

Линейное программирование

Задача о покрытии и
задача о максимальной
выполнимости

Задача «Покрытие»

- *Дано:* Совокупность U из n элементов, и набор подмножеств U , $\Omega = \{S_1, \dots, S_k\}$, и веса(стоимости) подмножеств $c: \Omega \rightarrow \mathbf{Q}^+$.
- *Найти* покрытие наименьшего веса.
- $\Omega' \subseteq \Omega$ является покрытием, если любой элемент из U принадлежит хотя бы одному элементу из Ω' .

Кратность

- Пусть f_i — кратность элемента e_i , то есть число множеств из Ω , в которые он входит.
- Пусть $f = \max_{i=1, \dots, n} f_i$.

ЦЛП (Задача о покрытии)

$$\begin{aligned} & \text{minimize} && \sum_{S \in \Omega} c(S) x_S \\ & \text{s.t.} && \sum_{S: e \in S} x_S \geq 1, && e \in U \quad (1) \\ & && x_S \in \{0, 1\} && S \in \Omega \end{aligned}$$

ЛП-релаксация (Задача о покрытии)

$$\begin{aligned} &\text{minimize} && \sum_{S \in \Omega} c(S) x_S \\ &\text{s.t.} && \sum_{S: e \in S} x_S \geq 1, && e \in U \quad (1) \\ &&& x_S \geq 0 && S \in \Omega \end{aligned}$$

Алгоритм «Округление ЛП-релаксации»

- 0) **Input** ($U, \Omega, c: \Omega \rightarrow \mathbf{Q}^+$)
- 1) Найти оптимальное решение
ЛП-релаксации.
- 2) Выбрать все множества S ,
для которых $x_S \geq 1/f$.
- 3) **Output**

f -приближение

Теорема 10.1

Алгоритм «Округление ЛП-релаксации» является f -приближенным алгоритмом для задачи о покрытии.

Доказательство.

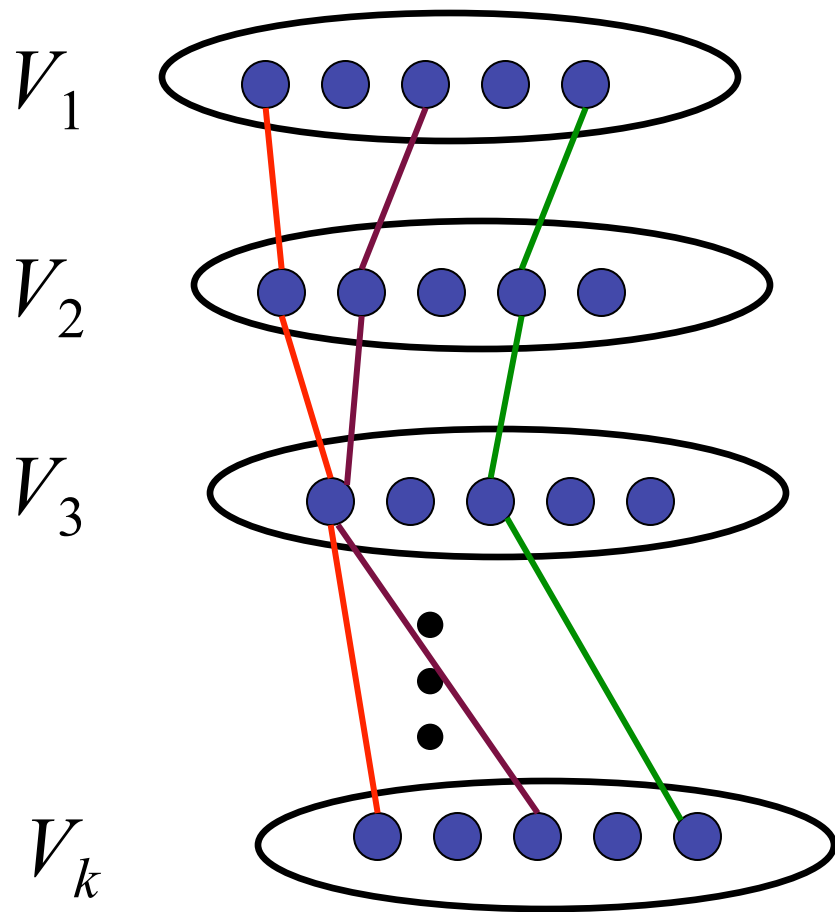
- $e \in U: (1) \Rightarrow \exists x_S \geq 1/f (e \in S) \Rightarrow$ все элементы покрыты.
- Так как $x_S \geq 1/f$ для всех выбранных S , стоимость полученного покрытия превышает стоимость дробного покрытия не больше чем в f раз.

2-приближение

Следствие 10.2

Алгоритм «Округление ЛП-релаксации» является 2-приближенным алгоритмом для задачи о вершинном покрытии.

Точный пример (гиперграф)



$$V = V_1 \cup V_2 \cup \dots \cup V_k$$

n^k гиперребер

Каждое гиперребро соответствует элементу, вершина – множеству (n^k множеств; стоимость всех 1), которое содержит инцидентные элементы-гиперребра.

Прямая и двойственная задачи

$$\begin{aligned} & \sum_{j=1}^n c_j x_j \rightarrow \min \\ s.t. \quad & \sum_{j=1}^n a_{ij} x_j \geq b_i, \quad i = 1, \dots, m \\ & x_j \geq 0, \quad j = 1, \dots, n \end{aligned}$$

$$\begin{aligned} & \sum_{i=1}^m b_i y_i \rightarrow \max \\ s.t. \quad & \sum_{i=1}^m a_{ij} y_i \leq c_j, \quad j = 1, \dots, n \\ & y_i \geq 0, \quad i = 1, \dots, m \end{aligned}$$

1-ая теорема двойственности

Прямая и двойственная к ним задачи либо одновременно разрешимы, либо одновременно неразрешимы. При этом в первом случае *значения целевых функций этих задач совпадают*, а во втором случае, по крайней мере, одна из задач неразрешима в силу несовместности ее ограничений.

2-ая теорема двойственности

- Допустимые решения x' и y' соответственно прямой и двойственной задачи оптимальны тогда и только тогда, когда

$$\forall j : x_j \left(c_j - \sum_{i=1}^m a_{ij} y_i \right) = 0.$$

$$\forall i : y_i \left(\sum_{j=1}^m a_{ij} x_j - b_i \right) = 0.$$

Релаксированные условия дополняющей нежесткости

$$\sum_{j=1}^n c_j x_j \rightarrow \min$$

$$\begin{aligned} s.t. \quad & \sum_{j=1}^n a_{ij} x_j \geq b_i, \quad i = 1, \dots, m \\ & x_j \geq 0, \quad j = 1, \dots, n \end{aligned}$$

$$\sum_{i=1}^m b_i y_i \rightarrow \max$$

$$\begin{aligned} s.t. \quad & \sum_{i=1}^m a_{ij} y_i \leq c_j, \quad j = 1, \dots, n \\ & y_i \geq 0, \quad i = 1, \dots, m \end{aligned}$$

Прямое условие

$$(\alpha \geq 1), \forall j, 1 \leq j \leq n : x_j = 0 \vee \frac{c_j}{\alpha} \leq \sum_{i=1}^m a_{ij} y_i \leq c_j.$$

Двойственное условие

$$(\beta \geq 1), \forall i, 1 \leq i \leq m : y_i = 0 \vee b_i \leq \sum_{j=1}^n a_{ij} x_j \leq \beta b_i.$$

Соотношение между целевыми функциями

- **Утверждение 10.3**

Если x и y прямое и двойственное допустимые решения, удовлетворяющие релаксированным условиям дополняющей нежесткости, то

$$\sum_{j=1}^n c_j x_j \leq \alpha \beta \sum_{i=1}^m b_i y_i.$$

Доказательство Утверждения 8.3

$$\begin{aligned} \sum_{j=1}^n c_j x_j \stackrel{primal}{\leq} \alpha \sum_{j=1}^n \left(\sum_{i=1}^m a_{ij} y_i \right) x_j &= \\ = \alpha \sum_{i=1}^m \left(\sum_{j=1}^n a_{ij} x_j \right) y_i \stackrel{dual}{\leq} \alpha \beta \sum_{i=1}^m b_i y_i. \end{aligned}$$

Идея прямо-двойственной схемы

- Алгоритм начинает работу с прямого недопустимого решения и с двойственного допустимого решения. Как правило это тривиальное решение $x = 0$ и $y = 0$.
- Итеративно улучшается допустимость прямого решения и оптимальность двойственного решения, так чтобы в конце получить допустимое прямое решение, при котором выполнены релаксированные условия дополняющей нежесткости для выбранных α и β .
- При расширении прямое решение всегда остается целочисленным.
- Улучшение прямого и двойственного решения происходит совместно, то есть текущее прямое решение используется для улучшения двойственного и наоборот.
- В конце стоимость двойственного решения используется как нижняя оценка на величину оптимального решения.

ЛП-релаксация (Задача о покрытии)

$$\begin{aligned} &\text{minimize} && \sum_{S \in \Omega} c(S) x_S \\ &\text{s.t.} && \sum_{S: e \in S} x_S \geq 1, && e \in U \\ &&& x_S \geq 0 && S \in \Omega \end{aligned}$$

Двойственная программа (Задача о покрытии)

$$\begin{aligned} &\text{maximize} && \sum_{e \in U} y_e \\ &\text{s.t.} && \sum_{e: e \in S} y_e \leq c(S), \quad S \in \Omega \\ &&& y_e \geq 0, \quad e \in U \end{aligned}$$

$$\alpha = 1, \beta = f$$

Прямое условие

$$\forall S \in \Omega : x_S \neq 0 \Rightarrow \sum_{e: e \in S} y_e = c(S)$$

Двойственное условие

$$\forall e : y_e \neq 0 \Rightarrow \sum_{S: e \in S} x_S \leq f$$

Множество S называется строгим, если $\sum_{e: e \in S} y_e = c(S)$

Каждый элемент имеющий ненулевое двойственное значение может быть покрыт не более f раз.

Прямо-двойственный алгоритм

0) **Input** ($U, \Omega, c: \Omega \rightarrow \mathbf{Q}^+$)

1) $x \leftarrow 0, y \leftarrow 0$.

2) **Until** все элементы не покрыты, **do**

- Выбрать непокрытый элемент, например e , и увеличить y_e до тех пор пока какое-либо множество не станет строгим.
- Поместить все строгие множества в покрытие и обновить x .
- Объявить все элементы появляющиеся в этих множествах «покрытыми».

3) **Output** (x)

f -приближение

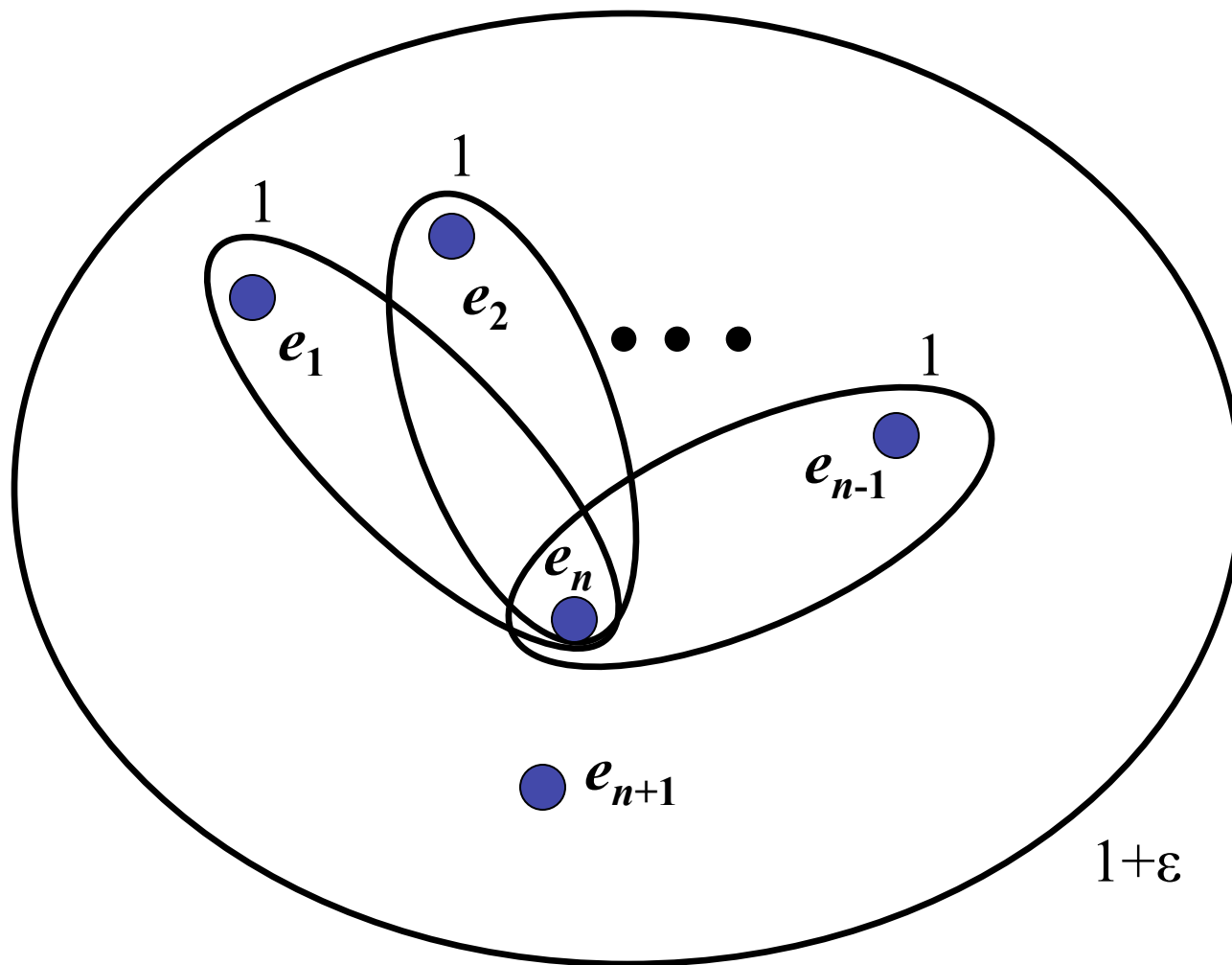
- Теорема 10.4

Прямо-двойственный алгоритм является f -приближенным алгоритмом для задачи о покрытии.

Доказательство Теоремы 10.4

- Алгоритм останавливается когда все элементы покрыты (допустимость).
- В покрытие выбираются только строгие множества и значения u_e элементов, которые в них входят больше не изменяются (допустимость и прямое условие).
- Каждый элемент входит не больше чем в f множеств по условию задачи (двойственное условие).
- Следовательно алгоритм находит допустимое решение и по утверждению 8.3 является f -приближенным.

Точный пример



$$f = n$$

Задача

«Максимальная выполнимость»

- *Дано:* Набор f дизъюнкций на булевых переменных $x_1, \dots, x_n$, и веса дизъюнкций $w: f \rightarrow \mathbf{Q}^+$.
- *Найти* назначение истинности на булевых переменных, максимизирующее общий вес выполненных дизъюнкций .
- Дизъюнкция c выполнена, если она принимает значение «истина» ($c=1$).

Обозначения

- W – общий вес выполненных дизъюнкций.
- Для каждой дизъюнкции $c \in f$, случайная величина W_c обозначает вес вносимый дизъюнкцией c в W .

$$W = \sum_{c \in f} W_c,$$

$$E[W_c] = w_c \Pr[c = 1]$$

Вероятностный алгоритм Джонсона

0) **Input** $(x_1, \dots, x_n, f, w: f \rightarrow \mathbf{Q}^+)$

1) Независимо для каждого i :

$x_i \leftarrow 1$ с вероятностью 0.5 и $x_i \leftarrow 0$, иначе.

Назовем полученное назначение τ .

3) **Output** (τ)

Оценка на вес дизъюнкций

- Для $k \geq 1$, $\alpha_k = 1 - 2^{-k}$.

- **Лемма 10.5**

Если $\text{size}(c) = k$, то $E[W_c] = \alpha_k w_c$.

- **Следствие 10.6**

$$E[W] \geq \frac{1}{2} \text{OPT}.$$

Вычисление условных средних

- Пусть $a_1, \dots, a_i$ будет назначение истинности на переменных $x_1, \dots, x_i$.
- **Лемма 10.7**

$E[W \mid x_1=a_1, \dots, x_i=a_i]$ может быть вычислено за время ограниченное полиномом от размера входа.

Дерандомизация

- Теорема 10.8

Существует такое назначение истинности $x_1=a_1, \dots, x_n=a_n$, что $W(a_1, \dots, a_n) \geq E[W]$ и оно может быть вычислено за полиномиальное время.

Доказательство (индуктивный шаг)

- $E[W \mid x_1=a_1, \dots, x_i=a_i] = E[W \mid x_1=a_1, \dots, x_i=a_i, x_{i+1}=\text{True}]/2 + E[W \mid x_1=a_1, \dots, x_i=a_i, x_{i+1}=\text{False}]/2$
- Следовательно,
либо $E[W \mid x_1=a_1, \dots, x_i=a_i, x_{i+1}=\text{True}] \geq E[W \mid x_1=a_1, \dots, x_i=a_i]$,
либо $E[W \mid x_1=a_1, \dots, x_i=a_i, x_{i+1}=\text{False}] \geq E[W \mid x_1=a_1, \dots, x_i=a_i]$.
- Выберем назначение с большим средним.
- Процедура требует n шагов, каждый из которых по лемме 8.7 выполняется за полиномиальное время.

Комментарий

- Подобная техника может быть использована и в более общем случае, даже если значения переменных зависят друг от друга. Действительно,

$$E[W | x_1=a_1, \dots, x_i=a_i] =$$

$$E[W | x_1=a_1, \dots, x_i=a_i, x_{i+1} = \text{True}] \cdot \Pr[x_{i+1} = \text{True} | x_1=a_1, \dots, x_i=a_i] +$$
$$E[W | x_1=a_1, \dots, x_i=a_i, x_{i+1} = \text{False}] \cdot \Pr[x_{i+1} = \text{False} | x_1=a_1, \dots, x_i=a_i].$$

$$\Pr[x_{i+1} = \text{True} | x_1=a_1, \dots, x_i=a_i] + \Pr[x_{i+1} = \text{False} | x_1=a_1, \dots, x_i=a_i] = 1.$$

Следовательно,

$$\text{либо } E[W | x_1=a_1, \dots, x_i=a_i, x_{i+1} = \text{True}] \geq E[W | x_1=a_1, \dots, x_i=a_i],$$
$$\text{либо } E[W | x_1=a_1, \dots, x_i=a_i, x_{i+1} = \text{False}] \geq E[W | x_1=a_1, \dots, x_i=a_i].$$

Несимметричные примеры

- Рассмотрим пример задачи о максимальной выполнимости в котором нет дизъюнкций состоящей из одного отрицательного литерала.
- Положим каждой переменной $x_i = \text{True}$ независимо с вероятностью $p > 1/2$.
- Для каждой дизъюнкции найдем вероятность, с которой она выполняется.

- **Лемма 10.9**

Назначим каждой переменной x_i значение 1 независимо с вероятностью $p > 1/2$, тогда вероятность, что любая дизъюнкция выполнена не меньше чем $\min(p, 1-p^2)$.

Доказательство

- Если дизъюнкция состоит из одного литерала, то вероятность, что она выполнена, равна p .
- Если дизъюнкция состоит из двух или более литералов, то вероятность, что она выполнена, равна $1 - p^a(1 - p)^b$, где a число ее отрицательных переменных, а b число ее положительных переменных, и $a + b \geq 2$.
- Так как $p > 1/2 > 1 - p$, получим $1 - p^{a+b} \geq 1 - p^2$, откуда следует утверждение леммы.

Улучшенная оценка точности

- Теорема 10.10

Назначение каждой переменной x_i значения 1 независимо с вероятностью $p > 1/2$ позволяет найти решение в задаче без единичных отрицательных дизъюнкций решение с $E[W] \geq \min(p, 1 - p^2) \text{ OPT}$

$$E[W] = \sum_{c \in f} E[W_c] \geq \min(p, 1 - p^2) \sum_{c \in f} w_c \geq \min(p, 1 - p^2) \text{OPT}.$$

- Следовательно, можно улучшить оценку точности, решив уравнение $p = 1 - p^2$, и получив $p = 1/2 (\sqrt{5} - 1) \approx 0.618$.

Общий случай

- Предположим, что для всех i вес литерала x_i не меньше, чем вес литерала $\bar{x}_i$.
- Пусть v_i равен весу литерала $\bar{x}_i$, если он есть в данном наборе дизъюнкций, и 0 иначе.

Новая верхняя граница

- Лемма 10.11

$$OPT \leq \sum_{c \in f} w_c - \sum_{\bar{x}_i \in f} v_i.$$

Доказательство. Для каждого i , в оптимальном решении выполнена, либо дизъюнкция x_i , либо $\bar{x}_i$. Вычитая для каждого i меньший из весов из общего веса всех дизъюнкций, получим утверждение леммы.

0.618-approximation algorithm

- Теорема 10.12

Существует $\frac{\sqrt{5}-1}{2}$ -приближенный алгоритм для задачи о максимальной выполнимости.

Доказательство

- Пусть набор φ получается из f удалением всех дизъюнкций, состоящих ровно из одного отрицательного литерала.
- Тогда $\sum_{c \in \varphi} w_c = \sum_{c \in f} w_c - \sum_{\bar{x}_i \in f} v_i$.
- Положим $x_i = 1$ независимо с вероятностью $p = \frac{\sqrt{5}-1}{2}$.

$$\mathbf{E}[W] = \sum_{c \in f} \mathbf{E}[W_c] = \sum_{c \in f} w_c \Pr[c = 1]$$

$$\geq \sum_{c \in \varphi} w_c \Pr[c = 1] \geq p \sum_{c \in \varphi} w_c = p \left(\sum_{c \in f} w_c - \sum_{\bar{x}_i \in f} v_i \right) \geq p \cdot OPT.$$

ЦЛП задачи «Максимальная выполнимость»

$$\begin{aligned} &\text{maximize} && \sum_{c \in f} w_c z_c \\ &\text{s.t.} && \sum_{i \in S_c^+} y_i + \sum_{i \in S_c^-} (1 - y_i) \geq z_c, \quad c \in f \\ &&& z_c \in \{0, 1\}, \quad c \in f \\ &&& y_i \in \{0, 1\}, \quad i = 1, \dots, n \end{aligned}$$

ЛП задачи

«Максимальная выполнимость»

$$\begin{aligned} &\text{maximize} && \sum_{c \in f} w_c z_c \\ &\text{s.t.} && \sum_{i \in S_c^+} y_i + \sum_{i \in S_c^-} (1 - y_i) \geq z_c, \quad c \in f \\ &&& 1 \geq z_c \geq 0, \quad c \in f \\ &&& 1 \geq y_i \geq 0, \quad i = 1, \dots, n \end{aligned}$$

Вероятностный алгоритм ЛП

- 0) **Input** $(x_1, \dots, x_n, f, w: f \rightarrow \mathbf{Q}^+)$
- 1) Решить ЛП задачи «Максимальная выполнимость». Пусть (y^*, z^*) обозначает оптимальное решение.
- 2) Независимо для каждого i : $x_i \leftarrow 1$ с вероятностью y_i^* , и $x_i \leftarrow 0$, иначе.
Назовем полученное назначение τ .
- 3) **Output** (τ)

Оценка на вес дизъюнкций

- Для $k \geq 1$, $\beta_k = 1 - (1 - 1/k)^k$.
- **Лемма 10.9**

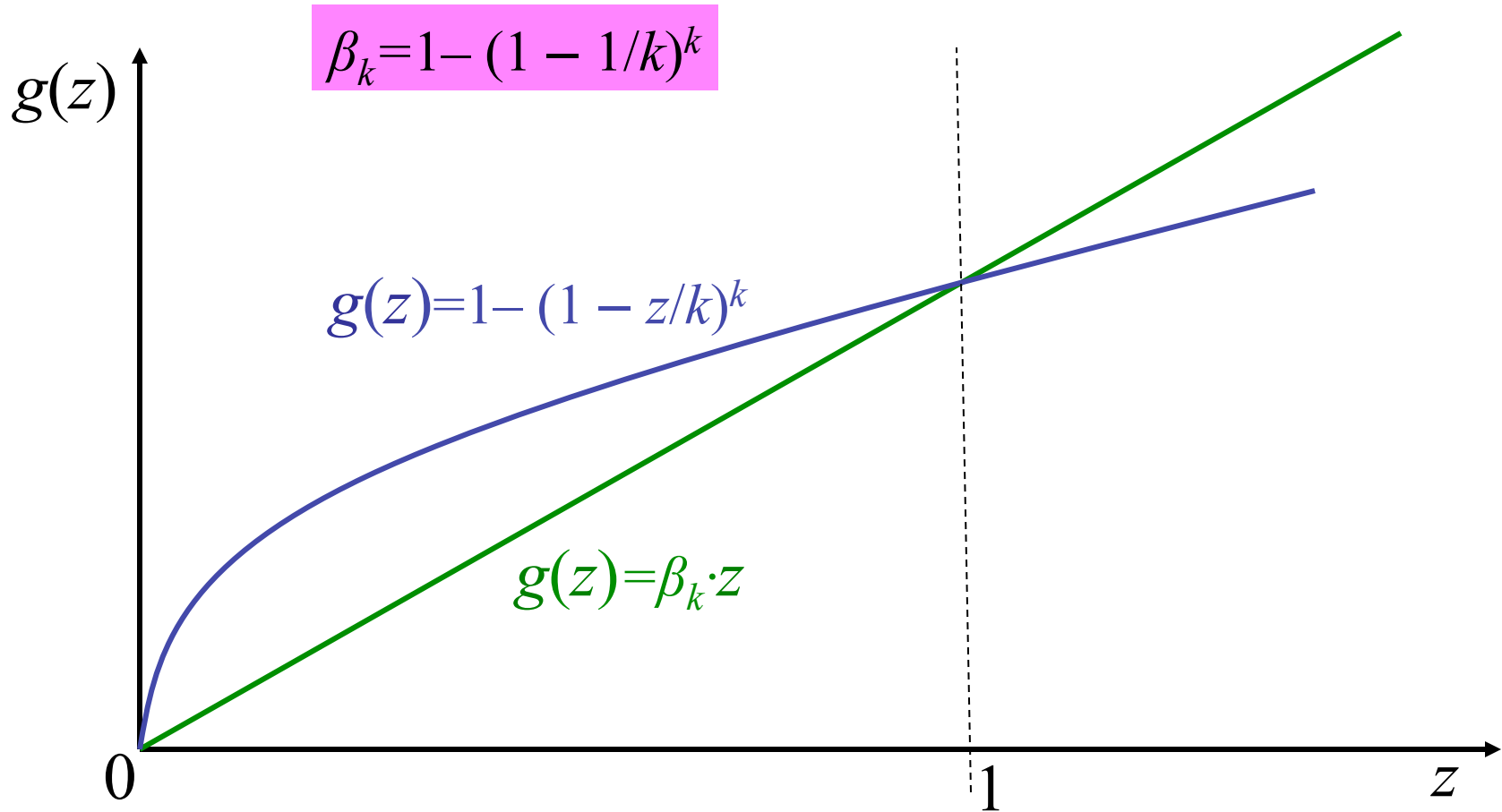
Если $\text{size}(c) = k$, то $E[W_c] \geq \beta_k w_c z^*(c)$.

Доказательство

$$c = (x_1 \vee \dots \vee x_k):$$

$$\begin{aligned} \Pr[c = \textit{True}] &= 1 - \prod_{i=1}^k (1 - y_i) \geq 1 - \left(\frac{\sum_{i=1}^k (1 - y_i)}{k} \right)^k = \\ &= 1 - \left(1 - \frac{\sum_{i=1}^k y_i}{k} \right)^k \stackrel{LP}{\geq} 1 - \left(1 - \frac{z^*(c)}{k} \right)^k. \end{aligned}$$

Доказательство



$$\Pr[c = \text{True}] \geq 1 - \left(1 - \frac{z^*(c)}{k}\right)^k \geq \beta_k z^*(c).$$

$$1 - 1/e$$

- **Следствие 10.10**

$E[W] \geq \beta_k \text{OPT}$ (если размер дизъюнкций $\leq k$).

$$E[W] = \sum_{c \in f} E[W_c] \geq \beta_k \sum_{c \in f} w_c z_c^* = \beta_k \text{OPT}$$

- **Теорема 10.11**

Дерандомизированный вероятностный алгоритм ЛП является $(1 - 1/e)$ -приближенным алгоритмом.

Идея $(3/4)$ -приближенного алгоритма

- С равной вероятностью применим один из двух описанных алгоритмов.
- Пусть $b=0$, если мы применили алгоритм Джонсона и $b=1$, иначе.
- Пусть z^* обозначает оптимальное решение ЛП.
- **Лемма 10.12**
$$E[W_c] \geq (3/4)w_c z^*(c).$$

$$\mathbb{E}[W_c] \geq (3/4)w_c z^*(c)$$

- Пусть $\text{size}(c)=k$.
- Л 8.5 $\Rightarrow \mathbb{E}[W_c|b=0] = \alpha_k w_c \geq \alpha_k w_c z^*(c)$
- Л 8.9 $\Rightarrow \mathbb{E}[W_c|b=1] \geq \beta_k w_c z^*(c)$
- $\mathbb{E}[W_c] = (1/2)(\mathbb{E}[W_c|b=0] + \mathbb{E}[W_c|b=1]) \geq$
 $\geq (1/2)w_c z^*(c)(\alpha_k + \beta_k)$
- $\alpha_1 + \beta_1 = \alpha_2 + \beta_2 = 3/2$
- $k \geq 3, \alpha_k + \beta_k \geq 7/8 + (1 - 1/e) > 3/2$
- $\mathbb{E}[W_c] \geq (3/4)w_c z^*(c)$

Оценка на $E[W]$

$$E[W] = \sum_{c \in f} E[W_c] \geq \frac{3}{4} \sum_{c \in f} w_c z^*(c) = \frac{3}{4} \text{OPT}_{LP} \geq \frac{3}{4} \text{OPT}$$

Алгоритм Гоеманса-Вильямсона

- 0) **Input** $(x_1, \dots, x_n, f, w: f \rightarrow \mathbf{Q}^+)$
- 1) Решить задачу дерандомизированным вероятностным алгоритмом Джонсона. Назовем полученное назначение τ_1 .
- 2) Решить задачу дерандомизированным вероятностным алгоритмом ЛП. Назовем полученное назначение τ_2 .
- 4) **Output** (лучшее из τ_1 и τ_2)

(3/4)-приближенный алгоритм

- Теорема 10.13

Алгоритм Гоеманса-Вильямсона
(3/4)-приближенный алгоритм для
задачи «Максимальная выполнимость».