MVE165/MMG630, Applied Optimization
Lecture 8
Integer linear programming algorithms

Ann-Brith Strömberg

2009–04–15

Methods for ILP: Overview (Ch. 14.1)

- Enumeration
  - Implicit enumeration: Branch–and–bound
- Relaxations
  - Decomposition methods: Solve simpler problems repeatedly
  - Add valid inequalities to an LP – "cutting plane methods"
  - Lagrangian relaxation
- Heuristic algorithms – optimum *not* guaranteed
  - "Simple" rules $\Rightarrow$ feasible solutions
  - Construction heuristics
  - Local search heuristics

Relaxations and feasible solutions (Ch. 14.2)

- Consider a minimization integer linear program (ILP):

$$
\text{[ILP]} \qquad z^* = \begin{array}{rrcll} \min & \mathbf{c}^{\mathrm{T}}\mathbf{x} & & & \\ \text{subject to} & \mathbf{A}\mathbf{x} & \leq & \mathbf{b} & \\ & \mathbf{x} & \geq & \mathbf{0} & \text{and integer} \end{array}
$$

- The feasible set $X = \{\mathbf{x} \in Z_+^n \,|\, \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ is *non*-convex
- How prove that a solution $\mathbf{x}^* \in X$ is optimal?
- We cannot use strong duality/complementarity as for linear programming (where X is polyhedral $\Rightarrow$ convex)
- Bounds on the optimal value
  - Optimistic estimate $\underline{z} \leq z^*$ from a *relaxation* of ILP
  - Pessimistic estimate $\bar{z} \geq z^*$ from a *feasible solution* to ILP
- Goal: Find "good" feasible solution and tight bounds for z^*:

$$\bar{z} - \underline{z} \leq \varepsilon \text{ and } \varepsilon > 0 \text{ "small"}$$

Optimistic estimates of z^* from relaxations

- Either: Enlarge the set X by removing constraints
- Or: Replace $\mathbf{c}^{\mathrm{T}}\mathbf{x}$ by an underestimating function f, i.e., such that $f(\mathbf{x}) \leq \mathbf{c}^{\mathrm{T}}\mathbf{x}$ for all $\mathbf{x} \in X$
- Or: Do both

$\Rightarrow$ solve a *relaxation* of (ILP)

- Example (enlarge X):

  $X = \{\mathbf{x} \geq \mathbf{0} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b},\ \mathbf{x} \text{ integer } \}$ and
  $X^{\mathrm{LP}} = \{\mathbf{x} \geq \mathbf{0} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$

$$\Rightarrow \quad z^{\mathrm{LP}} = \min_{\mathbf{x} \in X^{\mathrm{LP}}} \mathbf{c}^{\mathrm{T}}\mathbf{x}$$

- It holds that $z^{\mathrm{LP}} \leq z^*$ since $X \subseteq X^{\mathrm{LP}}$

Relaxation principles that yield more tractable problems

- *Linear programming relaxation*
  Remove integrality requirements (enlarge X)

- *Combinatorial relaxation*
  E.g. remove subcycle constraints from asymmetric TSP $\Rightarrow$ min-cost assignment (enlarge X)

- *Lagrangean relaxation*
  Move "complicating" constraints to the objective function, with penalties for infeasible solutions; then find "optimal" penalties (enlarge X and find $f(\mathbf{x}) \leq \mathbf{c}^{\mathrm{T}}\mathbf{x}$)

Tight bounds

- Suppose that $\bar{\mathbf{x}} \in X$ is a feasible solution to ILP (min-problem) and that $\underline{\mathbf{x}}$ solves a relaxation of ILP
- Then
$$\underline{z} := \mathbf{c}^{\mathrm{T}} \underline{\mathbf{x}} \leq z^* \leq \mathbf{c}^{\mathrm{T}} \bar{\mathbf{x}} =: \bar{z}$$
- $\underline{z}$ is an *optimistic* estimate of z^*
- $\bar{z}$ is a *pessimistic* estimate of z^*
- If $\bar{z} - \underline{z} \leq \varepsilon$ then the value of the solution candidate $\bar{\mathbf{x}}$ is at most ε from the optimal value z^*
- Efficient solution methods for ILP combine relaxation and heuristic methods to find tight bounds (small $\varepsilon \geq 0$)

Branch–&–Bound algorithms (B&B) (Ch. 15)

$$[\text{ILP}] \qquad z^* = \min_{\mathbf{x}\in X} \; \mathbf{c}^{\mathrm{T}}\mathbf{x}, \qquad X \subset Z^n$$

- A general principle for finding *optimal* solutions to optimization problems with integrality requirements
- Can be adopted to different types of models
- Can be combined with other (e.g. heuristic) algorithms
- Also called implicit enumeration and tree search
- *Idea:* Enumerate all feasible solutions by a successive partitioning of X into a family of subsets
- Enumeration organized in a tree using graph search; it is made implicit by utilizing approximations of z^* from relaxations of [ILP] for cutting off branches of the tree

Branch–&–bound for ILP: Main concepts

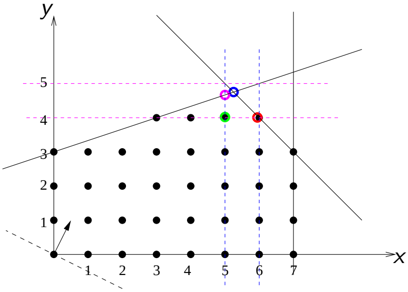
- Relaxation: a simplification of [ILP] in which some constraints are removed
  - Purpose: to get simple (polynomially solvable) (node) subproblems, and optimistic approximations of z^*.
  - Examples: remove integrality requirements, remove or Lagrangean relax complicating (linear) constraints (e.g. sub-tour constraints)
- Branching strategy: rules for partitioning a subset of X
  - Purpose: exclude the solution to a relaxation if it is not feasible in [ILP]; corresponds to a *partitioning* of the feasible set
  - Examples: Branch on fractional values, subtours, etc.

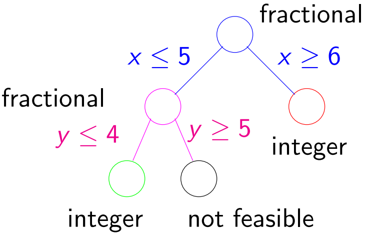
B&B: Main concepts (continued)

- Tree search strategy: defines the order in which the nodes in the B&B tree are created and searched
  - Purpose: quickly find good feasible solutions; limit the size of the tree
  - Examples: depth-, breadth-, best-first.
- Node cutting criteria: rules for deciding when a subset should not be further partitioned
  - Purpose: avoid searching parts of the tree that cannot contain an optimal solution
  - Cut off a node if the corresponding node subproblem has
    - no feasible solution, or
    - an optimal solution that is feasible in [ILP], or
    - an optimal objective value that is worse (higher) than that of any known feasible solution

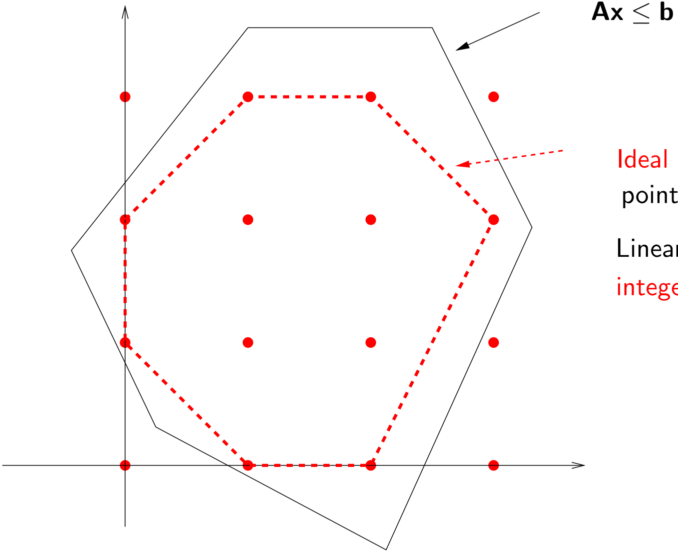
ILP: Solution by the branch–and–bound algorithm

- Relax integrality requirements ⇒ linear (continuous) problem
- B&B tree: branch over fractional variable values





Good and ideal formulations (Ch. 14.3)



Ideal since all extreme points are integral

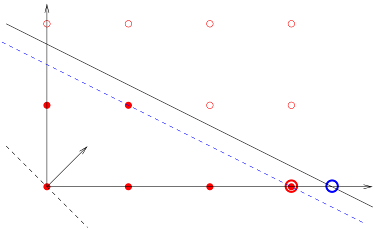
Linear program has integer extreme points

Cutting planes: A very small example

- Consider the following ILP:

$$\min\{-x_1 - x_2 : 2x_1 + 4x_2 \leq 7, x_1, x_2 \geq 0 \text{ and integer}\}$$

- ILP optimal solution: $z = -3$, $\mathbf{x} = (3, 0)$
- LP (continuous relaxation) optimum: $z = -3.5$, $\mathbf{x} = (3.5, 0)$
- Generate a simple cut:
  "Divide the constraint" by 2
  and round the RHS down
  $x_1 + 2x_2 \leq 3.5 \Rightarrow \color{blue}{x_1 + 2x_2 \leq 3}$
- Adding this cut to the continuous relaxation yields the optimal ILP solution

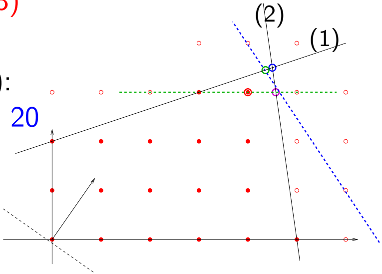


Cutting planes: valid inequalities (Ch. 14.4)

- Consider the ILP

$$\begin{array}{llcll}
\max & 7x_1 + 10x_2 & & & \\
\text{subject to} & -x_1 + 3x_2 & \leq & 6 & (1) \\
 & 7x_1 + x_2 & \leq & 35 & (2) \\
 & x_1, x_2 & \geq & 0, \text{ integer} &
\end{array}$$

- LP optimum: $z = 66.5$, $\mathbf{x} = (4.5, 3.5)$
- ILP optimum: $z = 58$, $\mathbf{x} = (4, 3)$
- Generate a VI by "adding" the two constraints (1) and (2): $6x_1 + 4x_2 \leq 41 \Rightarrow 3x_1 + 2x_2 \leq 20$ $\Rightarrow \mathbf{x} = (4.36, 3.45)$
- Generate a VI by "7·(1)+(2)": $22x_2 <= 77 \Rightarrow x_2 \leq 3$ $\Rightarrow \mathbf{x} = (4.57, 3)$



Cutting plane algorithms (iterativley better approximations of the convex hull) (Ch. 14.5)

- Choose a suitable mathematical formulation of the problem

1. Solve the linear programming (LP) relaxation
2. If the solution is integer, Stop. An optimal solution is found
3. Add one or several *valid inequalities* that cut off the fractional solution *but none of the integer solutions*
4. Resolve the new problem and go to step 2.

- *Remark:* An inequality in higher dimensions defines a *hyper-plane*; therefore the name cutting *plane*

About cutting plane algorithms

- Problem: It may be necessary to generate VERY MANY cuts
- Each cut should also pass through at least one integer point
  $\Rightarrow$ faster convergence
- Methods for generating valid inequalities
  - Chvatal-Gomory cuts (combine constraints, make beneficial roundings of LHS and RHS)
  - Gomory's method: generate cuts from an optimal simplex basis (Ch. 14.5.1)
- Pure cutting plane algorithms are usually less efficient than branch–&–bound
- In commercial solvers (e.g. CPLEX), cuts are used to help (presolve) the branch–&–bound algorithm
- For problems with specific structures (e.g. TSP and set covering) problem specific classes of cuts are used

Lagrangian relaxation ($\Rightarrow$ optimistic estimates of z^*) (Ch. 17.1–17.2)

- Consider a minimization integer linear program (ILP):

$$
\begin{array}{llrrcll}
\text{[ILP]} & z^* = & \min & \mathbf{c}^{\mathrm{T}}\mathbf{x} & & & \\
& & \text{subject to} & \mathbf{A}\mathbf{x} & \leq & \mathbf{b} & (1) \\
& & & \mathbf{D}\mathbf{x} & \leq & \mathbf{d} & (2) \\
& & & \mathbf{x} & \geq & \mathbf{0} \text{ and integer} &
\end{array}
$$

- Assume that the constraints (1) are complicating (subtour eliminating constraints for TSP, e.g.)
- Define the set $X = \{\mathbf{x} \in Z_+^n \mid \mathbf{D}\mathbf{x} \leq \mathbf{d}\}$
- Remove the constraints (1) and add them—with penalty parameters $\mathbf{v}$—to the objective function

$$h(\mathbf{v}) = \min_{\mathbf{x} \in X} \left\{ \mathbf{c}^{\mathrm{T}}\mathbf{x} + \mathbf{v}^{\mathrm{T}}(\mathbf{A}\mathbf{x} - \mathbf{b}) \right\} \qquad (3)$$

Weak duality of Lagrangian relaxations

Theorem: For any $\mathbf{v} \geq \mathbf{0}$ it holds that $h(\mathbf{v}) \leq z^*$.

Proof: Let $\bar{\mathbf{x}}$ be feasible in [ILP] $\Rightarrow \bar{\mathbf{x}} \in X$ and $\mathbf{A}\bar{\mathbf{x}} \leq \mathbf{b}$. It then holds that

$$h(\mathbf{v}) = \min_{\mathbf{x} \in X} \left\{ \mathbf{c}^{\mathrm{T}}\mathbf{x} + \mathbf{v}^{\mathrm{T}}(\mathbf{A}\mathbf{x} - \mathbf{b}) \right\} \leq \mathbf{c}^{\mathrm{T}}\bar{\mathbf{x}} + \mathbf{v}^{\mathrm{T}}(\mathbf{A}\bar{\mathbf{x}} - \mathbf{b}) \leq \mathbf{c}^{\mathrm{T}}\bar{\mathbf{x}}.$$

Since an optimal solution $\mathbf{x}^*$ to [ILP] is also feasible, it holds that

$$h(\mathbf{v}) \leq \mathbf{c}^{\mathrm{T}}\mathbf{x}^* = z^*.$$

□

$\Rightarrow$ $h(\mathbf{v})$ is a *lower bound* on the optimal value z^* for any $\mathbf{v} \geq \mathbf{0}$

- The best lower bound is given by

$$h^* = \max_{\mathbf{v} \geq \mathbf{0}} h(\mathbf{v}) = \max_{\mathbf{v} \geq \mathbf{0}} \left\{ \min_{\mathbf{x} \in X} \left\{ \mathbf{c}^{\mathrm{T}}\mathbf{x} + \mathbf{v}^{\mathrm{T}}(\mathbf{A}\mathbf{x} - \mathbf{b}) \right\} \right\}$$

Tractable Lagrangian relaxations

- Special algorithms for minimizing the Lagrangian dual function *h* exist (e.g., subgradient optimization, Ch. 17.3)
- *h* is always concave but typically nondifferentiable
- For each value of $\mathbf{v}$ chosen, a *subproblem* (3) must be solved
- For general ILP's: typically a non-zero duality gap $h^* < z^*$
- The Lagrangian relaxation bound is never worse that the linear programming relaxation bound, i.e. $z^{\mathrm{LP}} \leq h^* \leq z^*$
- If the set X has the integrality property (i.e., X^{LP} has integral extreme points) then $h^* = z^{\mathrm{LP}}$
- Choose the constraints ($\mathbf{Ax} \leq \mathbf{b}$) to dualize such that the relaxed problem (3) is computationally tractable but still does not possess the integrality property

An ILP Example

Find optimistic and pessimistic bounds for the following ILP example using the branch–&–bound algorithm, a cutting plane algorithm, and Lagrangean relaxation.

$$
\begin{array}{lrcl}
\max & 5x_1 + 4x_2 & & \\
\text{s.t.} & x_1 + x_2 & \leq & 5 \\
 & 10x_1 + 6x_2 & \leq & 45 \\
 & x_1, x_2 & \geq & 0 \text{ and integer}
\end{array}
$$

The linear programming optimal solution is given by $z = 23.75$, $x_1 = 3.75$ and $x_2 = 1.25$

Heuristic algorithms (Ch. 16)

- Constructive heuristics (Ch. 16.3): Start by an "empty set" and "add" elements according to some (simple) rule.
  - Difficult to guarantee that a feasible solution will be found
  - No measure of how close to a global optimum a solution is
  - Special rules for structured problems
  - E.g. the greedy algorithm (yields optimal spanning tree) is a constructive heuristic
  - For TSP: nearest neighbour, cheapest insertion, farthest insertion, etc. (Ch. 16.3)
- Local search heuristics (Ch. 16.4): Start from a feasible solution, which is iteratively improved by limited modification
  - Finds a local optimum
  - No measure of how close to a global optimum a solution is
  - Specialized for structured problems, but also general (Ch. 16.2)
  - For TSP: e.g. 2-interchange, 3-interchange,
- Approximation algorithms (Ch. 16.6):
  - Performance guarantee: $(\bar{z} - z^*)/z^* \leq \alpha$ for some $0 < \alpha < 1$
  - Specialized algorithms for structured problems

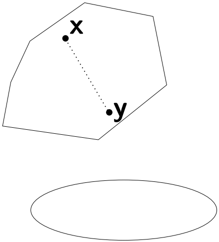
Recall: convex sets

- A set S is convex if, for any elements $\mathbf{x}, \mathbf{y} \in S$ it holds that

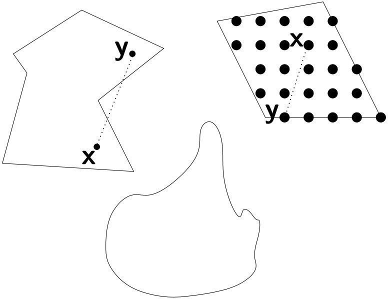
$$\alpha \mathbf{x} + (1 - \alpha)\mathbf{y} \in S \text{ for all } 0 \leq \alpha \leq 1$$

- Examples:

Convex sets



Non-convex sets



$\Rightarrow$ Integrality requirements $\Rightarrow$ nonconvex feasible set

Local vs. global optima

Consider a minimization problem:

$$\min_{\mathbf{x} \in X} \quad \mathbf{c}^{\mathrm{T}} \mathbf{x}$$

- Global optimum:
  A solution $\mathbf{x}^* \in X$ such that $\mathbf{c}^{\mathrm{T}} \mathbf{x}^* \leq \mathbf{c}^{\mathrm{T}} \mathbf{x}$ for all $\mathbf{x} \in X$
- ε-neighbourhood of $\bar{\mathbf{x}}$: $N_\varepsilon(\bar{\mathbf{x}}) = \left\{ \mathbf{x} \in X \,\middle|\, |\mathbf{x} - \bar{\mathbf{x}}| \leq \varepsilon \right\}$
- The distance measure $|\mathbf{x} - \bar{\mathbf{x}}|$ may be "freely" defined (e.g., number of arcs differing, euclidean, Manhattan, 2-interchange, ...)
- Local optimum:
  A solution $\bar{\mathbf{x}} \in X$ such that $\mathbf{c}^{\mathrm{T}} \bar{\mathbf{x}} \leq \mathbf{c}^{\mathrm{T}} \mathbf{x}$ for all $\mathbf{x} \in N_\varepsilon(\bar{\mathbf{x}})$

Local search heuristic algorithm (Ch. 16.4)

Consider a minimization problem:

$$\min_{\mathbf{x} \in X} \quad \mathbf{c}^{\mathrm{T}}\mathbf{x}$$

0. Initialization: Choose a feasible solution $\mathbf{x}^0 \in X$. Let $k = 0$.
1. Find all feasible points in an ε-neighbourhood $N_\varepsilon(\mathbf{x}^k)$ of $\mathbf{x}^k$
2. If $\mathbf{c}^{\mathrm{T}}\mathbf{x} \geq \mathbf{c}^{\mathrm{T}}\mathbf{x}^k$ for all $\mathbf{x} \in X \cap N_\varepsilon(\mathbf{x}^k) \Rightarrow$ Stop. $\mathbf{x}^k$ is a local optimum (w.r.t. ε)
3. Choose $\mathbf{x}^{k+1} \in X \cap N_\varepsilon(\mathbf{x}^k)$ such that $\mathbf{c}^{\mathrm{T}}\mathbf{x}^{k+1} < \mathbf{c}^{\mathrm{T}}\mathbf{x}^k$
4. Let $k := k + 1$ and go to step 1

More about heuristics

- Start using a constructive heuristic $\Rightarrow$ feasible solution
- The definition of a neighbourhood is model specific (e.g. geometrical distance, number of arcs differing,)
- Apply a local search algorithm
- Finds a *local* optimal solution
- *No guarantee* to find global optimal solutions
- Extensions (e.g. tabu search): Temporarily allow worse solutions to "move away" from a local optimum (Ch. 16.5)
- Larger neighbourhoods yield better local optima, but takes more computational time