Approximation algorithms: Linear and Integer Programming

Maria Serna

Spring 2022

1. LP and IP
2. Relax and round
3. Primal-Dual

Linear programming

- In a linear programming problem, we are given a set of variables, an objective linear function a set of linear constrains and want to assign real values to the variables as to:
    - satisfy the set of linear inequalities (equations or constraints),
    - maximize or minimize the objective function.
- LP is a pure algebraic problem.

Linear programming: An example

$$\max\ x_1 + 6x_2$$

subject to

$$x_1 \leq 200$$

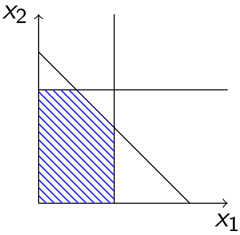
$$x_2 \leq 300$$

$$x_1 + x_2 \leq 400$$

$$x_1, x_2 \geq 0$$

Linear programming: feasible region

- A linear equality defines a hyperplane.
- A linear inequality defines a half-space.
- The solutions to the linear constraints lie inside a feasible region limited by the polytope (convex polygon in $\mathbb{R}^2$) defined by the linear constraints.

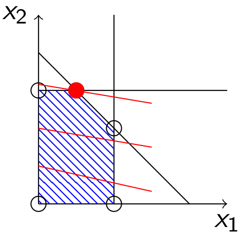


Linear programming: infeasibility

- A linear programming is infeasible if
  - The constrains are so tight that it is impossible to satisfy all of them.
    For ex. $x \geq 2$ and $x \leq 1$
  - The constrains are so loose that the feasible region is unbounded allowing the objective function to go to ∞ .
    For ex. $\max\ x_1 + x_2$ subject to $x_1, x_2 \geq 0$

Linear programming: optimum

- In a feasible linear programming the optimum is achieved at a vertex of the feasible region.



Linear programming: standard formulation

A LP has many degrees of freedom.

- maximization or minimization.
- constrains could be $=$, $\geq$, $\leq$, $<$ or $>$.
- variables are often restricted to be non-negative, but they also could be unrestricted.

- standard form?

Linear programming: standard formulation

- From max to min (or min to max)
  multiply by -1 the coefficients of the objective function.
- To reverse an inequality (for ex. $\geq$ to $\leq$)
  multiply all coefficients and the independent term by -1.
- From $<$ to $\leq$ (or to $=$)
  create a new positive variable and add it with coefficient 1 to the left par of the inequality.
- From $=$ to $\leq$ (or to $\geq$)
  put two versions one with $\leq$ and the other with $\geq$, multiply the last one by -1.
- From x unrestricted to non-negative variables,
  create two new variables x^+ and x^-, both non negative, replace x by $x^+ - x^-$.

Linear programming: standard formulation

LP standard form

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & x \geq 0 \end{aligned}$$

Where

- $x = (x_1, \ldots, x_n)$, $c = (c_1, \ldots, c_n)$.
- $b^T = (b_1, \ldots, b_m)$
- A is a $n \times m$ matrix.

Linear programming: problem

Given

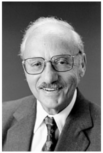
- $c = (c_1, \ldots, c_n)$,
- $b^T = (b_1, \ldots, b_m)$,
- and a $n \times m$ matrix A.

find $x = (x_1, \ldots, x_n) \geq 0$, so that

- $Ax \geq b$ and $c^T x$ is minimized.

Linear programming: algorithms

We can solve Linear Programming in polynomial time



- Simplex method: Dantzig in 1947
  (exponential time Klee and Minty 1972)
- Ellipsoid method: Khachiyan 1979 ($O(n^6)$)
- Interior-point method: Karmarkar 1984 ($O(n^3)$)
- Most used algorithm is still Simplex (fast on average).
- Many commercial LP solvers CPLEX and open source Gurobi

Integer programming

- An integer programming (IP) problem is a linear programming problem with the additional restriction that the values of the variables must be integers.
- A mixed integer programming (MIP) problem is a linear programming problem with the additional restriction that, the values of some variables must be integers.

- Many NPO problems can be easily expressed as IP or MIP problems
- IP is NP-hard

Max SAT as integer program

- Max Sat: Input a set of m clauses on n variables, find an assignment that maximizes the number of satisfied clauses.
- For a clause j, the set of variables that appear in C_j
  - positive is $P(j)$
  - negative is $N(j)$
- We consider $n + m$ integer variables,
  - $x_1, \dots, x_n$, one per each variable
  - $y_1, \dots, y_m$, one per each clause

  The variables will be restricted to have values in $\{0, 1\}$
  This is a simplification of saying that they must hold integer values and that all of them are ≤ 1.

Max SAT as integer program

Max SAT-IP

$$
\begin{aligned}
\max \quad & \sum_{j=1}^{m} y_j \\
\text{s.t.} \quad & \sum_{i \in P(j)} x_i + \sum_{i \in N(j)} (1 - x_i) \geq y_j \qquad 1 \leq j \leq m \\
& y_j \in \{0, 1\} \qquad 1 \leq j \leq m \\
& x_i \in \{0, 1\} \qquad 1 \leq i \leq n
\end{aligned}
$$

The size of the IP is polynomial in the size of the Max SAT, so the transformation is a polynomial Turing reduction from Max SAT to IP.

Vertex cover as integer program

VC

Given a graph $G = (V, E)$ we want to find a set $S \subset V$ with minimum cardinality, so that every edge in G has at least one end point in S.

VC-IP

$$
\begin{array}{lll}
\min & \displaystyle\sum_{i=1}^{n} x_i & \\
\text{s.t.} & x_i + x_j \geq 1 & \text{for all } (i,j) \in E \\
 & x_i \in \{0,1\} & \text{for all } i \in V
\end{array}
$$

Weighted Vertex cover as integer program

WVC

Given a graph $G = (V, A)$ with weights w associated to the vertices, we want to find a set $S \subset V$ with minimum weight, so that every edge in G has at least one end point in S.

VC-IP

$$\begin{array}{lll} \min & \sum_{i=1}^{n} w_i x_i & \\ \text{s.t.} & x_i + x_j \geq 1 & \text{for all } (i,j) \in E \\ & x_i \in \{0,1\} & \text{for all } i \in V \end{array}$$

Exercise

Try to write a LP or IP formulation for the problems

- Min Weighted Matching
- Set cover
- Max Flow

1. LP and IP
2. Relax and round
3. Primal-Dual

Relaxation and rounding

- Many real-life problems can be modeled as Integer Linear Programs (IP).
- The IP can be relaxed to a linear program (LP) by eliminating the integrity constraints.
- By doing so the optimum cost can only improve, i.e., opt of LP is better than opt of IP.
- We can solve the LP in polynomial time.
- The LP optimal solution might not be integral, when possible, transform it to get a feasible integer solution not far from opt of IP.

Vertex cover

VC

Given a graph $G = (V, A)$ we want to find a set $S \subset V$ with minimum cardinality, so that every edge in G has at least one end point in S.

VC-IP

$$\begin{array}{lll} \min & \sum_{i=1}^{n} x_i & \\ \text{s.t.} & x_i + x_j \geq 1 & \text{for all } (i,j) \in E \\ & x_i \in \{0,1\} & \text{for all } i \in V \end{array}$$

VC-LP

$$\begin{array}{lll} \min & \sum_{i=1}^{n} x_i & \\ \text{s.t.} & x_i + x_j \geq 1 & \text{for all} (i,j) \in E \\ & x_i \geq 0 & \text{for all } i \in V \end{array}$$

Vertex cover: another approximation algorithm

Lemma

VC-LP *has an optimal solution* x^* *such that* $x_i \in \{0, 1, 1/2\}$*.* *Furthermore, such a solution can be computed in polynomial time.*

Proof.

Let y be an optimal solution s.t. not all its coordinates are in $\{0, 1, 1/2\}$. Set $\epsilon = \min_{y_i \notin \{0,1,1/2\}} \{y_i, |y_i - 1/2|, 1 - y_i\}$. Consider

$$y'_i = \begin{cases} y_i - \epsilon & 0 < y_i < 1/2 \\ y_i + \epsilon & 1/2 < y_i < 1 \\ y_i & \text{otherwise} \end{cases} \qquad y''_i = \begin{cases} y_i + \epsilon & 0 < y_i < 1/2 \\ y_i - \epsilon & 1/2 < y_i < 1 \\ y_i & \text{otherwise} \end{cases}$$

$\sum y_i = (\sum y'_i + \sum y''_i)/2$, so both are optimal solutions. One of them has more $\{0, 1, 1/2\}$ coordinates than y. ☺

Vertex cover

function RELAX+ROUND VC(G)
  Construct the LP-VC associated G
  Let y be an optimal relaxed solution (of the LP instance)
  Using the previous lemma, construct an optimal relaxed solution y' such that $y'_i \in \{0, 1, 1/2\}$
  Let x defined as $x_i = 0$ if $y'_i = 0$, $x_i = 1$ otherwise.
  return (x)

RELAX+ROUND VC

- runs in polynomial time
- x defines a vertex cover
- $\sum_{i=1}^{n} x_i \leq 2 \sum_{i=1}^{n} y'_i \leq 2\text{opt}$
- is a 2-approximation for VC.

Weighted vertex cover: Relax+Round approximation

LP WVC

$$\begin{array}{ll} \min & \sum_{i=1}^{n} w_i x_i \\ \text{s.t.} & x_i + x_j \geq 1 \quad \text{for all}(i,j) \in E \\ & x_i \geq 0 \quad \text{for all } i \in V \end{array}$$

function WVC(G, c)
  Construct the LP WVC, I
  $y = LP.solve(I)$
  for $i = 1, \dots, n$ **do**
    if $y_i < 1/2$ **then**
      $x_i = 0$
    else
      $x_i = 1$
  return (x)

RELAX+ROUND WVC

- runs in polynomial time
- x defines a vertex cover
- $\sum_{i=1}^{n} w_i x_i \leq 2 \sum_{i=1}^{n} w_i y_i \leq 2\text{opt}$
- is a 2-approximation for WVC.

Minimum 2-Satisfiability

MIN 2-SAT

Given a Boolean formula in 2-CNF, determine whether it is satisfiable and, in such a case, find a satisfying assignment with minimum number of true variables.

- 2-SAT can be solved in polynomial time.
- MIN 2-SAT is NP-hard.
- MIN 2-SAT IP formulation?

Minimum 2-Satisfiability: IP formulation

Suppose that F has n variables $x_1, \dots x_n$ and m clauses with 2 literals per clause

IP Min 2-SAT

$$\begin{array}{lrl} \min & \displaystyle\sum_{i=1}^{n} x_i & \\ \text{s.t.} & x_i + x_j \geq 1 & \text{for all clauses } (x_i \vee x_j) \in F \\ & (1 - x_i) + x_j \geq 1 & \text{for all clauses } (\overline{x}_i \vee x_j) \in F \\ & (1 - x_i) + (1 - x_j) \geq 1 & \text{for all clauses } (\overline{x}_i \vee \overline{x}_j) \in F \\ & x_i \in \{0, 1\} & 1 \leq i \leq n \end{array}$$

LP Min 2-SAT is obtaining replacing $x_i \in \{0, 1\}$ by $x_i \geq 0$.

Minimum 2-Satisfiability: LP relaxation

LP Min 2-SAT

$$
\begin{array}{llll}
 & \min & \displaystyle\sum_{i=1}^{n} x_i & \\
\text{s.t.} & & x_i + x_j \geq 1 & \text{for all clauses } (x_i \vee x_j) \in F \\
 & & (1 - x_i) + x_j \geq 1 & \text{for all clauses } (\overline{x}_i \vee x_j) \in F \\
 & & (1 - x_i) + (1 - x_j) \geq 1 & \text{for all clauses } (\overline{x}_i \vee \overline{x}_j) \in F \\
 & & x_i \geq 0 & 1 \leq i \leq n
\end{array}
$$

- Let y be an optimal solution to LP Min 2-SAT.
- Can we use the same rounding scheme as for WVC?
- Setting $x_i = 1$ if $y_i > 1/2$ and $x_i = 0$ if $y_i < 1/2$ is safe, all clauses with at least one literal with value $> 1/2$ will be satisfied.
- When $y_i = 1/2$?

Minimum 2-Satisfiability: LP relaxation

- Let y be an optimal solution to IP Min 2-SAT.
- What to do when $y_i = 1/2$? 1? 0?
- If F contains the clauses $(x_i \vee x_j)$ and $(\overline{x}_i \vee \overline{x}_j)$ and $y_i = y_j = 1/2$, neither $x_i = x_j = 1$ nor $x_i = x_j = 0$ satisfy the formula.
- F_1 = clauses whose two variables have y value $= 1/2$.
- Rounding those values to 1 or 0 would keep the approximation ratio to 2, provided the constructed solution x to MIN 2-SAT is still a satisfying assignment.
- Any satisfying assignment for the clauses in F_1 and get a 2-approximation ☺

Minimum 2-Satisfiability: Relax+Round approximation

function RELAX+ROUND MIN 2-SAT(F)
  if F is not satisfiable **then return** false
  Construct the LP Min 2-SAT, I
  $y = LP.solve(I)$
  for $i = 1, \dots, n$ **do**
    if $y'_i < 1/2$ **then** $x_i = 0$
    if $y'_i > 1/2$ **then** $x_i = 1$
  F_1 = clauses with both y values $= 1/2$.
  Let $J = \{j \mid x_j \in F_1\}$
  for i=1,..., n **do**
    if $y_i = 1/2$ and $i \notin J$ **then** $x_i = 1$
  Complete x with a satisfying assignment for F_1
  return (x)

Minimum 2-Satisfiability: Relax+Round approximation

Theorem

RELAX+ROUND MIN 2-SAT *is a 2-approximation for* MIN 2-SAT.

Max Satisfiability

MAX SAT

Given a Boolean formula in CNF and weights for each clause, find a Boolean assignment to maximize the weight of the satisfied clauses.

Suppose that F has n variables $x_1, \ldots x_n$ and m clauses $C_1, \ldots, C_m$.

IP Max SAT

$$
\begin{array}{lll}
 & \max & \sum_{j=1}^{m} w_j z_j \\
\text{s.t.} & \sum_{x_i \in C_j} y_i + \sum_{\overline{x}_i \in C_j} (1 - y_i) \geq z_j & \jmath = 1, \ldots, m \\
 & y_i \in \{0,1\} & 1 \leq i \leq n \\
 & z_j \in \{0,1\} & 1 \leq j \leq m
\end{array}
$$

LP Max SAT is obtaining replacing $a \in \{0,1\}$ by $0 \leq a \leq 1$.

Max Satisfiability: Relax+RRound

function RELAX+RROUND(F)
  Construct the LP Max SAT, I
  $(y, z) = LP.solve(I)$
  for i=1,. . . , n **do**
    Set $x_i = 1$ with probability y_i
  return (x)

- The optimal LP solution is used as an indicator of the probability that the variable has to been set to 1.
- The performance of a randomized algorithm is the expected number of satisfiable clause.
- This expectation has to be compared with opt.

Max Satisfiability: Relax+RRound

- Let (y^*, z^*) be an optimal solution of LP Max SAT
- Let Z_j be the indicator random variable for the event that clause C_j is satisfied.
- Assume that C_j has k-literals and that ℓ of them are negated variables.

Lemma

For any $1 \leq j \leq m$, $E[Z_j] \geq z_j^*(1 - 1/e)$.

Recall $(a_1 \dots a_k)^{1/k} \leq (a_1 + \cdots + a_k)/k$ or equivalently $(a_1 \dots a_k) \leq ((a_1 + \cdots + a_k)/k)^k$

Max Satisfiability: Relax+RRound

Proof.

Z_j is an indicator random variable, and so
$E[Z_j] = Pr[Z_j = 1] = 1 - Pr[Z_j = 0]$

$$Pr[Z_j = 0] = \prod_{x_i \in C_j} (1 - y_i^*) \cdot \prod_{\overline{x}_i \in C_j} y_i^* \leq \left(\frac{(k - \ell) - \sum_{x_i \in C_j} y_i^* + \sum_{\overline{x}_i \in C_j} y_i^*}{k} \right)^k$$

$$\leq \left(\frac{(k - \sum_{x_i \in C_j} y_i^* - \sum_{\overline{x}_i \in C_j} (1 - y_i^*))}{k} \right)^k \leq \left(\frac{(k - z_j^*)}{k} \right)^k \leq \left(1 - \frac{z_j^*}{k} \right)^k$$

$$E[Z_j] \geq 1 - \left(1 - \frac{z_j^*}{k} \right)^k \geq z_j^* \left(1 - \frac{1}{k} \right)^k \geq z_j^* (1 - 1/e)$$

☺

Max Satisfiability: Relax+RRound approximation

Theorem

RELAX+RROUND *is a* $e/(e-1)$*-approximation for* MAX SAT.

Proof.

- Let (y^*, z^*) be an optimal solution of LP Max SAT
- Let Z_j be the indicator r.v.a for clause C_j is satisfied.
- Let W be the r.v. weight of satisfied clauses: $W = \sum_{j=1}^{m} w_j Z_j$.
- $E[W] = \sum_{j=1}^{m} w_j E[Z_j] \geq (1 - 1/e) \sum_{j=1}^{m} w_j z_j^* \geq (1 - 1/e)\text{opt}$

☺

Max Satisfiability:RandAssign

function RANDASSIGN(F)
  for i=1,. . . , n **do**
    Set $x_i = 1$ with probability 1/2
  return (x)

Theorem

RANDASSIGN *is a 2-approximation for* MAX SAT.

Proof.

$E[W] = \sum_{j=1}^{m} w_j E[Z_j] = \sum_{j=1}^{m} w_j \left(1 - \left(\frac{1}{2}\right)^{k_j}\right) \geq \frac{1}{2} \sum_{j=1}^{m} w_j \geq \frac{1}{2}\text{opt}.$ ☺

We move from $r = 2$ (RANDASSIGN) to $r = 1.581977$ (RELAX+RROUND).

Max Satisfiability:Best2

function $\textsc{Best2}(F)$
    $x_1, W_1 = \textsc{RandAssign}(F)$
    $x_2, W_2 = \textsc{Relax+RRound}(F)$
    if $W_1 \geq W_2$ **then**
        return (x_1)
    else
        return (x_2)

Theorem

$\textsc{Best2}$ *is a* $4/3$ *(1.33333)-approximation for* $\textsc{Max SAT}$.

Max Satisfiability:Best2

Proof.

- $E[W] = E[\max\{W_1, W_2\}] \geq E[(W_1 + W_2)/2\}]$.

$$E[W] \geq \sum_{j=1}^{m} w_j \left[\frac{1}{2}\left(1 - \left(\frac{1}{2}\right)^{k_j}\right) + \frac{1}{2} z_j^* \left(1 - \left(\frac{1}{k_j}\right)^{k_j}\right)\right]$$

$$\geq \sum_{j=1}^{m} w_j \frac{3}{4} z_j^* \geq \frac{3}{4} \sum_{j=1}^{m} w_j z_j^* \geq \frac{3}{4}\text{opt}.$$

Max Satisfiability:Best2

Proof.

- Is $\left[\frac{1}{2}\left(1-\left(\frac{1}{2}\right)^{k_j}\right)+\frac{1}{2}z_j^*\left(1-\left(\frac{1}{k_j}\right)^{k_j}\right)\right] \geq \frac{3}{4}z_j^*$?
- $k_j = 1$: $\frac{1}{2}\frac{1}{2}+\frac{1}{2}z_j^* \geq \frac{3}{4}z_j^*$.
- $k_j = 2$: $\frac{1}{2}\frac{3}{4}+\frac{1}{2}\frac{3}{4}z_j^* \geq \frac{3}{4}z_j^*$.
- $k_j \geq 3$: the minimum possible of each term is

$$\frac{1}{2}\frac{7}{8}+\frac{1}{2}\left(1-\frac{1}{e}\right)z_j^* \geq \frac{3}{4}z_j^*$$

☺

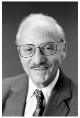
1. LP and IP
2. Relax and round
3. Primal-Dual

Primal Dual

- Many real-life problems can be modeled as Integer Linear Programs (IP).
- Since IPs are NP-hard to solve, they are often relaxed to a linear program (shortened as LP).
- Modus operandi: solve te linear program in polynomial time, and extract useful information about an integer optimum solution.
- However, for certain problems, we do not need to even solve the LP to get good (reasonable approximation factor) solutions to our problem using duality to control improvements.

History

- George Dantzig started linear programming (1947) , and his ideas contain the first germs of primal dual algorithms. The Hungarian method was an application of the paradigm.
- Jack R. Edmonds gave the first (sophisticated) application of the paradigm in his work on maximum weight matchings in arbitrary graphs (1965).
- Bar-Yehuda and Even first enunciated the paradigm in their work on the weighted Vertex Cover problem (1981).

Dantzig  Edmonds 

Primal, Dual and Weak Duality

Consider a LP in n variables $x = (x_1, \ldots, x_n)$ with m constraints represented by matrix A, independent terms b, and objective function b.

Primal

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax \geq b \\ & x \geq 0 \end{array}$$

The dual is an effort to construct the best lower bound for the primal objective function.

Searching for a lower bound: The best one?

LP (PRIMAL)

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b \\ & x \geq 0 \end{aligned}$$

if x^* opt, $y^T Ax$ is a general linear combination of equations, if we can select y so that
$y^T Ax^* = c^T x^*$,
$c^T x^* \geq y^T b$

The best lower bound, for any x?

$$\begin{aligned} \max \quad & b^T y \\ \text{s.t.} \quad & A^T y = c \\ & y \geq 0 \end{aligned}$$

But as we are maximizing this is equivalent to

$$\begin{aligned} \max \quad & b^T y \\ \text{s.t.} \quad & A^T y \leq c \qquad \text{DUAL} \\ & y \geq 0 \end{aligned}$$

Primal - Dual: an example

- Working from the dual trying to get the best lower bound we come back to the primal.
- Another example that you know is the pair MaxFlow-MinCut if you write the LP formulation of MaxFlow you can check that the dual is a LP formulation for MinCut

Strong and Weak duality theorem

There are additional conditions for a pair (x, y) of primal-dual optimal/feasible solutions.

Theorem (Strong duality)

If the primal has an optimal solution x^* then the dual has an optimal solution y^* such that $c^T x^* = b^T y^*$

Theorem (Weak Duality)

For every feasible solution x to the primal and every solution z to the dual,

$$\sum_{i=1}^{n} c_i x_i \geq \sum_{j=1}^{m} b_j z_j$$

Conditions for optimality: Complementary slackness

Let x be a feasible solution to the primal and let z be a feasible solution to the dual.

Primal complementary slackness

If $x_i > 0$, then $\sum_{j=1}^{m} a_{ij} z_j = c_i$.

Dual complementary slackness

If $z_j > 0$, then $\sum_{i=1}^{n} a_{ij} x_i = b_j$.

Conditions for optimality: Complementary slackness

Theorem

If (x, y) satisfies complementary slackness, then x and y are optimal solutions for primal and dual problems, respectively.

Proof.

$$\sum_{i=1}^{n} c_i x_i = \sum_{i=1}^{n} (\sum_{j=1}^{m} a_{ij} z_j) x_i = \sum_{j=1}^{m} (\sum_{i=1}^{n} a_{ij} x_i) z_j = \sum_{j=1}^{m} b_j z_j$$

☺

Relaxed complementary slackness

Let x be a feasible solution to the primal and let z be a feasible solution to the dual.

Primal relaxed complementary slackness

If $x_i > 0$, then $\sum_{j=1}^{m} a_{ij} z_j \geq c_i/\alpha$.

Dual relaxed complementary slackness

If $z_j > 0$, then $\sum_{i=1}^{n} a_{ij} x_i \leq \beta \, b_j$.

for some factors $\alpha, \beta \geq 1$

If x is integral and primal and dual relaxed complementary slackness hold?

Relaxed complementary slackness

Theorem

Let Π be a minimization integer program and Π-LP its LP-relaxation. Suppose a primal (integer) feasible solution x of Π and a dual feasible solution y of Π-LP satisfy the primal-dual relaxed complementary slackness, for some $\alpha, \beta > 1$, and x is integral, then x is a $\alpha\beta$-approximation.

Relaxed complementary slackness

Proof.

$$\sum_{i=1}^{n} c_i x_i \leq \alpha \sum_{i=1}^{n} (\sum_{j=1}^{m} a_{ij} z_j) x_i = \alpha \sum_{j=1}^{m} (\sum_{i=1}^{n} a_{ij} x_i) z_j \leq \alpha\beta \sum_{j=1}^{m} b_j z_j$$

By weak duality $\sum_{j=1}^{m} b_j z_j \leq \sum_{i=1}^{n} c_i x'_i$ for any feasible x', in particular for the optimal solution of the IP, therefore

$$\sum_{i=1}^{n} c_i x_i \leq \alpha\beta \sum_{j=1}^{m} b_j z_j \leq \alpha\beta \,\text{opt}$$

☺

Primal dual algorithms

- Primal-Dual algorithms iterate obtaining primal/dual feasible solutions by increasing values of variables until a restriction is tight (fulfilled with equality).
- If at some point objective functions match, we have found an optimal solution.
- If at some point complementary slackness holds for some r, we have found a r-approximate solution.

Primal-Dual for vertex cover

VC

Given a graph $G = (V, E)$, we want to find a set S, with minimum number of vertices, so that every edge in G has at least one end point in S.

- We know how to formulate VC as an IP problem
- We know how to relax the IP formulation as LP problem
- We know how to compute the dual of the LP problem

Vertex cover: LP relaxation

IP	LP
$\min \sum_{i=1}^{n} x_i$	$\min \sum_{i=1}^{n} x_i$
s.t. $x_i + x_j \geq 1$ for all $(i,j) \in E$	s.t. $x_i + x_j \geq 1$ for all $(i,j) \in E$
$x_i \in \{0,1\}$ for all $i \in V$	$x_i \geq 0$ for all $i \in V$

- Let opt be the size of an optimal solution of the VC instance.
- Let x^* be an optimal solution of the LP and $s^* = \sum_{i=1}^{n} x_i^*$.
- $s^* \leq \text{opt}$

Vertex cover: Primal-Dual approximation

LP primal

$$\begin{aligned} \min \quad & \sum_{i=1}^{n} x_i \\ \text{s.t.} \quad & x_i + x_j \geq 1 \quad e = (i,j) \in E \\ & x_i \geq 0 \quad i \in V \end{aligned}$$

LP dual

$$\begin{aligned} \max \quad & \sum_{e \in E} z_e \\ \text{s.t.} \quad & \sum_{i \in e} z_e \leq 1 \quad \text{for all } i \in V \\ & z_e \geq 0 \quad \text{for all } e \in E \end{aligned}$$

- Start with the integer infeasible primal solution $x = 0$, and the dual feasible solution $z = 0$.
- Repeat while some constraint in primal is unsatisfied:
  - Increase all (unfrozen) variables z_e until some dual constraint becomes tight (say, for vertex i).
  - Set $x_i = 1$. Freeze all the variables z_e such that $i \in e$.

Vertex cover: Primal-Dual approximation

- When the process stops, we have increased the variables z_e suitably.
- Some vertices i were chosen ($x_i = 1$)
- This set S of vertices is our output.
- Is S a vertex cover?
  Otherwise, we would have continued as some primal constraint were still unsatisfied.
- Cost of the solution?
  At the end of the algorithm x, z are feasible. Relaxed complementary slackness?.

Primal-Dual approximation: relaxed complementary conditions

Two conditions hold:

- Primal:
  If $x_i > 0$, we have frozen $x_i = 1$ at some step,
  then $\sum_{i \in e} z_e = 1$.
- Dual:
  If $z_e > 0$, z_e is increased. We do not know if it is because for one or both endpoints the constraints got tight, but
  $x_i + x_j \leq 2 \leq 2c_i$, for $e = (i, j)$.
- So, relaxed complementary slackness conditions hold for $r = 2$. A 2-approximation ☺.

Primal-Dual for weighted vertex cover

WVC

Given a vertex weighted graph $G = (V, A, c)$ we want to find a set $S \subset V$ with minimum weight, so that every edge in G has at least one end point in S.

- The problem is NP-hard and belongs to NPO.
- Can we formulate WVC as an IP problem?
- Variables: $x_1 \dots x_n$, $x_i = 1$ iff $i \in S$.
- Objective functon: $\sum_{i=1}^{n} c_i x_i$.
- Restrictions: for every edge $(i,j) \in E$, $x_i + x_j \geq 1$
- $x_i \in \{0, 1\}$
- The IP can be computed in polytime.

Weighted vertex cover: LP relaxation

IP

$$\begin{array}{lll} \min & \sum_{i=1}^{n} c_i x_i & \\ \text{s.t.} & x_i + x_j \geq 1 & \text{for all } (i,j) \in E \\ & x_i \in \{0,1\} & \text{for all } i \in V \end{array}$$

LP

$$\begin{array}{lll} \min & \sum_{i=1}^{n} c_i x_i & \\ \text{s.t.} & x_i + x_j \geq 1 & \text{for all} (i,j) \in E \\ & x_i \geq 0 & \text{for all } i \in V \end{array}$$

Weighted vertex cover: Primal-Dual approximation

LP primal

$$\begin{aligned} \min \quad & \sum_{i=1}^{n} c_i x_i \\ \text{s.t.} \quad & x_i + x_j \geq 1 \quad \text{for all} (i,j) \in E \\ & x_i \geq 0 \quad \text{for all } i \in V \end{aligned}$$

LP dual

$$\begin{aligned} \max \quad & \sum_{e \in E} z_e \\ \text{s.t.} \quad & \sum_{i \in e} z_e \leq c_i \quad \text{for all } i \in V \\ & z_e \geq 0 \quad \text{for all } e \in E \end{aligned}$$

- Start with the integer infeasible primal solution $x = 0$, and the dual feasible solution $z = 0$.
- Repeat while some constraint in primal is unsatisfied:
    - Increase all (unfrozen) variables z_e until some dual constraint becomes tight (say, for vertex i).
    - Set $x_i = 1$. Freeze all the variables z_e such that $i \in e$.

Weighted vertex cover: Primal-Dual approximation

- When the process stops, we have increased the variables z_e suitably.
- Some vertices i were chosen ($x_i = 1$)
- This set S of vertices is our output and again is a vertex cover.
- Cost of the solution? x, z are feasible. Relaxed complementary slackness conditions?

Primal-Dual approximation: relaxed complementary conditions

Two conditions hold:

- Primal:
  If $x_i > 0$, we have frozen $x_i = 1$ at some step,
  then $\sum_{i \in e} z_e = c_i$.
- Dual:
  If $z_e > 0$, z_e is increased. We do not know if it is because for one or both endpoints the constraints got tight, but
  $x_i + x_j \leq 2 \leq 2c_i$, for $e = (i, j)$.
- So, relaxed complementary conditions hold for $r = 2$ and we have a 2-approximation for WVC.

Primal-Dual approximation: generalizing the approach

- In the algorithm, we increased the (active) dual variables simultaneously.
- Trying to get the highest (the best) lower bound that we can get for the primal minimization objective.
  In general, this step can be implemented solving another LP program!
- We can also increase edge variables one by one. This leads to another primal-dual approximation algorithm PRICING METHOD

Pricing method: another view of Primal-Dual

- Each edge must be covered by some vertex.
- Edge $e = (i, j)$ pays price $z_e \geq 0$ to use both vertex i and j.
- Fairness: Edges incident to vertex i should pay $\leq c_i$ in total.
- Prices z_e are fair if, for any vertex cover S, $\sum_e z_e \leq w(S)$.

- A vertex is tight with respect to a pricing z if $\sum_{i \in e} z_e = c_i$.

Pricing algorithm

Set prices and find vertex cover simultaneously.

function $\textsc{Pricing WVC}(G, c)$
  $S = \emptyset$;
  for $e \in E$ **do**
    $z[e] = 0$ % initial price is 0
  while there is $(i, j) \in E$ so that neither i nor j is tight **do**
    select such an edge $e = (i, j)$
    Increase $z[e]$ until i or j became tight.
    Add to S the vertex (vertices) that became tight.
  return S

Pricing algorithm

Theorem

PRICING WVC *is a 2-approximation for* WVC.

- Follows directly from primal-dual arguments.
- However, PRICING WVC is a greedy algorithm.
- No LP solver has been used!