Exercise 1

Given n "items" and a "container", a "weight" p_j and a "cost" c_j (with p_j and c_j positive integers) are associated with each item j ($j = 1, \ldots, n$).

Determine a subset M of the n items so that:

a) the sum of the weights of the items in M is not smaller than a given value a;

b) the cardinality of M is not smaller than a given value b;

c) the sum of the costs of the items in M is minimum.

1) Prove that the problem is NP-hard.
2) Define a Linear Integer Programming model for the considered problem.
3) Define the complexity of the problem for determining a feasible solution for the following problems:

3.1) original problem;

3.2) problem with constraint a) imposed, and constraint b) replaced by the constraint imposing that the cardinality of M must be equal to b;

3.3) problem with constraint a) imposed, and constraint b) replaced by the constraint imposing that the cardinality of M is not greater than b;

3.4) problem with constraint b) imposed, and constraint a) replaced by the constraint imposing that the sum of the weights of the items of M must be equal to a.

EXERCIXE 1

1.2) Possible mathematical model (BLP)

$$x_j = \begin{cases} 1 & \text{if item } j \text{ is selected} \\ 0 & \text{otherwise} \end{cases} \qquad j = 1, \dots, n$$

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

s.t.

$$\sum_{j=1}^{n} p_j x_j \geq a \qquad (a)$$

$$\sum_{j=1}^{n} x_j \geq b \qquad (b)$$

$$x_j \in \{0, 1\} \qquad j = 1, \dots, n$$

1.1) Size of P: $n, a, b, (c_j), (p_j) \Rightarrow O(n) : n$

- $P \in NP$ (decision tree with n levels, 2 descendent nodes)
- KP01-min $\propto P$ (KP01-min: $\bar{n}, \bar{b}, (P_j), (W_j)$ size $\bar{n}$)

$n := \bar{n}$; $a := \bar{b}$; $c_j := P_j$, $p_j := W_j$ $j = 1, \dots, \bar{n}$ $\quad O(\bar{n})$

$b := 0$

$\Rightarrow P \in NP\text{-Hard}$

1.3.1) $F\text{-}P \in \mathcal{P}$ (set $x_j := 1$ for $j = 1, \dots, n$; check if (a) and (b) are satisfied) $\quad O(n)$

1.3.2) $F\text{-}P \in \mathcal{P}$ (1. sort the n items according to non-increasing values of p_j;
2. set $x_j := 1$ for $j = 1, \dots, b$; $x_j := 0$ for $j = b+1, \dots, n$
3. check if (a) is satisfied)
$O(n \log n)$

1.3.3) $F\text{-}P \in \mathcal{P}$: as done for 1.3.2).

1.3.4) $F\text{-}P \in NP$ (...)

$PP \propto F\text{-}P$ (...; $b := 0$) $\Rightarrow F\text{-}P \in NP\text{-Hard}$

Exercise 2

Given n "operations" and m "machines", an "initial time" a_j and a "final time" b_j are associated with each operation j (j = 1, …, n). Each machine can process at any time at most one operation, and can globally work for a time period not greater than a given value C.

1) Prove that the problem for determining a feasible solution of the considered problem is NP-hard.

2) Define a Linear Integer Programming model for the considered problem in the case in which the number of used machines must be minimized.

EXERCISE 2

2.1) Size of F-P: n, m, (a_j), (b_j), $C \Rightarrow O(n): n \; (m < n)$

* $F\text{-}P \in NP$: (decision tree with n levels (one for each operation; m descendent nodes (one for each machine).

* $PP2 \propto F\text{-}P$ (PP2: Partion Problem with $\bar{C} = \sum_{j=1}^{\bar{n}} \bar{p}_j / 2$)

Size of PP2: $\bar{n}$, $(\bar{p}_j)$, $\bar{C} \Rightarrow O(\bar{n}): \bar{n}$

$$O(\bar{n}) \left\{ \begin{array}{l} n := \bar{n};\ C := \bar{C};\ m := 2; \\ a_1 := 0,\ b_1 := \bar{p}_1;\ a_j := b_{j-1} + 1,\ b_j := a_j + \bar{p}_j \ (j = 2, \dots, n). \end{array} \right.$$

PP2 is feasible if and only if F-P has a solution

$\Rightarrow F\text{-}P \in NP\text{-Hard}$

2.2)

$$x_{ij} = \begin{cases} 1 & \text{if operation } j \text{ is processed on machine } i \\ 0 & \text{otherwise} \end{cases} \quad i = 1, \dots, m;\ j = 1, \dots, n$$

$$y_i = \begin{cases} 1 & \text{if machine } i \text{ is used} \\ 0 & \text{otherwise} \end{cases} \quad i = 1, \dots, m$$

$$\min z = \sum_{i=1}^{m} y_i$$

s.t.

$$\sum_{i=1}^{m} x_{ij} = 1 \qquad j = 1, \dots, n$$

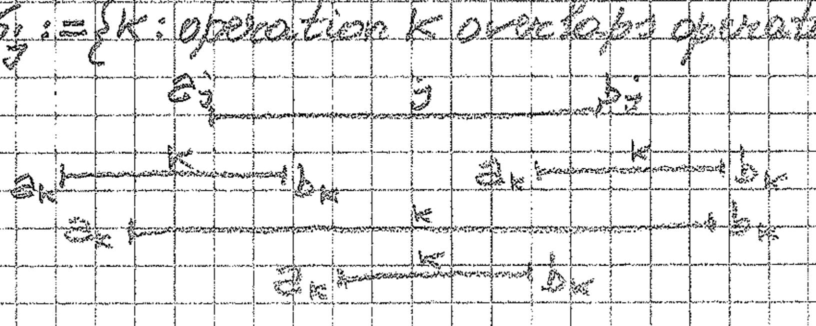
$$\sum_{j=1}^{n} (b_j - a_j) x_{ij} \le C y_i \qquad i = 1, \dots, m$$

$$x_{ij} + x_{ik} \le 1 \qquad i = 1, \dots, m;\ j = 1, \dots, n;\ k \in S_j$$

$$x_{ij} \in \{0, 1\} \ i = 1, \dots, m;\ j = 1, \dots, n;\ y_i \in \{0, 1\} \ i = 1, \dots, m$$

with:

$S_j := \{k : \text{operation } k \text{ overlaps operation } j\} \quad j = 1, \dots, n$



Exercise 3

Given a "depot" which must serve m "customers". The customers can be served by using n different "routes". In particular, each customer i ($i = 1, \ldots, m$) can be served by a subset V_i of routes (with V_i contained in the set $\{1, 2, \ldots, n\}$). Each route j ($j = 1, \ldots, n$) has a "cost" c_j and a "traveling time" t_j (with c_j e t_j non-negative). Determine a subset S of the n routes such that:

a) each customer is served by at least one route of S;

b) the sum of the traveling times of the routes of S is not smaller than a given value d;

c) the sum of the costs of the routes of S is minimum.

1) Define a Linear Integer Programming model for the considered problem.
2) Prove that the problem is NP-hard.
3) Define the complexity of the problem for determining a feasible solution for the considered problem.
4) As at point 3) in the case in which in constraint b) it is imposed that the sum of the traveling times of the routes of S is equal to a given value d

EXERCISE 3

3.1)

$$x_j = \begin{cases} 1 & \text{route } j \text{ is selected (i.e. if } j \in S) \\ 0 & \text{otherwise} \end{cases} \qquad j = 1, \dots, n$$

$$\min z = \sum_{j=1}^{n} c_j x_j \qquad (c)$$

s.t.

$$\sum_{j \in V_i} x_j \geq 1 \qquad i = 1, \dots, m \quad (a)$$

$$\sum_{j=1}^{n} t_j x_j \geq d \qquad (b)$$

$$x_j \in \{0, 1\} \qquad j = 1, \dots, n \quad (d)$$

Constraints (a) can also be written as:

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1 \qquad i = 1, \dots, m \quad (a')$$

where

$$a_{ij} = \begin{cases} 1 & \text{if route } j \in V_i \\ 0 & \text{otherwise} \end{cases} \qquad j = 1, \dots, n \quad i = 1, \dots, m$$

3.2) Size of P: $m, n, (c_j), d, (t_j), (V_i) \Rightarrow O(mn)$: mn

- $P \in NP$: binary decision tree with n levels
- KP01-Min $\propto P$: size of KP01-Min: $\bar{n}, \bar{b}, (\bar{c}_j), (\bar{w}_j) \Rightarrow \bar{n}$

$O(\bar{n})\{$ $n := \bar{n}$; $d := \bar{b}$; $(c_j) := (\bar{c}_j)$; $(t_j) := (\bar{w}_j)$; $m := 1$; $V_1 := \{1, \dots, n\}$.
The optimal solution of P is also optimal for KP01-Min.

3.3) F-$P \in P$: set $x_j = 1$ for $j = 1, \dots, n$ and check (a) and (b).

3.4) • F-$P2 \in NP$: binary decision tree with n levels

$O(\bar{n})\{$ • $PP \propto F$-$P2$: ... ; $m := 1$, $V_1 := \{1, \dots, \bar{n}\}$;
PP is feasible if and only if F-P has a solution
$\Rightarrow F$-$P2$ is NP-Hard

Exercise 4

Given m "items" and n "vehicles": a positive "weight" p_j is associated with each item j (j= 1, … , m); a positive "capacity" a_i is associated with each vehicle i (i = 1, … , n). Also assume: m > n > 0.

Determine the items to be loaded into the vehicles so that:

a) the sum of the weights of the items loaded into each vehicle i is not greater than the capacity a_i ;

b) each item j is loaded into no more than one vehicle;

c) the global number of items loaded into the vehicles is smaller than a given value k;

d) the sum of the weights of the items loaded into the vehicles is maximum.

1) Define a Linear Integer Programming model for the considered problem.
2) Prove that the problem is NP-hard.
3) Define the complexity of the problem for determining a feasible solution for the following problems:
   - 3.1) original problem;
   - 3.2) problem with "not smaller" instead of "smaller" in constraint c);
   - 3.3) problem with "equal" instead of "smaller" in constraint c).

Exercise 4

4.1)

$$x_{i,j} = \begin{cases} 1 & \text{if item } j \text{ loaded into vehicle } i \\ 0 & \text{otherwise} \end{cases} \qquad i=1,\dots,n;\ j=1,\dots,m$$

$$\max\ Z = \sum_{i=1}^{n} \sum_{j=1}^{m} p_j x_{i,j} \qquad (d)$$

s.t.

$$\sum_{j=1}^{m} p_j x_{i,j} \le a_i \qquad i=1,\dots,n \qquad (a)$$

$$\sum_{i=1}^{n} x_{i,j} \le 1 \qquad j=1,\dots,m \qquad (b)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{m} x_{i,j} < k \qquad (c)$$

$$x_{i,j} \in \{0,1\} \qquad i=1,\dots,n;\ j=1,\dots,m$$

4.2) Size: n, m, (p_j), (a_i), $k \Rightarrow O(m+n)$: m

* $P \in NP$: decision tree with m levels (one for each item) and $(n+1)$ descendent nodes

* $SSP \propto P$: size of SSP: $\bar{n}$, $(\bar{w}_j)$, $\bar{c} \Rightarrow \bar{n}$

$O(\bar{n})$ $\left\{ \begin{array}{l} n := 1;\ m := \bar{n};\ p_j := \bar{w}_j\ (j=1,\dots,m);\ a_1 := \bar{c}; \\ k := m+1 \end{array} \right.$

The optimal solution of P is also optimal for SSP.

4.3.1) $F\text{-}P \in \mathcal{P}$: no items are loaded (always feasible) : $S=\emptyset$.

4.3.2) * $F\text{-}P2 \in NP$: decision tree with m levels and $(n+1)$ descendent nodes.

* $PP2 \propto F\text{-}P2$ (PP2: Partition Problem with $\bar{c} = \sum_{j=1}^{\bar{n}} \bar{p}_j / 2$)

Size of PP2: $\bar{n}$, $(\bar{p}_j)$, $(\bar{c}) \Rightarrow O(\bar{n})$: $\bar{n}$

$O(\bar{n})$ $\{$ $m := \bar{n}$, $k := m$, $n = 2$, $a_1 = a_2 = \bar{c}$, $p_j := \bar{p}_j$ $(j=1,\dots,m)$

PP2 is feasible if and only if F-P2 has a solution.

$\Rightarrow$ F-P2 $\in$ NP-Hard

4.3.3) As for 4.3.2

Exercise 5

Given a "directed graph" $G = (V, A)$, with $|V| = n$ and $|A| = m$. A positive "cost" $c_{i,j}$ is associated with each arc (i, j) in A. Assume also that the vertex set V is partitioned into K subsets ("regions") $R_1, R_2, \ldots, R_K$, with $R_1 = \{1\}$.

Determine an "elementary circuit" of G (i.e., a circuit passing at most once through each vertex of G) visiting at least one vertex of each of the K regions, and such that the sum of the costs of the arcs of the circuit is minimum.

1)- Prove that the problem is NP-hard.

2)- Define a Linear Integer Programming model for the considered problem.

3)- Assuming that the graph G is complete and that the cost matrix $(c_{i,j})$ satisfies the "triangularity condition" (i.e.: $c_{i,k} + c_{k,j} <= c_{i,j}$ for each triple (i, j, k) of vertices of V), define the new Linear Integer Programming model so as to "strengthen" the constraints of the model.

EXERCISE 5

5.1) Size of P: $n, m, (c_{ij}), K, (R_h) \Rightarrow O(m)$: m $(\leq n^2)$

* $P \in NP$: decision tree with $(n-1)$ level (one for each successor vertex in the circuit) with at most $(n-1)$ descendent nodes (at the first level).

* ATSP $\propto$ P

- Size of ATSP: $\bar{n}, (\bar{c}_{ij}) \Rightarrow O(\bar{n}^2)$: $\bar{n}^2$

$$O(\bar{n}^2) \left\{ \begin{array}{l} n := \bar{n};\ c_{ij} := \bar{c}_{ij} \text{ for } i=1,\dots,\bar{n} \text{ and } j=1,\dots,\bar{n};\ m = \bar{n}^2; \\ K := n;\ R_h := \{h\} \text{ for } h=1,\dots,n. \end{array} \right.$$

- The optimal solution of P is also optimal for ATSP.

5.2) Transform graph G into a complete graph:

for $i=1,\dots,n$ and $j=1,\dots,n$: if $(i,j) \notin A$ then $c_{ij} := \infty$

$$x_{ij} = \begin{cases} 1 & \text{if arc } (i,j) \text{ in the optimal circuit} \\ 0 & \text{otherwise} \qquad i=1,\dots,n;\ j=1,\dots,n \end{cases}$$

$$y_i = \begin{cases} 1 & \text{if vertex } i \text{ in the optimal circuit} \\ 0 & \text{otherwise} \qquad i=1,\dots,n \end{cases}$$

$$\min\ z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \qquad (a)$$

s.t.

$$\sum_{j=1}^{n} x_{ij} = y_i \qquad i=1,\dots,n \qquad (b)$$

$$\sum_{j=1}^{n} x_{ij} = \sum_{j=1}^{n} x_{ji} \qquad i=1,\dots,n \qquad (c)$$

$$\sum_{i \in R_h} y_i \geq 1 \qquad h=1,\dots,K \qquad (d)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \leq |S|-1 \qquad \forall S \subseteq V-\{1\},\ S \neq \emptyset$$

$$x_{ij} \in \{0,1\}\ i=1,\dots,n;\ j=1,\dots,n;\ y_i \in \{0,1\}\ i=1,\dots,n$$

5.3) Replace (d) with $\sum_{i \in R_h} y_i = 1 \quad h=1,\dots,K \qquad (d'')$