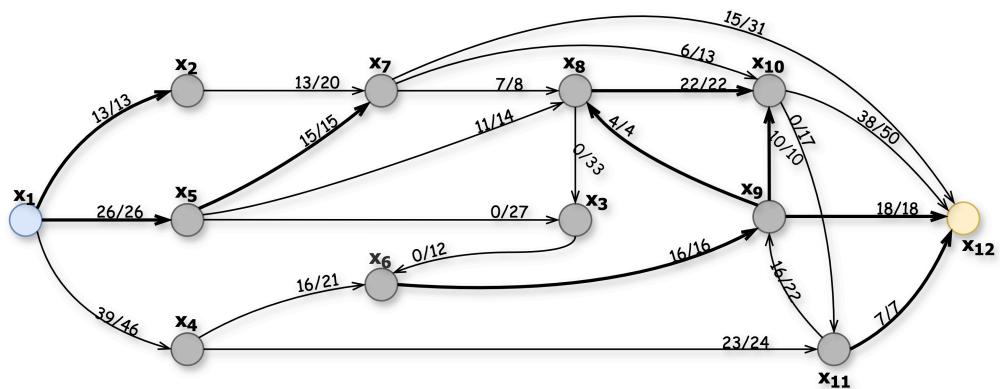


Рисунок 33 — Граф с максимальным общим потоком

Максимальный общий поток получился равным $\varphi_{\max} = 78$.

3.4. Проверки

Посчитаем сумму потоков ребер, исходящих из вершины x_1 .

$$\varphi_{x_1} = \varphi(x_1, x_2) + \varphi(x_1, x_4) + \varphi(x_1, x_5) = 13 + 26 + 38 = 78 = \varphi_{\max}$$

Посчитаем сумму потоков ребер, приходящих в вершину x_{12} .

$$\begin{aligned}\varphi_{x_{12}} &= \varphi(x_7, x_{12}) + \varphi(x_9, x_{12}) + \varphi(x_{10}, x_{12}) + \varphi(x_{11}, x_{12}) = \\ &= 15 + 38 + 18 + 7 = 78 = \varphi_{x_1} = \varphi_{\max}\end{aligned}$$

Значения исходящего из истока, входящего в сток и общего потока совпадали.

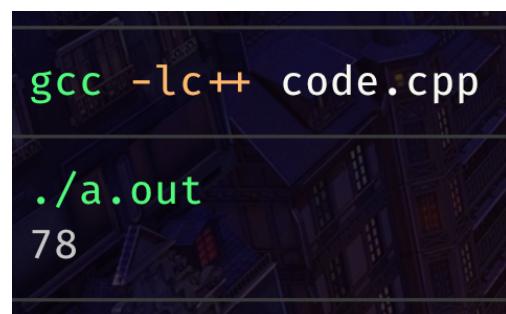
Построим матрицу потоков ребер.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}
x_1		13		39	26							
x_2							13					
x_3						0						
x_4						16					23	
x_5			0				15	11				
x_6									16			
x_7								7		6		15
x_8			0							22		
x_9								4		10		18
x_{10}										0	38	
x_{11}									16			7
x_{12}												

По данной таблице можно посчитать сумму входящего потока и исходящего для любой вершины. Эти потоки будут равны (за исключением вершин источника и стока). Для подсчета входящего потока вершины x_n суммируем все значения столбца x_n таблицы, для подсчета исходящего потока суммируем все значения строки.

Вершина	Входящий поток	Исходящий поток
x_1		78
x_2	13	13
x_3	0	0
x_4	39	39
x_5	26	26
x_6	16	16
x_7	28	28
x_8	22	22
x_9	32	32
x_{10}	38	38
x_{11}	23	23
x_{12}	78	

Напишем код, который считает максимальный поток по входной матрице пропускных способностей. Результатом работы кода тоже является число 78.



```
gcc -lc++ code.cpp
./a.out
78
```

Рисунок 34 — Вывод кода

```

#include <iostream>
#include <queue>

bool search(int se, int t, int graph[12][12], int* par){
    bool in[12] = {0,0,0,0,0,0,0,0,0,0,0,0};
    std::queue<int> que;
    que.push(se);
    in[se] = true;
    while (!que.empty()){
        int last = que.front();
        que.pop();
        for (int j = 0; j < 12; j++){
            if (!in[j] && graph[last][j] > 0){
                que.push(j);
                in[j] = 1;
                par[j] = last;
            }
        }
    }
    return in[t];
}

int algo(int matrix[12][12]){
    int par[12] = {-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1};
    int flowMax = 0;
    while (search(0, 11, matrix, par)){
        int flow = 1000;
        int t = 11;
        while (t != 0){
            flow = flow > matrix[par[t]][t] ? matrix[par[t]][t]: flow;
            t = par[t];
        }
        flowMax += flow;
        t = 11;
        while (t != 0){
            int pt = par[t];
            matrix[pt][t] -= flow;
            matrix[t][pt] += flow;
            t = pt;
        }
    }
    return flowMax;
}

int main(){
    int matrix[12][12] = {
        {0, 13, 0, 46, 26, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 20, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 12, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 21, 0, 0, 0, 0, 24, 0}, {0, 0, 27, 0, 0, 0, 15, 14, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0, 16, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 8, 0, 13, 0, 31}, {0, 0, 33, 0, 0, 0, 0, 0, 0, 22, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 4, 0, 10, 0, 18}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 17, 50}, {0, 0, 0, 0, 0, 0, 0, 22, 0, 0, 0, 7}, {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}};

    std::cout << algo(matrix) << "\n";
}

```

4. Построение минимального разреза

Для любой сети с одним источником и одним стоком величина максимального потока, доставляемая от источника к стоку равна пропускной способности минимального разреза.

Пусть подмножество вершин, оставшихся непомеченными при попытке построить увеличивающий маршрут будет $A = \{x_2, x_3, x_5, x_7, x_8, x_{10}, x_{12}\}$, тогда

$$\varphi_t = \sum \varphi(U_i) - \sum \varphi(U_j)$$

$U_i \in U_A^+$ – подмножество ребер, заходящих в вершины подмножества А

$U_j \in U_A^-$ – подмножество ребер, исходящих из вершин подмножества А

В случае максимального потока $\sum \varphi(U_j) = 0$, тогда:

$$\varphi_t = \sum \varphi(U_i)$$

Все ребра из U_A^+ являются насыщенными и $\sum \varphi(U_i) = \varphi_{\max}$ сумма пропускных способностей дает максимальный поток. Именно эти ребра образуют минимальный разрез.

Насыщенные ребра имеют свойства: начальная вершина $\in X$ ($X = \{x_1, x_4, x_6, x_9, x_{11}\}$ – вершины, которые удалось пометить), конечная в подмножестве А. Таким образом получаем, что разрез будет проходить между теми вершинами, которые удалось пометить и теми, которые пометить не удалось. Получаем разрез ($X \rightarrow A$):

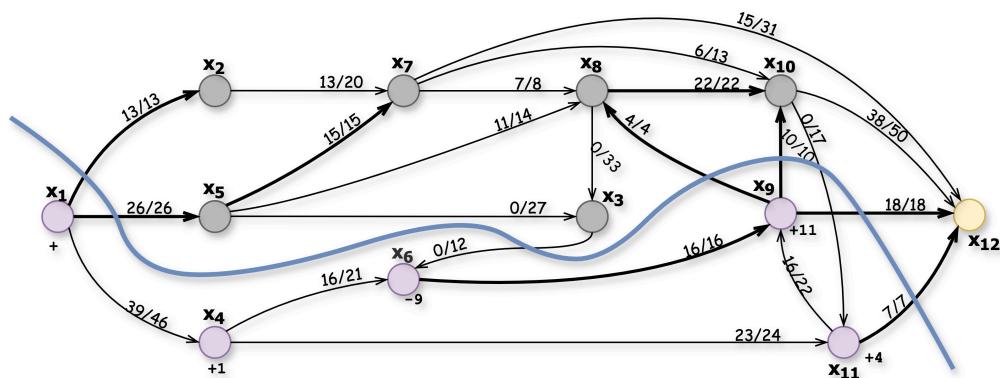


Рисунок 35 — Разрез ($X \rightarrow A$)

Минимальный разрез в данной сети составляют ребра

$$(X \rightarrow A) = \{(x_1, x_2), (x_1, x_5), (x_9, x_8), (x_9, x_{10}), (x_9, x_{12}), (x_{11}, x_{12})\}$$

Рассчитаем пропускную способность минимального разреза

$$C(X \rightarrow A) = C(x_1, x_2) + C(x_1, x_5) + C(x_9, x_8) + C(x_9, x_{10}) + \\ + C(x_9, x_{12}) + C(x_{11}, x_{12}) = 13 + 26 + 4 + 10 + 18 + 7 = 78$$

Пропускная способность минимального разреза совпала с максимальным потоком сети.

5. Вывод

- Для определения полного потока было исследовано 9 путей, был достигнут полный поток $\varphi_n = 61$. Был проверен баланс.
- В оптимизационной части алгоритма был построен 1 увеличивающий маршрут, который увеличил полный поток, полученный в предыдущем пункте на 17 единиц, полный поток стал равен максимальному. Был проверен баланс и написан код по поиску максимального потока в сети, вывод программы совпал с максимальным потоком $\varphi_{\max} = 78$.
- В минимальном разрезе 6 ребер оказались насыщенными, пропускная способность минимального разреза совпала с максимальным потоком. Условие теоремы Форда-Фалкерсона не нарушено, максимальный поток найден верно.

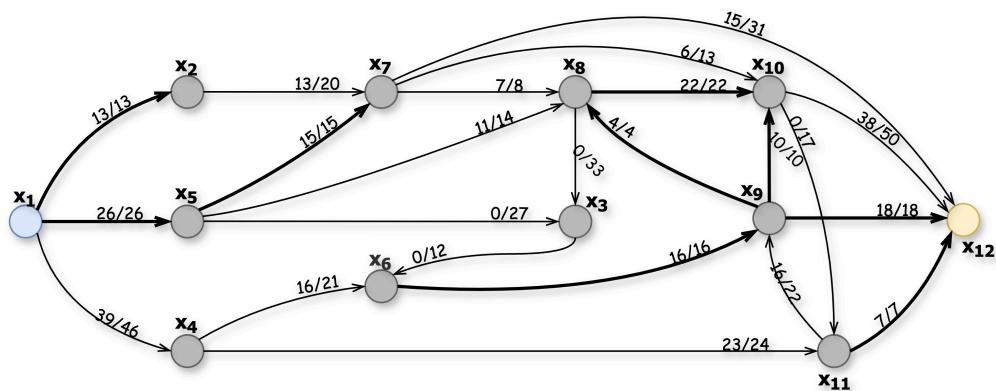


Рисунок 36 — Получившийся график