



Лекция 6. **Динамическое программирование**

Содержание лекции:

1. Формулировка задачи динамического программирования
2. Принцип оптимальности Беллмана
3. Алгоритм решения задач динамического программирования
4. Экономические приложения задач динамического программирования



Литература

- *Фомин Г.П.* Математические методы и модели в коммерческой деятельности: Учебник. – 2-е изд. М.: Финансы и статистика, 2005. — Глава 5.
- *Экономико-математические методы и прикладные модели: Учеб. пособие для вузов / Под ред. В.В. Федосеева.* — 2-е изд. М.: ЮНИТИ-ДАНА, 2005. — Раздел 3.5.



6.1. Формулировка задачи динамического программирования

■ Дано:

- ◆ множество состояний
 - ◆ в том числе *начальное* и *конечное*
- ◆ множество возможных переходов из одного состояния в другое
 - ◆ с каждым переходом связывается числовой параметр
 - интерпретируется как затраты, выгода, расстояние, время и т. п.

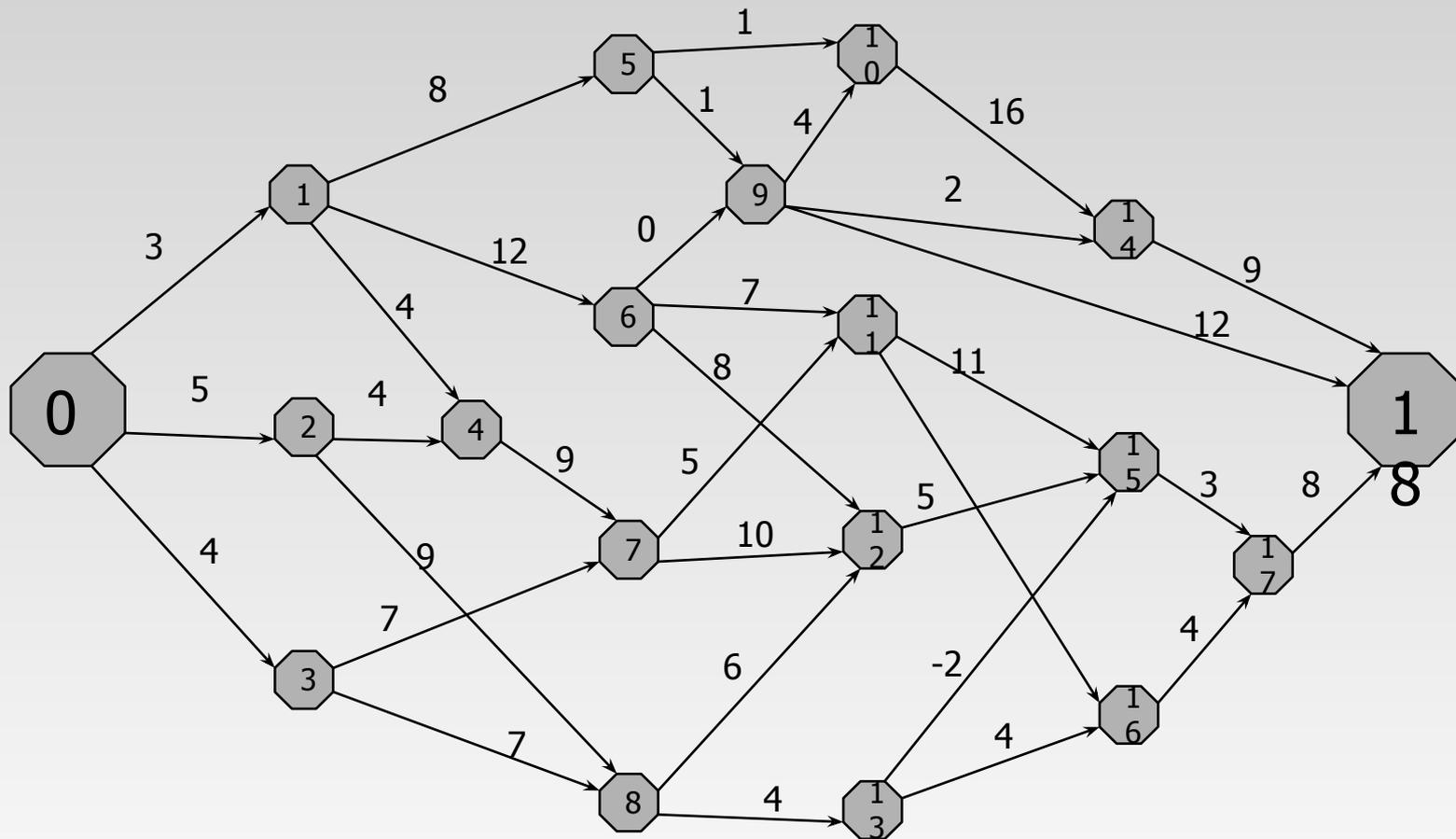
■ Найти:

- ◆ оптимальную последовательность переходов (*путь*) из начального состояния в конечное
 - ◆ максимум или минимум суммы числовых параметров
 - ◆ предполагается, что хотя бы один путь из начального состояния в конечное существует



6.1

Пример





сумма числовых значений (e.g. расстояний) по всему пути

Математическая запись

6.1

$$\max_{x_{ij}} \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij}$$

$$\sum_{i \in I} x_{ij} \leq 1, j \in J$$

$$\sum_{i \in I} x_{ik} = \sum_{j \in J} x_{kj}, k \in I \setminus \{0\}$$

$$x_{ij} \in \{0;1\}, i \in I, j \in J$$

$$x_{ij} = 0, i \in I, j \in J, (i, j) \notin A$$

единственность искомого пути: в каждую вершину можно прийти только из одной вершины (или вообще нельзя)

если искомый путь пришёл в вершину k , то он должен из неё выйти (если только она не конечная)

1, если путь проходит через дугу (i,j)
0, если не проходит или такой дуги нет

Например, расстояние между пунктами i и j , км

Условие целочисленности переменных

Между вершинами i и j нет дуги.

x_{ij} переменная включения дуги (i, j) в путь

c_{ij} числовое значение, приписанное дуге

I множество вершин, кроме конечной

J множество вершин, кроме начальной вершины 0

A множество всех дуг

\setminus оператор исключения множества из множества X

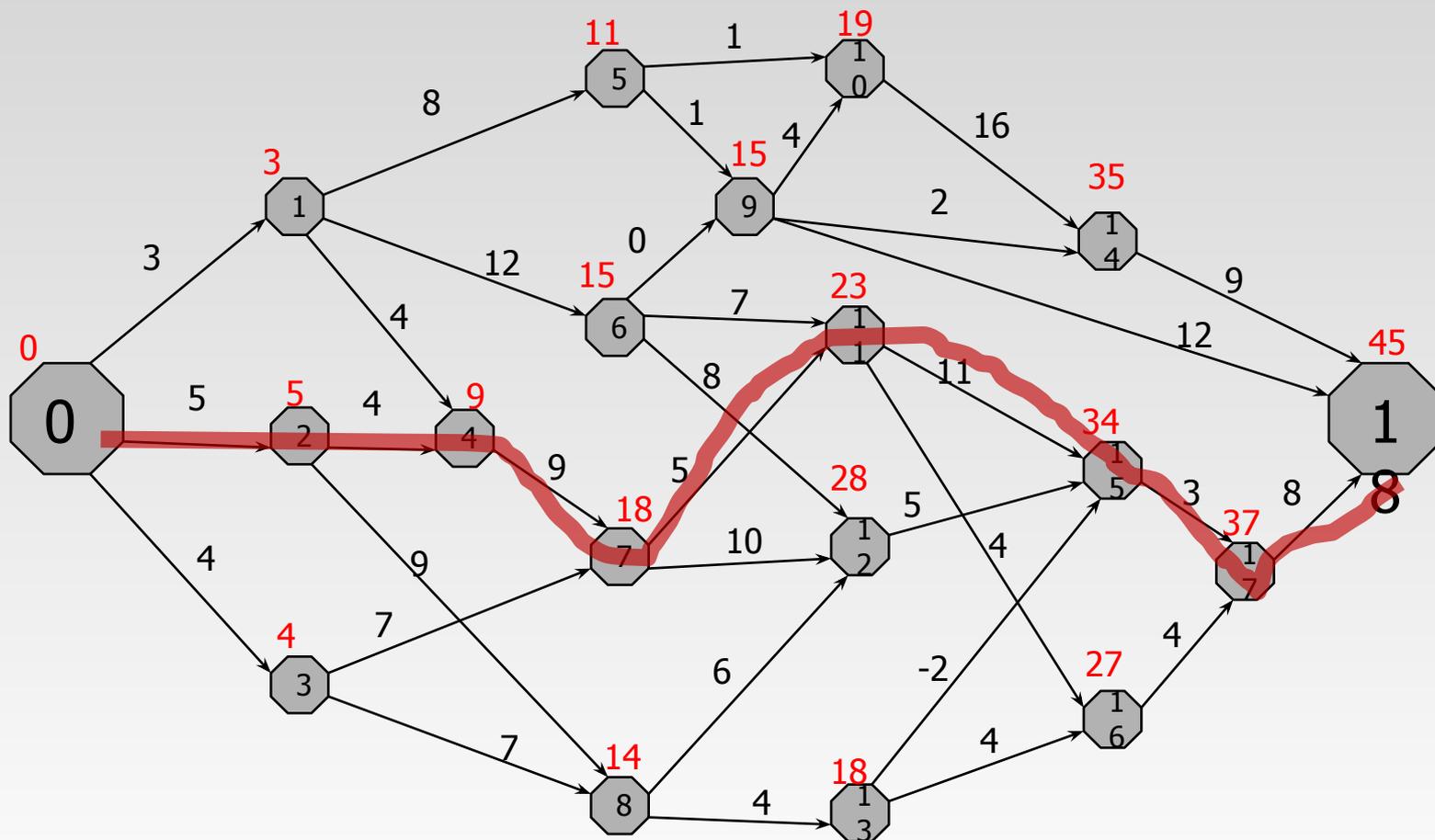
$\{(1,2), (1,3), (2,4), (2,5), (3,5)...\}$



6.2. Принцип оптимальности Беллмана

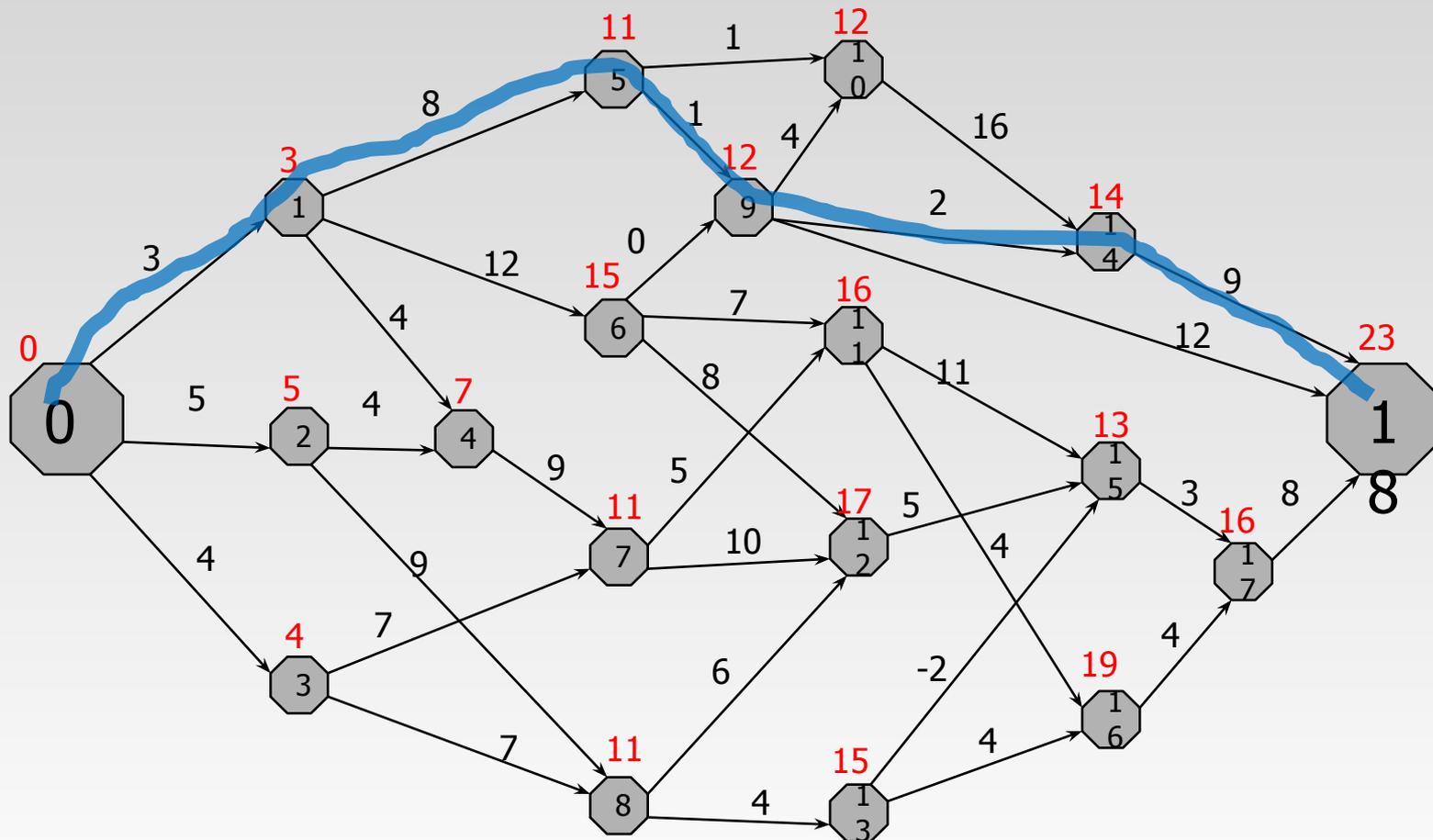
- Если вершины A и B лежат на оптимальном пути между вершинами O и X , то часть оптимального пути от O до X между вершинами A и B непременно является оптимальным путём от A до B .
- Следствие
 - ◆ Чтобы найти оптимальный путь от O до A , достаточно исследовать продолжения к A всех оптимальных путей до вершин, предшествующих A
 - ◆ Продолжения неоптимальных путей к предшествующим вершинам можно не просчитывать: они никогда не дадут оптимального пути к A
- Принцип Беллмана позволяет построить простую и эффективную вычислительную процедуру для решения задач динамического программирования

6.3. Алгоритм решения задач динамического программирования



максимум

6.3. Алгоритм решения задач динамического программирования



МИНИМУМ



6.4. Экономические приложения

Управление проектами

- Дуги - работы, которые должны быть выполнены
- Параметры – продолжительность работ
- Самый длинный путь (max) определяет минимальный срок выполнения проекта

Управление реновацией основных средств производства

- Дуги соответствуют решениям:
 - *e.g.*, эксплуатировать; ремонтировать; списать
- Параметры –доходы
- Самый выгодный путь (max) определяет жизненный цикл элемента основных средств

Бизнес-планирование

- Дуги – операции, переводящие фирму в новое состояние
- Параметры – доходы (расходы)
- Самый выгодный путь определяет наилучший бизнес-план

Поиск оптимального маршрута

- Дуги – пути сообщения
- Параметры – время (или стоимость) перевозки
- Самый выгодный (min) путь определяет оптимальную перевозку