

РЕШЕНИЙ

ПРИНЯТИЕ ОПТИМАЛЬНЫХ РЕШЕНИЙ МЕТОДАМИ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

СОДЕРЖАНИЕ

- Текущий контроль знаний
- Часть 1. Общие принципы динамического программирования.
- Часть 2. Принятие решений на моделях, сводимых к задачам дискретной оптимизации с булевыми переменными.
- Часть 3. Принятие решений на моделях, сводимых к задачам дискретной оптимизации с небулевыми переменными.
- Часть 4. Принятие решений на моделях оптимального упорядочения.

ТЕКУЩИЙ КОНТРОЛЬ ЗНАНИЙ

- На бихроматическом графе $G(X,U)$, $|X| = 8$, $|X_1| = |X_2| = 4$, матрица которого приведена ниже, определить оптимальное распределение работ при условии, что:
 - 1. Минимизируется время выполнения плана, при условии, что фонд зарплаты равен:

$$S = 4 \cdot \max\{|k-5|; |k-25|\}.$$
 - 2. Минимизируются затраты на выполнение плана при условии, что время его выполнения не превышает величины $T = \max\{|k-15|; |k-35|\}.$
 - 3. Минимизируются затраты на выполнение плана при условии, что время его выполнения не ограничено.

	1	2	3	4
1	$ k-5 ; k-15 $	$ k-10 ; k-25 $	$ k-7 ; k-17 $	$ k-8 ; k-15 $
2	$ k-3 ; k-30 $	$ k-25 ; k-15 $	$ k-2 ; k-19 $	$ k-4 ; k-32 $
3	$ k-31 ; k-15 $	$ k-6 ; k-14 $	$ k-3 ; k-35 $	$ k-23 ; k-25 $
4	$ k-16 ; k-14 $	$ k-35 ; k-5 $	$ k-11 ; k-10 $	$ k-5 ; k-15 $

ЧАСТЬ 1

Общие принципы динамического программирования

ОПРЕДЕЛЕНИЕ

Динамическое программирование представляет собой многошаговый процесс принятия решений, направленных на достижение единой цели. При этом на каждом шаге этого процесса решается задача меньшей размерности, чем исходная.

Принцип оптимальности Беллмана

- **Оптимальная стратегия обладает тем свойством, что независимо от начального состояния и начального решения задачи, последующие решения должны составлять оптимальную стратегию лишь в рассматриваемый момент времени. Иными словами оптимальная стратегия в каждый момент времени определяется лишь состоянием системы, но не ее предысторией.**

Часть 2

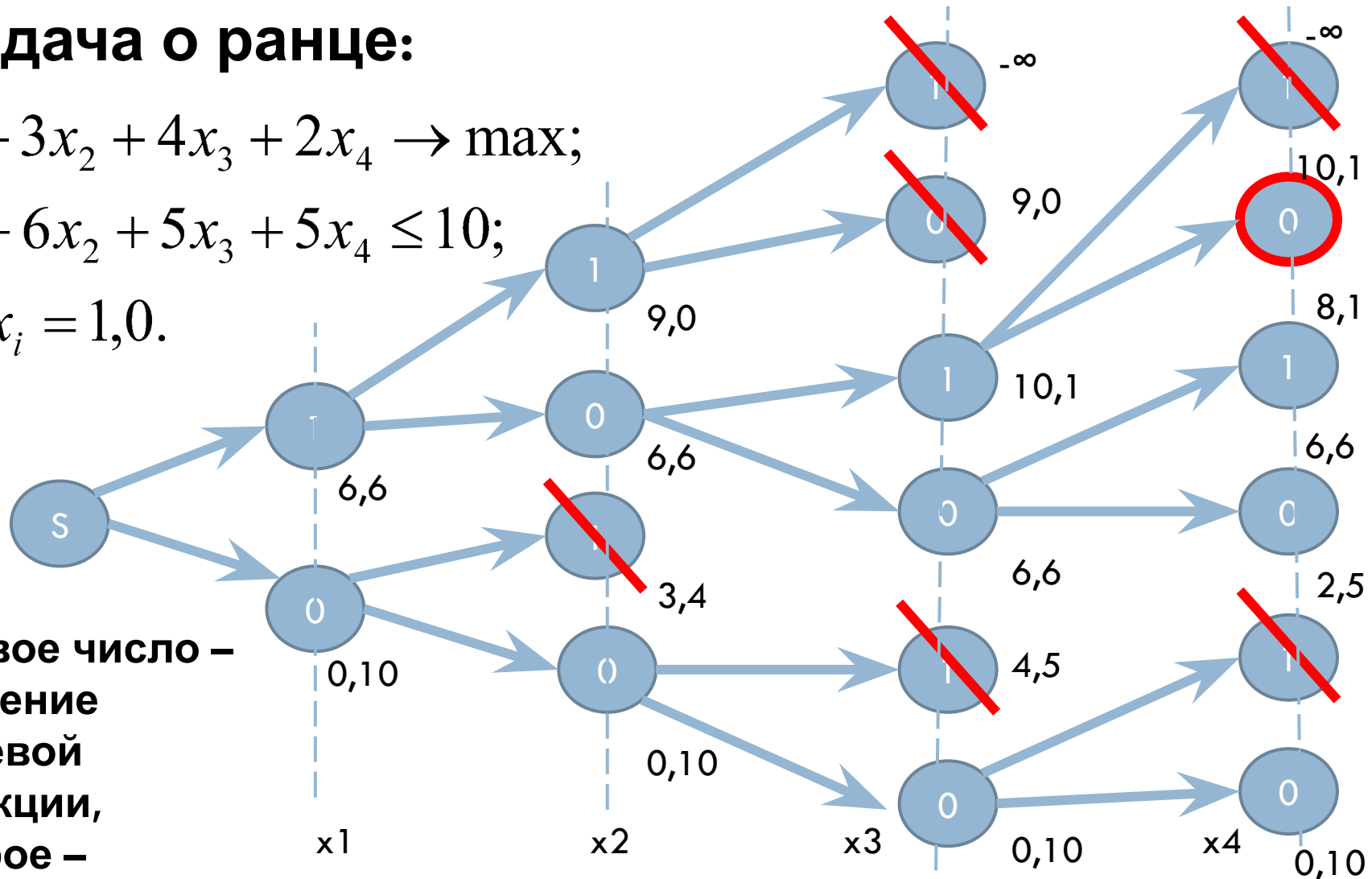
**Принятие решений на
моделях, сводимых к
задачам дискретной
оптимизации с
булевыми переменными**

ПРИМЕР 1: Решение задач с булевыми переменными

Задача о ранце:

$$\begin{cases} 6x_1 + 3x_2 + 4x_3 + 2x_4 \rightarrow \max; \\ 4x_1 + 6x_2 + 5x_3 + 5x_4 \leq 10; \\ \forall i: x_i = 1, 0. \end{cases}$$

Первое число –
значение
целевой
функции,
второе –



САМОСТОЯТЕЛЬНО

- Пользуясь методом динамического программирования, решить задачу о ранце:

$$\begin{cases} 4x_1 + 7x_2 + 9x_3 + 3x_4 \rightarrow \max; \\ 6x_1 + 5x_2 + 8x_3 + 7x_4 \leq 12; \\ \forall i : x_i = 1, 0. \end{cases}$$

ЧАСТЬ 3

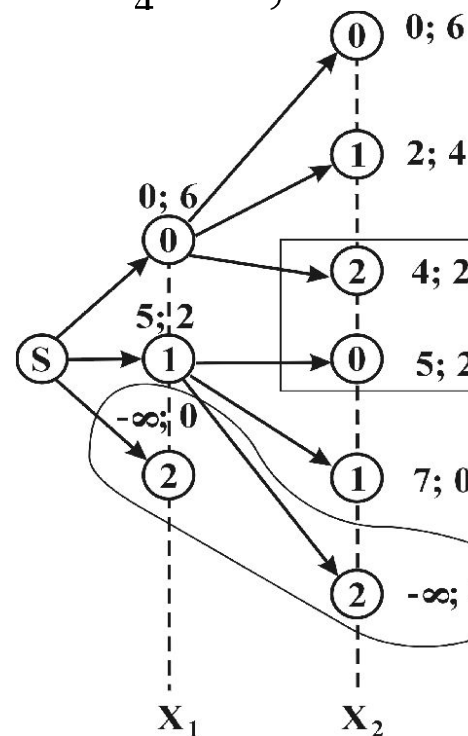
**Принятие решений на
моделях, сводимых к
задачам дискретной
оптимизации с
небулевыми
переменными**

ПРИМЕР 2: Решение задачи с небулевыми переменными

Решение задачи вида:

$$\begin{cases} 5x_1 + 2x_2 + 9x_3 + 7x_4 \rightarrow \max; \\ 4x_1 + 2x_2 + 2x_3 + 4x_4 \leq 6; \\ \forall i, x_i \in \{0, 1, 2\}. \end{cases}$$

Первые две



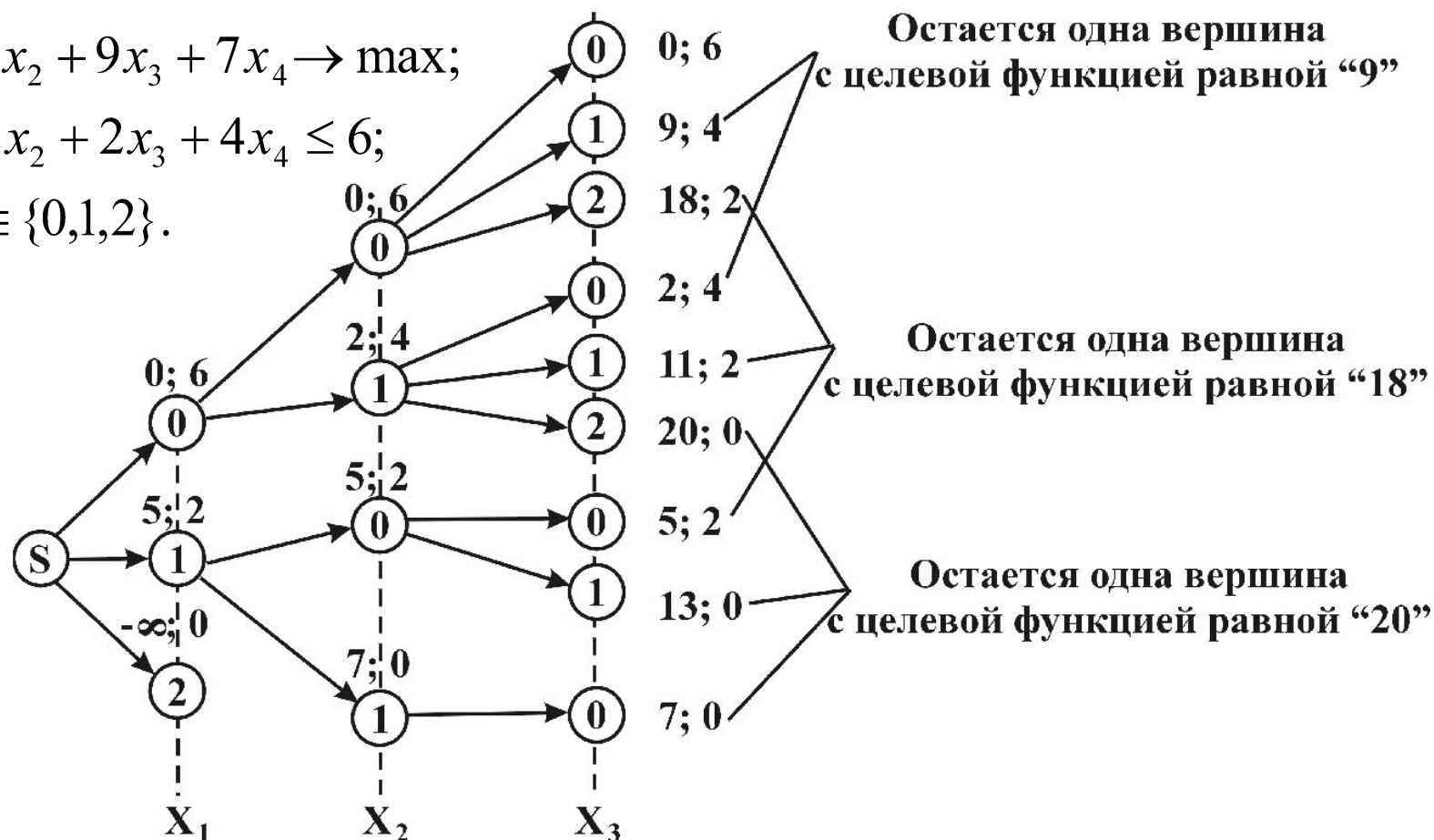
т.к. значения ресурса те же, из этих планов остается лишь один, тот, у которого значение целевой функции больше.

Недопустимые планы

ПРИМЕР 2 (ПРОДОЛЖЕНИЕ)

Третья итерация:

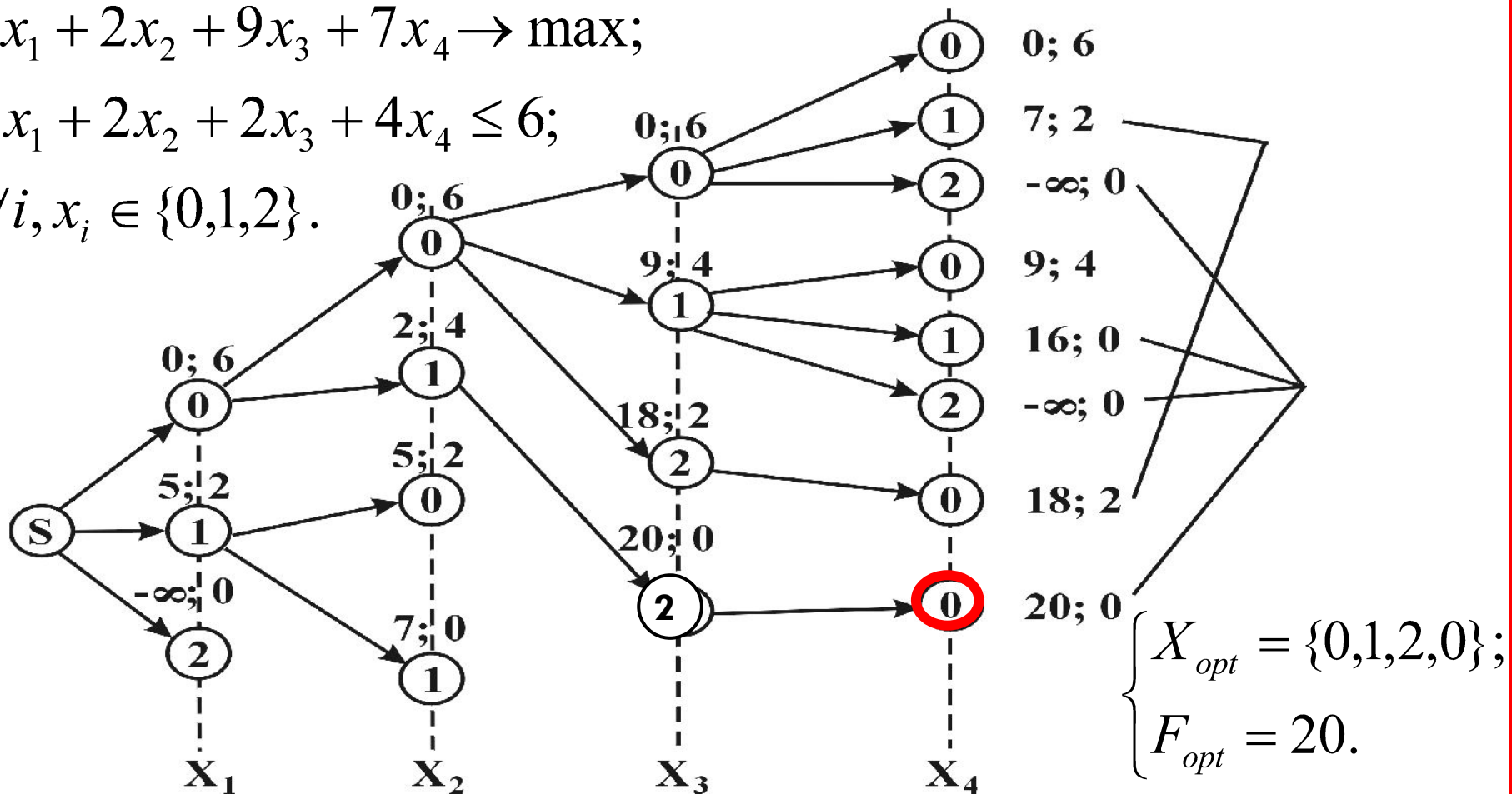
$$\begin{cases} 5x_1 + 2x_2 + 9x_3 + 7x_4 \rightarrow \max; \\ 4x_1 + 2x_2 + 2x_3 + 4x_4 \leq 6; \\ \forall i, x_i \in \{0,1,2\}. \end{cases}$$



Пример 2 (завершение)

Четвертая итерация:

$$\begin{cases} 5x_1 + 2x_2 + 9x_3 + 7x_4 \rightarrow \max; \\ 4x_1 + 2x_2 + 2x_3 + 4x_4 \leq 6; \\ \forall i, x_i \in \{0,1,2\}. \end{cases}$$



САМОСТОЯТЕЛЬНО:

**Решить задачу с небулевыми
и с булевыми переменными**

вида:

$$\begin{cases} 7x_1 + 2x_2 + 4x_3 + 5x_4 \rightarrow \max; \\ 4x_1 + 2x_2 + 2x_3 + 4x_4 \leq 8; \\ \forall i < 3, x_i \in \{0,1,2\}, \\ \forall 3 \leq i \leq 4 : x_i \in \{0,1\}. \end{cases}$$

Часть 4

Принятие решений на моделях оптимального упорядочения

ПРИМЕР 3: ЗАДАЧА КОММИВОЯЖЕРА

- Решить, пользуясь методом динамического программирования, разомкнутую задачу коммивояжера, условия которой отвечают графу $G(X, U)$, изображенному на рисунке ниже.

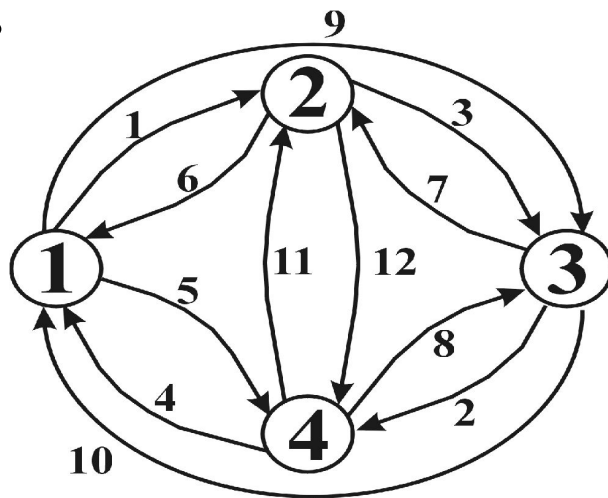


Рис. 1.
Граф $G(X, U)$

$$\left\{ \begin{array}{l} \sum_i \sum_j r(i, j) z(i, j) \rightarrow \min; \\ \exists x_s \in X : \sum_i z(i, s) = 0; \sum_j z(s, j) = 1; \\ \exists x_t \in X : \sum_i z(i, t) = 1; \sum_j z(t, j) = 0; \\ \forall x_k \in X \setminus (x_s \boxtimes x_t) : \sum_i z(i, k) = \sum_j z(k, j) = 1; \\ \forall (i, j) \in U : z(i, j) = 1, 0. \end{array} \right.$$

ПРИМЕР 3. ХОД РЕШЕНИЯ

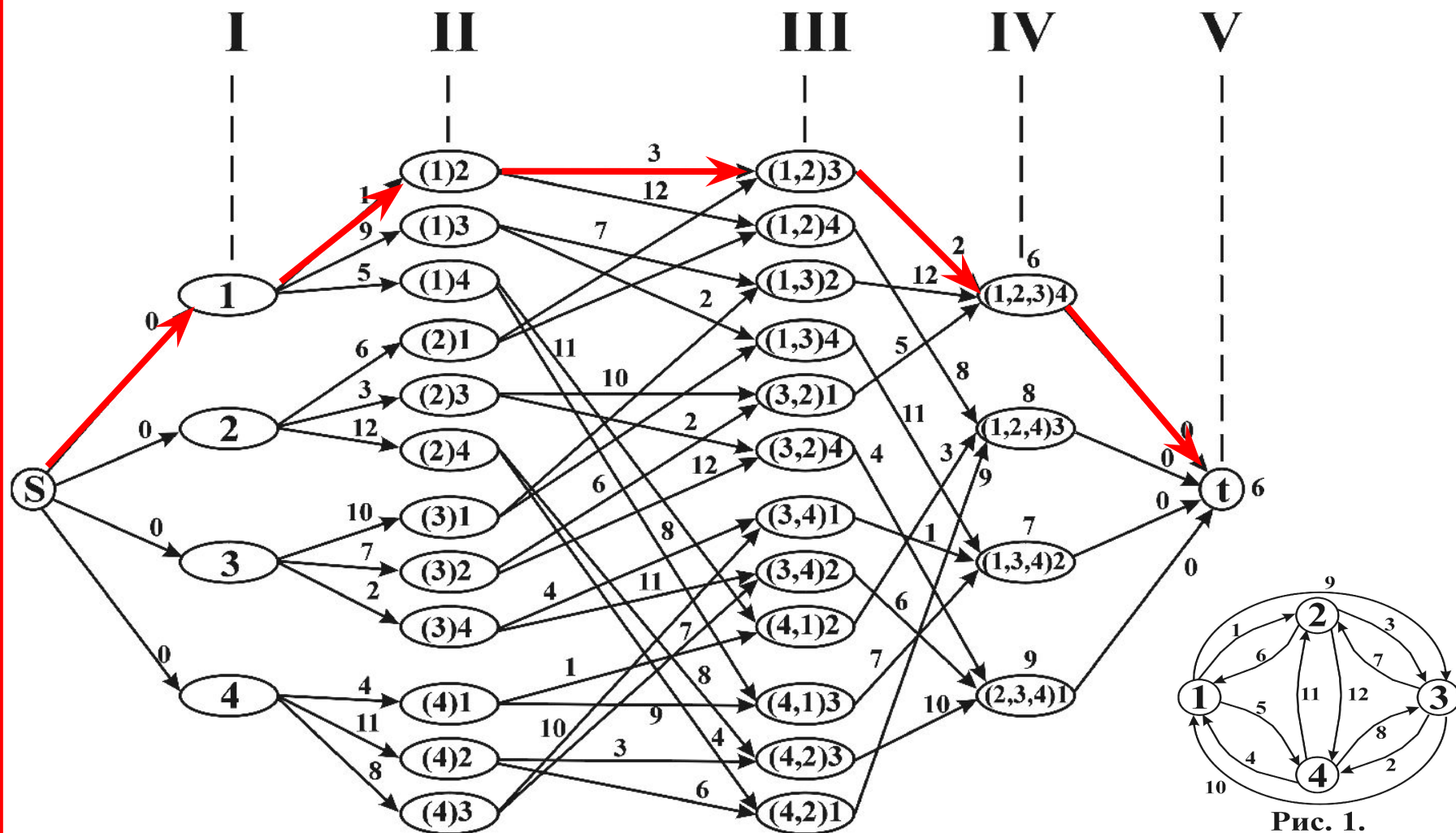


Рис. 1.
Граф $G(X,U)$

Самостоятельно вывести:

Формулы, определяющие:

- 1. Число вершин каждого слоя построенной сети.**
- 2. Число дуг, заходящих в каждую вершину i -го слоя.**
- 3. Число дуг, исходящих из каждой вершины i -го слоя.**

САМОСТОЯТЕЛЬНО:

- Решить разомкнутую задачу коммивояжера на графе $G(X, U)$, изображенном ниже:

