

# Метод ветвей и сечений

Виктор Васильевич Лепин

# План лекции

- 1 Метод ветвей и границ
  - Дерево поиска
  
- 2 Метод ветвей и сечений
  - Описание метода

# Задача СЦП

Рассмотрим задачу СЦП

$$\max\{c^T x : b^1 \leq Ax \leq b^2, \quad d^1 \leq x \leq d^2, \quad x_j \in \mathbb{Z} \text{ для } j \in S\}.$$

где

- $b^1, b^2 \in \mathbb{R}^m$ ,

# Задача СЦП

Рассмотрим задачу СЦП

$$\max\{c^T x : b^1 \leq Ax \leq b^2, \quad d^1 \leq x \leq d^2, \quad x_j \in \mathbb{Z} \text{ для } j \in S\}.$$

где

- $b^1, b^2 \in \mathbb{R}^m$ ,
- $c, d^1, d^2 \in \mathbb{R}^n$ ,

# Задача СЦП

Рассмотрим задачу СЦП

$$\max\{c^T x : b^1 \leq Ax \leq b^2, \quad d^1 \leq x \leq d^2, \quad x_j \in \mathbb{Z} \text{ для } j \in S\}.$$

где

- $b^1, b^2 \in \mathbb{R}^m$ ,
- $c, d^1, d^2 \in \mathbb{R}^n$ ,
- $A$  – действительная  $m \times n$ -матрица,

# Задача СЦП

Рассмотрим задачу СЦП

$$\max\{c^T x : b^1 \leq Ax \leq b^2, \quad d^1 \leq x \leq d^2, \quad x_j \in \mathbb{Z} \text{ для } j \in S\}.$$

где

- $b^1, b^2 \in \mathbb{R}^m$ ,
- $c, d^1, d^2 \in \mathbb{R}^n$ ,
- $A$  – действительная  $m \times n$ -матрица,
- $x$  –  $n$ -вектор переменных (неизвестных),

# Задача СЦП

Рассмотрим задачу СЦП

$$\max\{c^T x : b^1 \leq Ax \leq b^2, \quad d^1 \leq x \leq d^2, \quad x_j \in \mathbb{Z} \text{ для } j \in S\}.$$

где

- $b^1, b^2 \in \mathbb{R}^m$ ,
- $c, d^1, d^2 \in \mathbb{R}^n$ ,
- $A$  – действительная  $m \times n$ -матрица,
- $x$  –  $n$ -вектор переменных (неизвестных),
- а  $S \subseteq \{1, \dots, n\}$  есть множество целочисленных переменных.

# Ветвление

- Базовой структурой метода ветвей и границ является **дерево поиска**.



# Ветвление

- Базовой структурой метода ветвей и границ является **дерево поиска**.
- **Корень** дерева поиска представляет исходную задачу.

# Ветвление

- Базовой структурой метода ветвей и границ является **дерево поиска**.
- **Корень** дерева поиска представляет исходную задачу.
- В ходе решения задачи дерево растет благодаря процессу, называемому **ветвлением**, который создает двух или более сыновей для одного из листьев текущего дерева поиска.

# Ветвление

- Базовой структурой метода ветвей и границ является **дерево поиска**.
- **Корень** дерева поиска представляет исходную задачу.
- В ходе решения задачи дерево растет благодаря процессу, называемому **ветвлением**, который создает двух или более сыновей для одного из листьев текущего дерева поиска.
- Каждая из задач СЦП в сыновних узлах получается из родительской задачи СЦП добавлением одного или нескольких новых ограничений.

# Ветвление

- Базовой структурой метода ветвей и границ является **дерево поиска**.
- **Корень** дерева поиска представляет исходную задачу.
- В ходе решения задачи дерево растет благодаря процессу, называемому **ветвлением**, который создает двух или более сыновей для одного из листьев текущего дерева поиска.
- Каждая из задач СЦП в сыновних узлах получается из родительской задачи СЦП добавлением одного или нескольких новых ограничений.
- Обычно новое ограничение – это верхняя или нижняя граница для переменной.

# Ветвление

- Базовой структурой метода ветвей и границ является **дерево поиска**.
- **Корень** дерева поиска представляет исходную задачу.
- В ходе решения задачи дерево растет благодаря процессу, называемому **ветвлением**, который создает двух или более сыновей для одного из листьев текущего дерева поиска.
- Каждая из задач СЦП в сыновних узлах получается из родительской задачи СЦП добавлением одного или нескольких новых ограничений.
- Обычно новое ограничение – это верхняя или нижняя граница для переменной.
- В процессе ветвления мы не должны потерять допустимые решения:

# Ветвление

- Базовой структурой метода ветвей и границ является **дерево поиска**.
- **Корень** дерева поиска представляет исходную задачу.
- В ходе решения задачи дерево растет благодаря процессу, называемому **ветвлением**, который создает двух или более сыновей для одного из листьев текущего дерева поиска.
- Каждая из задач СЦП в сыновних узлах получается из родительской задачи СЦП добавлением одного или нескольких новых ограничений.
- Обычно новое ограничение – это верхняя или нижняя граница для переменной.
- В процессе ветвления мы не должны потерять допустимые решения: объединение допустимых областей задач сыновей должно давать допустимую область их родителя.

## Верхняя и нижняя границы

- Главная идея в методе ветвей и границ состоит в том, чтобы не давать дереву поиска разрастаться, отсекая “бесперспективные” ветви.

## Верхняя и нижняя границы

- Главная идея в методе ветвей и границ состоит в том, чтобы не давать дереву поиска разрастаться, отсекая “бесперспективные” ветви.
- В методах ветвей и границ, основанных на линейном программировании, **верхней границей** в узле  $k$  является оптимальное значение  $\gamma(k)$  целевой функции релаксационной (без учета требований о целочисленности переменных) задачи ЛП в данном узле.



## Верхняя и нижняя границы

- Главная идея в методе ветвей и границ состоит в том, чтобы не давать дереву поиска разрастаться, отсекая “бесперспективные” ветви.
- В методах ветвей и границ, основанных на линейном программировании, **верхней границей** в узле  $k$  является оптимальное значение  $\gamma(k)$  целевой функции релаксационной (без учета требований о целочисленности переменных) задачи ЛП в данном узле.
- **Нижней границей** (или **рекордом**) называется наибольшее значение  $R$  целевой функции для уже найденных допустимых решений исходной задачи СЦП.

## Верхняя и нижняя границы

- Главная идея в методе ветвей и границ состоит в том, чтобы не давать дереву поиска разрастаться, отсекая “бесперспективные” ветви.
- В методах ветвей и границ, основанных на линейном программировании, **верхней границей** в узле  $k$  является оптимальное значение  $\gamma(k)$  целевой функции релаксационной (без учета требований о целочисленности переменных) задачи ЛП в данном узле.
- **Нижней границей** (или **рекордом**) называется наибольшее значение  $R$  целевой функции для уже найденных допустимых решений исходной задачи СЦП.
- Само наилучшее из полученных решений называется **рекордным решением**.

## Верхняя и нижняя границы

- Главная идея в методе ветвей и границ состоит в том, чтобы не давать дереву поиска разрастаться, отсекая “бесперспективные” ветви.
- В методах ветвей и границ, основанных на линейном программировании, **верхней границей** в узле  $k$  является оптимальное значение  $\gamma(k)$  целевой функции релаксационной (без учета требований о целочисленности переменных) задачи ЛП в данном узле.
- **Нижней границей** (или **рекордом**) называется наибольшее значение  $R$  целевой функции для уже найденных допустимых решений исходной задачи СЦП.
- Само наилучшее из полученных решений называется **рекордным решением**.
- Если  $\gamma(k) \leq R$ , то узел  $k$  и всех его потомков можно отсечь от дерева поиска.

# План лекции

- 1 Метод ветвей и границ
  - Дерево поиска
- 2 Метод ветвей и сечений
  - Описание метода

# Метод ветвей и сечений

- Метод ветвей и сечений – это метод ветвей и границ,

# Метод ветвей и сечений

- **Метод ветвей и сечений** – это метод ветвей и границ,
- в котором отсечения генерируются при решении релаксационной задачи ЛП во всех (или только некоторых) узлах дерева поиска.

# Метод ветвей и сечений

- **Метод ветвей и сечений** – это метод ветвей и границ,
- в котором отсекающие генерируются при решении релаксационной задачи ЛП во всех (или только некоторых) узлах дерева поиска.
- В отличие от “чистых” методов сечений, мы теперь не надеемся, что одних отсекающих будет достаточно для получения оптимального решения.

# Метод ветвей и сечений

- **Метод ветвей и сечений** – это метод ветвей и границ,
- в котором отсекающие генерируются при решении релаксационной задачи ЛП во всех (или только некоторых) узлах дерева поиска.
- В отличие от “чистых” методов сечений, мы теперь не надеемся, что одних отсекающих будет достаточно для получения оптимального решения.
- Заметим также, что раньше, как правило, генерировалось только одно неравенство, отсекающее текущее дробное решение.



# Метод ветвей и сечений

- **Метод ветвей и сечений** – это метод ветвей и границ,
- в котором отсекающие генерируются при решении релаксационной задачи ЛП во всех (или только некоторых) узлах дерева поиска.
- В отличие от “чистых” методов сечений, мы теперь не надеемся, что одних отсекающих будет достаточно для получения оптимального решения.
- Заметим также, что раньше, как правило, генерировалось только одно неравенство, отсекающее текущее дробное решение.
- Сегодня такой способ считается плохим, отсекающие теперь добавляются группами из многих неравенств.

## Разрыв двойственности

- Две величины определяют поведение метода ветвей и сечений (равно как и метода ветвей и границ) – это нижняя граница (рекорд) и верхние границы (оптимальные значения целевых функций для релаксационных задач ЛП) в узлах дерева поиска.

## Разрыв двойственности

- Две величины определяют поведение метода ветвей и сечений (равно как и метода ветвей и границ) – это нижняя граница (рекорд) и верхние границы (оптимальные значения целевых функций для релаксационных задач ЛП) в узлах дерева поиска.
- Добавление отсечений способствует уменьшению верхних границ.

## Разрыв двойственности

- Две величины определяют поведение метода ветвей и сечений (равно как и метода ветвей и границ) – это нижняя граница (рекорд) и верхние границы (оптимальные значения целевых функций для релаксационных задач ЛП) в узлах дерева поиска.
- Добавление отсечений способствует уменьшению верхних границ.
- Если в методе ветвей и границ при обработке очередного узла дерева поиска главной целью было поскорее решить соответствующую ему релаксационную задачу ЛП,

## Разрыв двойственности

- Две величины определяют поведение метода ветвей и сечений (равно как и метода ветвей и границ) – это нижняя граница (рекорд) и верхние границы (оптимальные значения целевых функций для релаксационных задач ЛП) в узлах дерева поиска.
- Добавление отсечений способствует уменьшению верхних границ.
- Если в методе ветвей и границ при обработке очередного узла дерева поиска главной целью было поскорее решить соответствующую ему релаксационную задачу ЛП,
- то теперь мы выполняем существенно большую работу в каждом узле, генерируя отсечения с целью минимизировать верхнюю границу.

## Разрыв двойственности

- Две величины определяют поведение метода ветвей и сечений (равно как и метода ветвей и границ) – это нижняя граница (рекорд) и верхние границы (оптимальные значения целевых функций для релаксационных задач ЛП) в узлах дерева поиска.
- Добавление отсечений способствует уменьшению верхних границ.
- Если в методе ветвей и границ при обработке очередного узла дерева поиска главной целью было поскорее решить соответствующую ему релаксационную задачу ЛП,
- то теперь мы выполняем существенно большую работу в каждом узле, генерируя отсечения с целью минимизировать верхнюю границу.
- При этом **разрыв двойственности** (разность между верхней и нижней границей) в узле также уменьшается.

# Генерирование отсечений

- На практике очень важно определить момент, когда нужно прекратить генерировать новые отсечения и приступить к ветвлению.

## Генерирование отсечений

- На практике очень важно определить момент, когда нужно прекратить генерировать новые отсечения и приступить к ветвлению.
- Если добавляется много отсечений в каждом узле, то на дооптимизацию узловых задач ЛП может потребоваться существенно большее время.



## Генерирование отсечений

- На практике очень важно определить момент, когда нужно прекратить генерировать новые отсечения и приступить к ветвлению.
- Если добавляется много отсечений в каждом узле, то на дооптимизацию узловых задач ЛП может потребоваться существенно большее время.
- Разумная стратегия состоит в том, чтобы следить за тем, как сокращается разрыв двойственности.

## Генерирование отсечений

- На практике очень важно определить момент, когда нужно прекратить генерировать новые отсечения и приступить к ветвлению.
- Если добавляется много отсечений в каждом узле, то на дооптимизацию узловых задач ЛП может потребоваться существенно большее время.
- Разумная стратегия состоит в том, чтобы следить за тем, как сокращается разрыв двойственности.
- Если прогресса нет на протяжении нескольких раундов, то самое время остановиться.

## Узловые эвристики

- Другой способ сократить разрыв двойственности в узле состоит в применении узловых эвристик с целью увеличить нижнюю границу (рекорд).

## Узловые эвристики

- Другой способ сократить разрыв двойственности в узле состоит в применении узловых эвристик с целью увеличить нижнюю границу (рекорд).
- Идея **узловой эвристики** проста. Прежде чем приступить к ветвлению в конкретном узле, можно попробовать “округлить” оптимальное решение задачи ЛП в данном узле.

## Узловые эвристики

- Другой способ сократить разрыв двойственности в узле состоит в применении узловых эвристик с целью увеличить нижнюю границу (рекорд).
- Идея **узловой эвристики** проста. Прежде чем приступить к ветвлению в конкретном узле, можно попробовать “округлить” оптимальное решение задачи ЛП в данном узле.
- Обычно округление состоит в выполнении некоторого типа “ныряния”, когда

## Узловые эвристики

- Другой способ сократить разрыв двойственности в узле состоит в применении узловых эвристик с целью увеличить нижнюю границу (рекорд).
- Идея **узловой эвристики** проста. Прежде чем приступить к ветвлению в конкретном узле, можно попробовать “округлить” оптимальное решение задачи ЛП в данном узле.
- Обычно округление состоит в выполнении некоторого типа “ныряния”, когда
  - фиксируются значения группы целочисленных переменных с почти целыми значениями,

## Узловые эвристики

- Другой способ сократить разрыв двойственности в узле состоит в применении узловых эвристик с целью увеличить нижнюю границу (рекорд).
- Идея **узловой эвристики** проста. Прежде чем приступить к ветвлению в конкретном узле, можно попробовать “округлить” оптимальное решение задачи ЛП в данном узле.
- Обычно округление состоит в выполнении некоторого типа “ныряния”, когда
  - фиксируются значения группы целочисленных переменных с почти целыми значениями,
  - решается полученная задача ЛП,

## Узловые эвристики

- Другой способ сократить разрыв двойственности в узле состоит в применении узловых эвристик с целью увеличить нижнюю границу (рекорд).
- Идея **узловой эвристики** проста. Прежде чем приступить к ветвлению в конкретном узле, можно попробовать “округлить” оптимальное решение задачи ЛП в данном узле.
- Обычно округление состоит в выполнении некоторого типа “ныряния”, когда
  - фиксируются значения группы целочисленных переменных с почти целыми значениями,
  - решается полученная задача ЛП,
  - затем фиксируется еще одна группа переменных,



## Узловые эвристики

- Другой способ сократить разрыв двойственности в узле состоит в применении узловых эвристик с целью увеличить нижнюю границу (рекорд).
- Идея **узловой эвристики** проста. Прежде чем приступить к ветвлению в конкретном узле, можно попробовать “округлить” оптимальное решение задачи ЛП в данном узле.
- Обычно округление состоит в выполнении некоторого типа “ныряния”, когда
  - фиксируются значения группы целочисленных переменных с почти целыми значениями,
  - решается полученная задача ЛП,
  - затем фиксируется еще одна группа переменных,
  - и так до тех пор, пока не будет получено целочисленное решение или фиксирование переменных приведет к недопустимости.

## Узловые эвристики

- Другой способ сократить разрыв двойственности в узле состоит в применении узловых эвристик с целью увеличить нижнюю границу (рекорд).
- Идея **узловой эвристики** проста. Прежде чем приступить к ветвлению в конкретном узле, можно попробовать “округлить” оптимальное решение задачи ЛП в данном узле.
- Если таким образом удастся увеличить нижнюю границу (будет получено целочисленное решение, лучшее рекордного),

## Узловые эвристики

- Другой способ сократить разрыв двойственности в узле состоит в применении узловых эвристик с целью увеличить нижнюю границу (рекорд).
- Идея **узловой эвристики** проста. Прежде чем приступить к ветвлению в конкретном узле, можно попробовать “округлить” оптимальное решение задачи ЛП в данном узле.
- Если таким образом удастся увеличить нижнюю границу (будет получено целочисленное решение, лучшее рекордного),
- то это может позволить отсеять некоторые активные узлы дерева поиска и тем самым ускорить решение всей задачи.