Prof. Friedrich Eisenbrand — Question session: 8.12.10
Location: MA A3 31 — Discussion: 15.12.10

Exercises

Optimization Methods in Finance

Fall 2010

Sheet 6

Note: This is just one way, a solution could look like. We do not guarantee correctness. It is your task to find and report mistakes.

Exercise 6.1 (*)
We consider a combinatorial action in which we have $b_i \in \mathbb{N}$ times item $i \in \{1,\ldots,n\}$, $B := \max_{i=1,\ldots,n}\{b_i\}$ (in other words, there are just n many different undistinguishable items). Bids are pairs (S_j, p_j), i.e. bidder $j \in 1,\ldots,m$ is willing to pay p_j for items $S_j \subseteq \{1,\ldots,n\}$. Design a dynamic programming approach to select a feasible subset of bids that maximize the profit.
Remark: This can be done with $O((B+1)^n)$ table entries.

Solution:
Let

$$A(i,b'_1,\ldots,b'_n) = \max\left\{\sum_{j\in I} p_j \mid I \subseteq \{1,\ldots,i\} : |\{j \in I : i \in S_j\}| \le b'_i\right\}$$

be the profit that we can make from selling b'_j times item j to the first i bidders. Define

$$\gamma(S_j,i) = \begin{cases} 1 & i \in S_j \\ 0 & \text{otherwise} \end{cases}$$

We compute table entries using the recursion

$$A(i,b'_1,\ldots,b'_n) = \max\{A(i-1,b'_i,\ldots,b'_i),\, A(i-1,b'_1-\gamma(i,S_j),\ldots,b'_n-\gamma(i,S_j))\}$$

Then $A(n,b_1,\ldots,b_n)$ denotes the maximum profit.

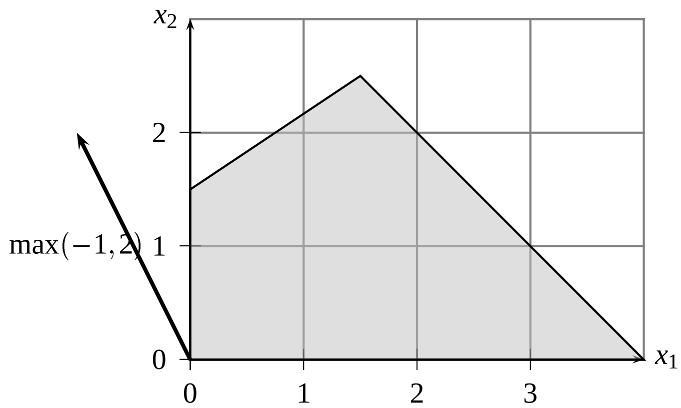
Exercise 6.2 (*)
Solve the following integer linear program with Branch & Bound.

$$\begin{aligned} &\min(1,-2)x \\ \begin{pmatrix}-4 & 6\\ 1 & 1\end{pmatrix} x &\le \begin{pmatrix}9\\4\end{pmatrix} \\ x &\ge \mathbf{0} \\ x &\in \mathbb{Z}^2 \end{aligned}$$

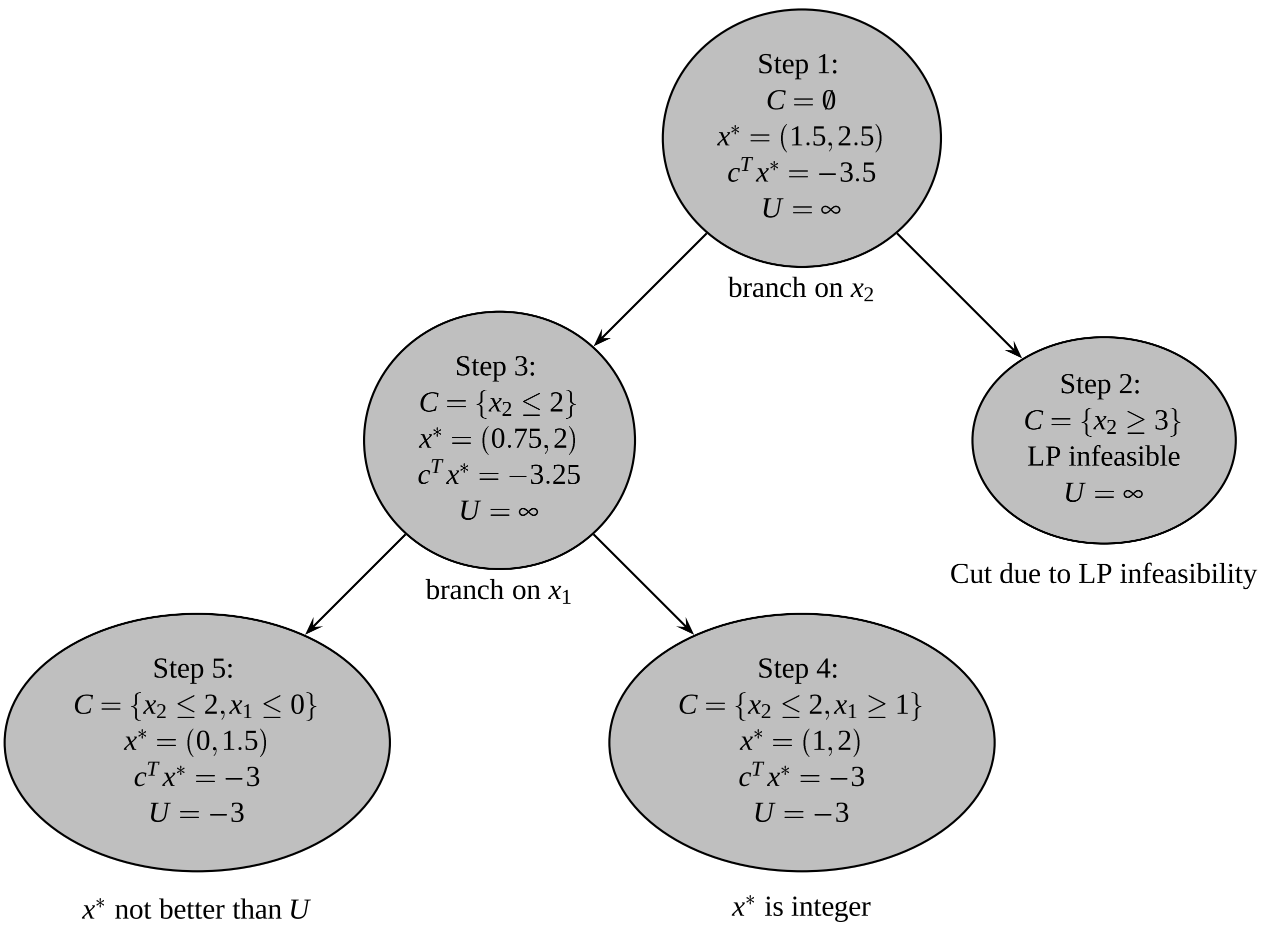
You can solve the LP subproblems with any computer algebra system or by inspecting a drawing.

Solution:

The set of feasible solutions looks as follows



We display the branch & bound process in a tree structure. Each node corresponds to one iteration. C gives the additional constraints, U gives the value of the best found integer solution (at the end of the iteration). x^* gives the optimum fractional solution.



The optimum solution is consequently $x = (1, 2)$.

Exercise 6.3 (*)
Recall that for a *combinatorial auction*, an actioneer is selling items $M = \{1,\dots,n\}$ and receives bids $B_j = (S_j, p_j)$ with $j = 1,\dots,n$, where $S_j \subseteq M$ is a subset of the items and $p_j \geq 0$ is the price. The goal is to select a feasible subset of the bids that maximize the cumulated price. This problem can be formulated as an integer linear program

$$\begin{array}{rcll} \max \sum_{i=1}^{n} x_j p_j & & & \\ \sum_{j:i\in S_j} x_j & \leq & 1 & \forall i \in M \\ 0 \leq x_j & \leq & 1 & \forall j = 1,\dots,n \\ x_j & \in & \mathbb{Z} & \forall j = 1,\dots,n \end{array}$$

Let IP be its optimum objective function value and let LP be the optimum value of the linear programming relaxation (i.e. the value of the above formulation without the integrality constraint). Of course $LP \geq IP$. Can you find a family of instances, where $\frac{LP}{IP}$ tends to ∞, when n grows?
Remark: The supremum of the ratio $\frac{LP}{IP}$ is usually called the *integrality gap*. A gap of roughly $\sqrt{n}$ is possible, even if all prices are 1.

Solution:
Consider $\sqrt{n}$ lines $L_1,\dots,L_{\sqrt{n}}$ in $\mathbb{R}^2$ that are in *general position*. That means no lines are parallel (hence they intersect in exaclty one point). Additionally no 3 lines intersect in a point. Such lines can be easily obtained either by a probabilitstic argument or by an explicit construction. We consider each of the $\binom{\sqrt{n}}{2} \leq (\sqrt{n})^2 = n$ intersection points $p_1, p_2, \dots, p_{\binom{\sqrt{n}}{2}} \subseteq \mathbb{R}^2$ as a single item. We consider every line L_i as a bid with price $p_i = 1$ and the set of corresponding items is the set $\{p_j \mid p_j \in L_i\}$ of points on the line. Then the best optimum solution picks just a single line (we cannot pick 2 lines, since they intersect in a point, i.e. they want the same item). On the other hand, if $x_i = \frac{1}{2}\ \forall i = 1,\dots,\sqrt{n}$ is a feasible fractional solution of value $\frac{\sqrt{n}}{2}$.

Exercise 6.4 - Practical exercise - 1 points
Implement the branch and bound algorithm.

1. You can implement the algorithm in one of the programming languages C/C++/Java/Pascal/Basic/Matlab (you can choose your favourite one).

2. Your submission should contain your (compilable) code and both, the optimum integral solution and its value for the 2 problems

$$\begin{array}{lrcll} \max & (5,\ 6,\ 10,\ 10,\ 8,\ 10,\ 10,\ 10,\ 9,\ 10)^T x & & (IP1) & \\ & (4788,\ 3703,\ 8104,\ 8357,\ 5089,\ 6832,\ 9723,\ 7054,\ 3680,\ 6088)x & \leq & 10000 & \\ & -x_i & \leq & 0 & \forall i = 1,\dots,10 \\ & x_i & \in & \mathbb{Z} & \forall i = 1,\dots,10 \end{array}$$

$$\max (1.1, 1.2, 1.3, 1.4, 1.5, 1.6)^T x \qquad (IP2)$$

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix} x \leq \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$x_1, \dots, x_6 \in \mathbb{Z}$$

3. Send the files till **22.12.10 (23:59h)** to thomas.rothvoss@epfl.ch.
4. You can work in groups up to 3 people (you need only one submission per group).

Hints:

1. You can use any LP library to solve the relaxations (in Matlab, there is a build-in command to solve linear programs, for C/C++/Java there are plenty of libraries available).
2. We recommend to test your instance with a small example, say from exercise 6.2.
3. If for a fractional solution x^* one has several variables $x_i^* \notin \mathbb{Z}$ you can branch on an arbitrary such i.