Integer / linear programming

“Programming”

- A class of mathematical techniques (e.g. dynamic programming) to solve the general optimization problems as

$$Max\ f(x), \qquad x \in S \subseteq R^n$$

R^n: the set of *n*-dimensional vectors of real numbers

f(x): objective function, a real-valued function defined on *S*

S the *constraint set*.

- By choosing f and S appropriately, we can model a wide variety of real-life problems in this way.
- Hard problems: $|S| \sim O(k^n)$, k > 1

Feasibility and optimality

- Any $x \in S$ is called a *feasible solution*
- If the there is an $x_0 \in S$ such that $f(x) \leq f(x_0)$ for all $x \in S$ then x_0 is called an *optimal solution*

- The aim is to find an optimal solution for a given f and S
- S can be defined by *constraints*

Example

MAX: $350X_1 + 300X_2$ → f

S.T.: $X_1 + X_2 \leq 200$

$9X_1 + 6X_2 \leq 1566$ → constraints

$12X_1 + 16X_2 \leq 2880$

$X_1, X_2 \geq 0$

X_1, X_2 must be integers → integer programming

Integer (linear) programming

- *f* & *S* are restricted by linear form (functions)
  - Linear programming
- *S* is restricted to have only integer values
  - Integer programming (IP), often referred to as integer linear programming (ILP)
- mixed integer programming problem: some elements of *S* are restricted to integers
- ILP is often harder than the corresponding LP problem

Linear programming

- $f(x) = c^{\mathrm{T}}x$, $S = \{\ x \mid \boldsymbol{A}x = \boldsymbol{b},\ x \geq 0\ \}$
  - c is an n x 1 vector, A is an m x n matrix and b is an $m \times 1$ vector
- For general x, these problems can be solved exactly (e.g. simplex technique). For integer x, the problem is *NP*-complete.

Inequality

- Inequality constraints can easily be introduced by adding an extra variable
  - max $2x_1 + 3x_2$ subject to $x_1 + x_2 \leq 10$ is equivalent to max $2x_1 + 3x_2$ subject to $x_1 + x_2 + x_3 = 10$
  - For "≥", we would insert ($-x3$) into the constraint
  - The extra variable is called a slack variable – it does not appear in the objective function. Because this is so straight-forward, many ILP solving programs allow you to express constraints with inequality directly.

Example: capital budgeting

- A firm has n projects that it would like to undertake, but due to budget limitations, not all can be selected. In particular, project j has a value of c_j, and requires an investment of a_{ij} in the time period i, $i = 1,\ldots,m$. The capital available in time period i is b_i.
- Objective: maximize the total value, subject to given budget constraints

Example: capital budgeting

A set of variables x_j, which we interpret as:

- $x_j = 1$, project j is selected
- $x_j = 0$, project j is not selected

Then the objective function can be formulated as

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

The constraints are

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i,\ i = 1,\ldots,m; \quad x_j \leq 1,\ j = 1,\ldots,n$$

Linear programming problems

maximize	**z =**	$-4x_1 + x_2 - x_3 + x_4$
subject to		$-7x_1 + 5x_2 + x_3 + x_4 = 8$
		$-2x_1 + 4x_2 + 2x_3 - x_4 = 10$
		$x_1, x_2, x_3, x_4 \geq 0$

We will describe LPs that start with the following (canonical) form:

- **equality constraints**
- **nonnegative (right hand side, RHS) variables**

Fundamental theorem

Extreme point (or Simplex filter) theorem:

If the maximum or minimum value of a linear function defined over a polygonal convex region exists, then it is to be found at the boundary of the region

Boundary points: basic feasible solutions

What does the extreme point theorem imply?

- A finite number of extreme points (bfs) implies a finite number of solutions!
- Hence, search is reduced to a finite set of points
- However, a finite set can still be too large for practical purposes
- **Simplex method provides an** efficient systematic search guaranteed to converge in a finite number of steps.

Basic feasible solutions

- Each corner point solution of the polyhedron is a basic feasible solution.
- The simplex method is a systematic way of moving from one basic feasible solution to another, always improving the solution, until the optimum solution is obtained.

Basic Feasible Solutions

maximize $\quad z = -4x_1 + x_2 - x_3 + x_4$

subject to

$$-7x_1 + 5x_2 + x_3 + x_4 = 8$$
$$-2x_1 + 4x_2 + 2x_3 - x_4 = 10$$
$$x_1, \quad x_2, \quad x_3, \quad x_4 \geq 0$$

Suppose there are m constraints, n variables

A basic solution is found by setting n-m variables to 0 and solving the remaining system with m variables and m constraints.

- **The n – m variables are called non-basic variables**
- **The m variables are called basic variables**

Basic feasible solutions

maximize $z = -4x_1 + x_2 - x_3 + x_4$

subject to

$$-7x_1 + 5x_2 + x_3 + x_4 = 8$$

$$-2x_1 + 4x_2 + 2x_3 - x_4 = 10$$

$$x_1, \quad x_2, \quad x_3, \quad x_4 \geq 0$$

$-z$	x_1	x_2	x_3	x_4		
1	-4	1	-1	1	=	0
0	-7	5	1	1	=	8
0	-2	4	2	-1	=	10

Basic feasible solutions

-z	x_1	x_2	x_3	x_4		
1	-11	6	0	2	=	8
0	-7	5	1	1	=	8
0	12	6	0	-3	=	-6

Example: Suppose we want the solution with basic variables x_3 and x_4, and thus x_1 and x_2 are non-basic.

We then perform pivot operations.

Basic Feasible Solutions

-z	x_1	x_2	x_3	x_4		
1	-3	2	0	0	=	4
0	-3	3	1	0	=	6
0	-4	2	0	1	=	2

Next pivot on the –3.

Basic Feasible Solutions

Canonical form: basic variables have a single one in the column.

-z	x_1	x_2	x_3	x_4		
1	-3	2	0	0	=	4
0	-3	3	1	0	=	6
0	-4	2	0	1	=	2

reduced costs →

The basic solution is found by setting non-basic variables to 0. We get $x_1=0$, $x_2=0$, $x_3=6$, $x_4=2$.

This solution also satisfies $x \geq 0$. It is called a basic feasible solution.

Simplex algorithm

	-z	x_1	x_2	x_3	x_4		
$z = -3x_1 + 2x_2 - 4$	1	-3	2	0	0	=	4
	0	-3	3	1	0	=	6
	0	-4	2	0	1	=	2

The entering variable for a max problem is a variable with positive reduced cost.

The pivot element is chosen uniquely in the column of the entering variable so that the next basis is feasible.

Simplex algorithm

The pivot element is chosen to leave the basis according to pivot rules (e.g. a min ratio rule).

$-z$	x_1	x_2	x_3	x_4		
1	1	0	0	-1	=	-2
0	3	0	1	-1.5	=	3
0	-2	1	0	.5	=	1

A pivot is carried out, leading to the next bfs.

Variable x_4 has left the basis.
The new basis consists of x_2 and x_3.

Simplex algorithm

$-z$	x_1	x_2	x_3	x_4		
1	0	0	-1/3	-1/2	=	-3
0	1	0	1/3	-1/2	=	1
0	0	1	2/3	-1/2	=	3

Optimality conditions for a maximization problem: all reduced costs are non-positive.

Simplex algorithm

Pivots are carried out until the bfs is optimal.

-z	x_1	x_2	x_3	x_4		
1	0	0	-1/3	-1/2	=	-3
0	1	0	1/3	-1/2	=	1
0	0	1	2/3	-1/2	=	3

$$z = -x_3/3 - x_4/2 + 3$$

This new bfs is optimal. Increasing x_3 or x_4 makes the solution worse.

Duality in linear programming

- Every primal problem there exists a corresponding dual problem
  - Primal Maximization → Dual Minimization
  - Primal Minimization → Dual Maximization

Duality in linear programming

- Primal has n choice variable and m constraints and dual has m variables and n constraints
- Right hand side elements (b_i) in the primal correspond to coefficient of the objective function of the dual
- The a_{ij} constraint coefficients become a_{ji} in dual

Primal and dual in matrix form

- Primal

  max $Z = c'x$

  s. t. $Ax \le r$

  $x \ge 0$

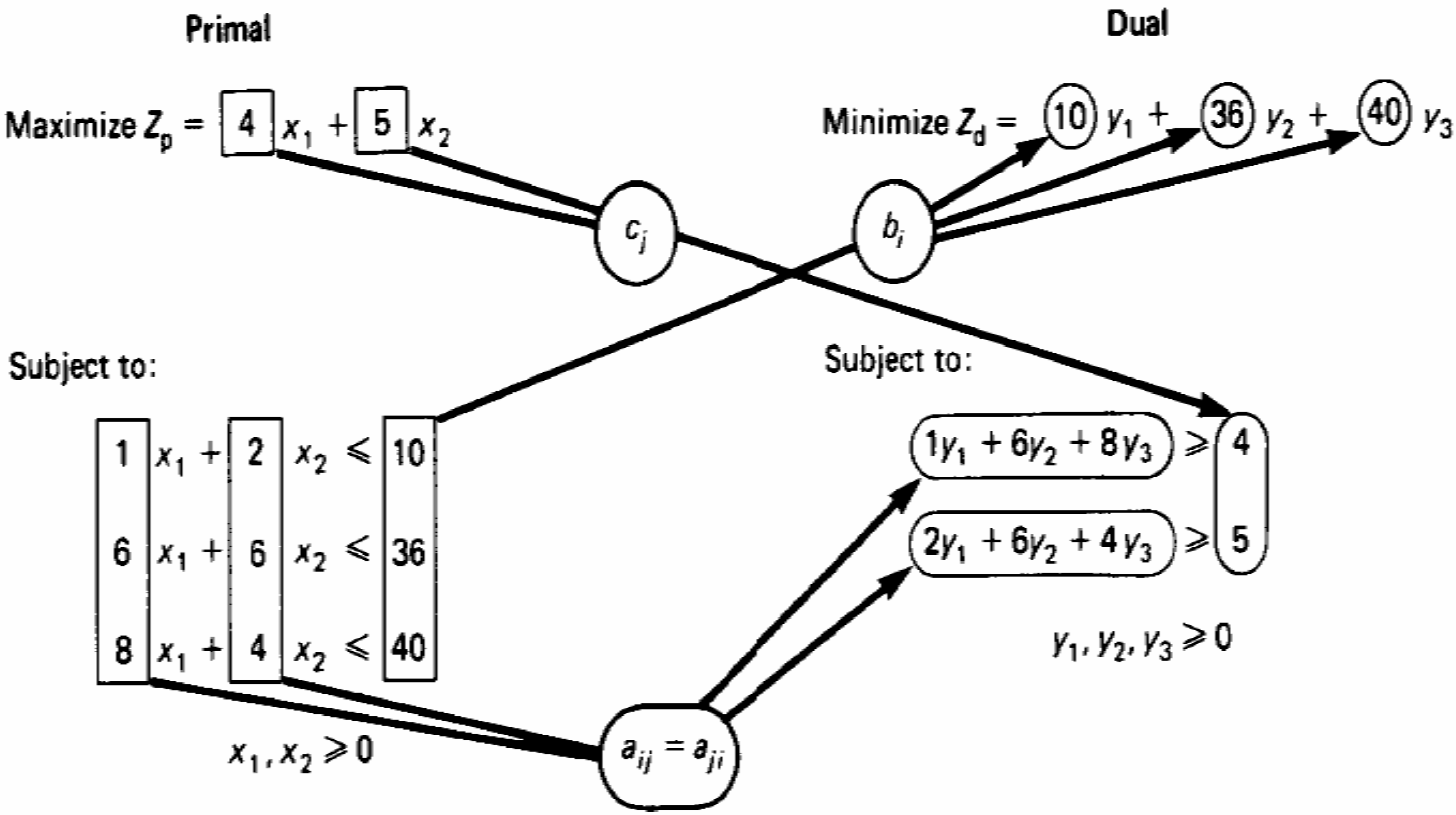
- Dual

  max $Z^* = r'y$

  s. t. $A'y \ge c$

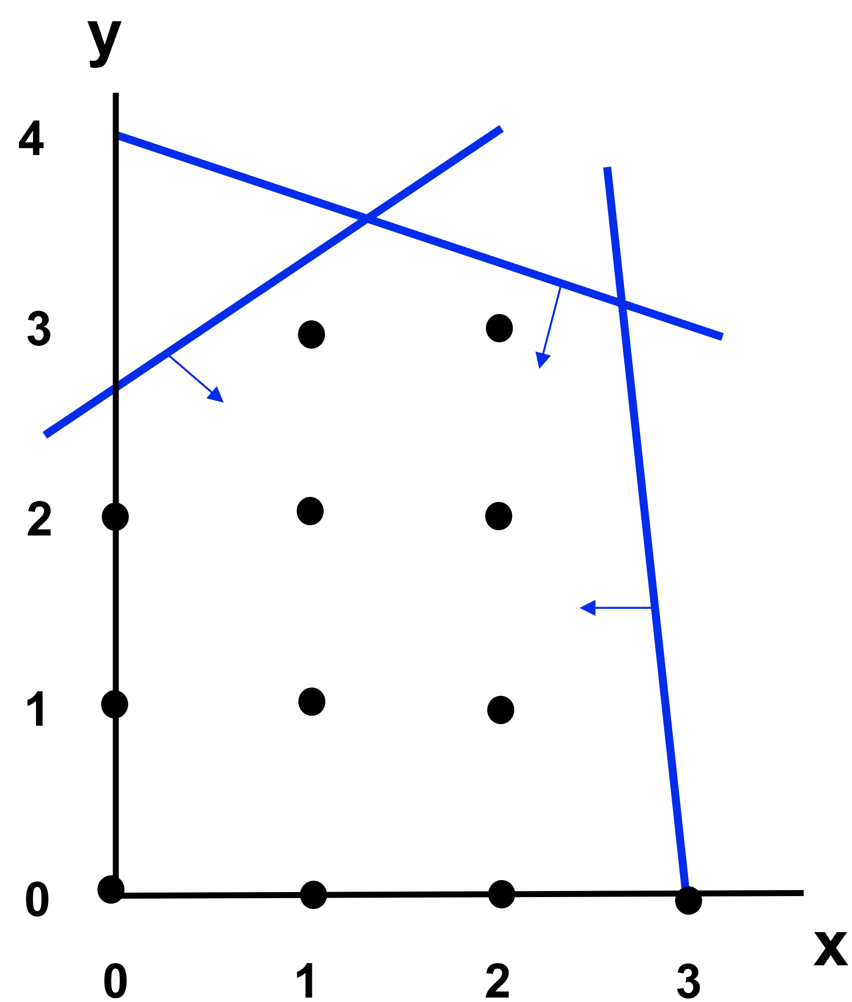
  $y \ge 0$

Primal and dual relationship



LP → Integer Programming

- Feasible region is a set of discrete points.
- Can't be assured a corner point solution.
- IP is NP-hard problem (By reduction from Satisfiability)
- Solving it as an LP provides a relaxation and a bound on the solution.



Duality theory

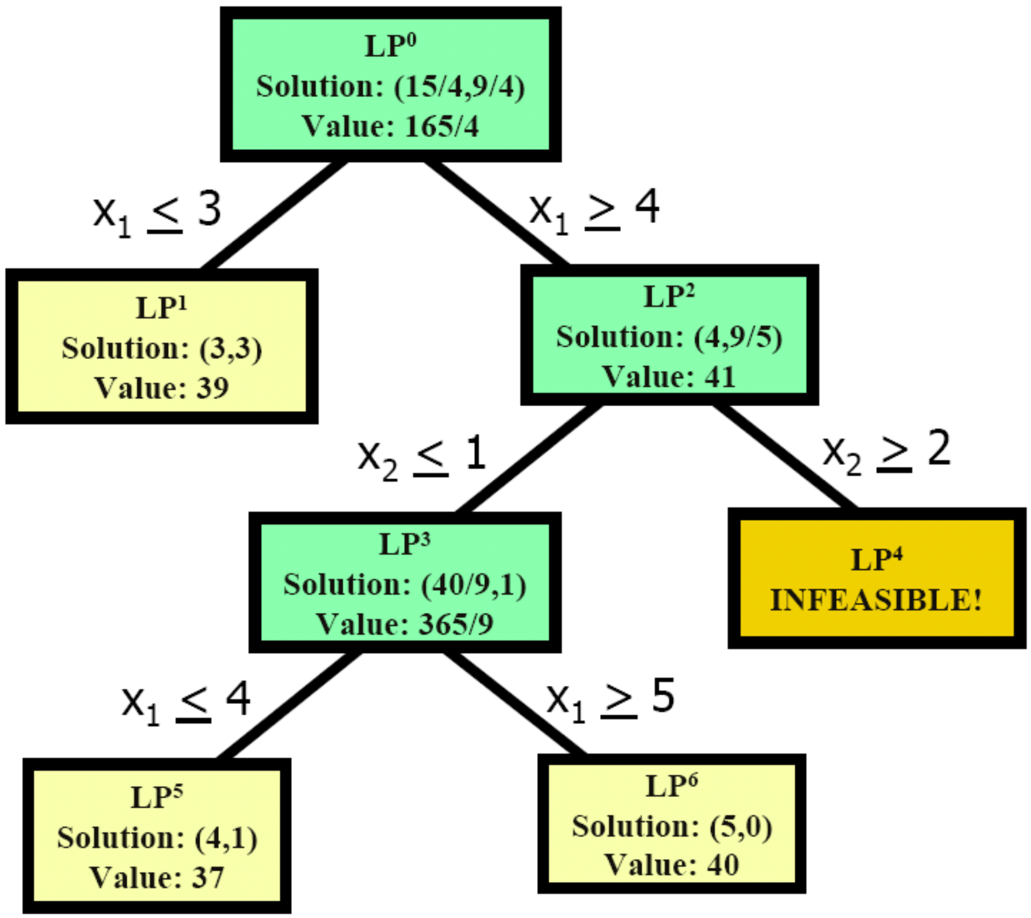
- (**Weak duality theorem**) the objective function value of the dual (*min*) at any feasible solution ≥ the objective function value of the primal (*max*) at any feasible solution.
- (**Strong duality theorem**) if the primal has an optimal solution, x^*, then the dual also has an optimal solution, y^*, such that $c' x^* = r' y^*$.

Solutions to IP

- Package
  - GNU Linear Programming Kit (GLPK)
    http://www.gnu.org/software/glpk/ → free
  - Cplex: mathematical optimizer
    http://www.ilog.com/products/cplex/ → commercial
- Enumeration
  - Small size of problems
- Branch and bound (tree-like search)
- Specially designed algorithms
  - LP relaxation

IP Solver = LP Solver + B&B

- Solve the LP relaxation of the IP
- If the LP solution has a non-integer variable, branch on that variable and solve the 2 resultant LPs
- Traverse the tree recursively until:
  - All terminals exposed (unsolvable LPs are terminal); or
  - A sub-LP solution is worse than an existing integer solutio



IP:

Maximize $8x_1 + 5x_2$

S.t. $x_1 + x_2 \leq 6$

$9x_1 + 5x_2 \leq 45$

$x_1, x_2 \in Z^+$

$\Downarrow$

LP Relaxation:

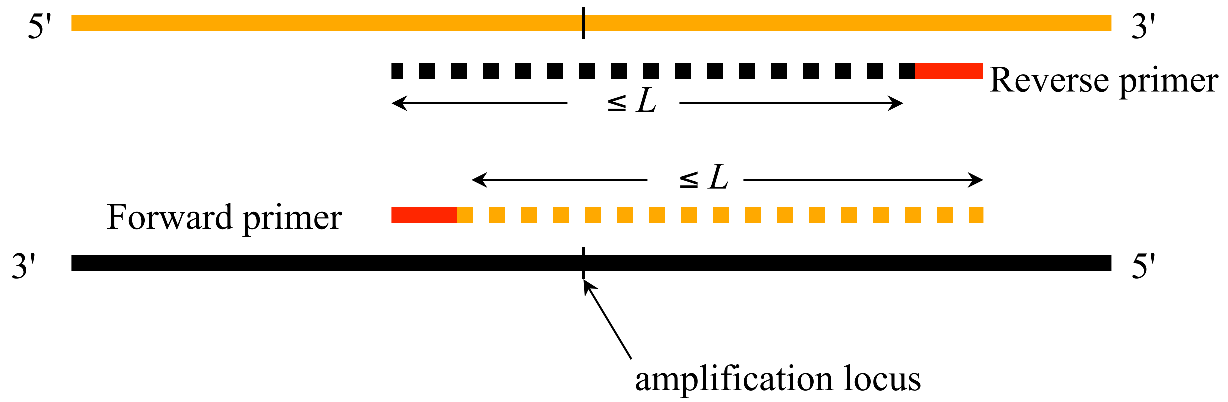
Maximize $8x_1 + 5x_2$

S.t. $x_1 + x_2 \leq 6$

$9x_1 + 5x_2 \leq 45$

$x_1, x_2 \geq 0$

Primer pair selection problem



- **Given:**
  - Genomic sequence around amplification locus
  - Primer length k
  - Amplification upper-bound L
- **Find:** Forward and reverse primers of length k that hybridize within a distance of L of each other and optimize amplification efficiency **(melting temperatures, secondary structure, cross hybridization, etc.)**

Primer set selection: multiplex experiment

- Spotted microarray synthesis [Fernandes and Skiena' 02]
  - Need unique pair for each amplification product, but primers can be re-used to minimize cost
  - Potential to reduce #primers from O(n) to O($n^{1/2}$) for n products

Primer set selection: application

- SNP Genotyping
  - Thousands of SNPs that must genotyped using hybridization based methods (e.g., SBE)
  - Selective PCR amplification needed to improve accuracy of detection steps (whole-genome amplification not appropriate)
  - No need for unique amplification!
  - Primer minimization is critical
    - Fewer primers to buy
    - Fewer multiplex PCR reactions

Primer set selection problem

- **Given:**
  - Genomic sequences around each amplification locus
  - Primer length k
  - Amplification upperbound L
- **Find:**
  - Minimum size set of primers S of length k such that, for each amplification locus, there are two primers in S hybridizing to the forward and reverse sequences within a distance of L of each other
  - Uniqueness constraint: S should contain a unique pair of primers amplifying each locus

Selection with uniqueness Constraints

- Can be modeled as minimum multicolored subgraph problem:
  - Vertices of the graph correspond to candidate primers
  - add edge colored by color *i* between two primers if they amplify i-th SNP and do not amplify any other SNP
  - Goal is to find minimum size set of vertices inducing edges of all colors
- NP-hard problem
- Trivial approximation algorithm: select 2 primers for each SNP
  - O($n^{1/2}$) approximation since at least $n^{1/2}$ primers required by every solution

Integer program formulation

- Variable x_u for every vertex (candidate primer) u
  - x_u set to 1 if u is selected, and to 0 otherwise
- Variable y_e for every edge e
  - y_e set to 1 if corresponding primer pair selected to amplify one of the SNPs
- Objective: minimize sum of x_u
- Constraints:
  - for each i, sum of $\{y_e : e$ amplifying SNP $i\} \geq 1$
  - $2y_e \leq x_u + x_v$ for every e incident to u & v