Exercise 6

Given a "complete directed graph" $G = (V, A)$, with $|V| = n$. A positive "cost" $c_{i,j}$ (with $c_{i,i}$ = infinity for each vertex i of V) is associated with each arc (i, j) of A. A positive "prize" p_i is associated with each vertex i of V

Given a vertex r of V, determine an "elementary circuit" of G visiting vertex r and such that:

a) the sum of the costs of the arcs of the circuit is minimum;

b) the sum of the prizes associated with the vertices of the circuit is not smaller than a given value a;

c) the number of vertices of the circuit having a prize smaller than a given value b is not smaller than a given percentage d of the number of vertices of the circuit (with d between 0 and 1).

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

2)- Prove that the problem is NP-hard. (determine two NP-hard problems reducible to the considered problem).

3) Define the complexity of the problem for determining a feasible solution for the considered problem.

4)- Define the complexity of the problem in the case in which: $c_{i,j} = K$ for each arc (i, j) of G (with K given value).

EXERCISE 6

6.1)
$$x_{i,j} = \begin{cases} 1 & \text{if arc } (i,j) \text{ in the optimal circuit} \\ 0 & \text{otherwise} \quad i = 1,\dots,n;\ j = 1,\dots,n \end{cases}$$

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

$$\min\ z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{i,j}\, x_{i,j}$$

s.t.

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

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

$$y_r = 1$$

$$\sum_{i \in S} \sum_{j \in S} x_{i,j} \le |S| - 1 \qquad \forall S \subseteq V \setminus \{r\},\ S \ne \emptyset$$

$$\sum_{i=1}^{n} p_i\, y_i \ge a$$

$$\sum_{i \in R} y_i \ge d \sum_{i=1}^{n} y_i$$

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

$$y_i \in \{0,1\} \quad i = 1,\dots,n$$

with $R = \{ i \in V : p_i < b \}$

6.2) Size: $n, r, a, d, b, (p_i), (c_{ij})$: $5+n+n^2 \Rightarrow n^2$

- $P \in NP$: decision tree with $(n-1)$ levels and at most $(n-1)$ descendent nodes.

- ATSP α P : size of ATSP: $\bar{n}, (\bar{c}_{ij})$
  $n := \bar{n}$; $c_{ij} := \bar{c}_{ij}$ $i=1,\dots,n$; $j=1,\dots,n$
  $r := 1$; $p_i := 1$ for $i=1,\dots,n$; $a := n$; $b := 2$; $d := 1$.
  The optimal solution of P is also optimal for ATSP

- KP01-Min α P: size of KP01-Min: $\bar{n}, (\bar{c}_j), (\bar{w}_j), \bar{f}$
  $n := \bar{n}+1$; $p_i := \bar{w}_i$ for $i=1,\dots,\bar{n}$; $p_n := 1$;
  $r := n$; $a := \bar{f}+1$; $b := \bar{f}+1$; $d := 0$
  $\bar{c}_n := 1$; $c_{ij} := \bar{c}_j$ for $i=1,\dots,n$ and $j=1,\dots,n$
  The optimal solution of P is also optimal for KP01-Min.

6.3) F-P $\in$ P

1) - $y_i := 1$ for $i \in R$.
2) - $K = \lfloor |R| / d \rfloor$ (K = max number of vertices in the circuit);
   - if $K \geq n$ then $y_i := 1$ for $i \in V \setminus R$ and go to Step 4.
3) - sort the vertices in $V \setminus R$ according to non increasing values of p_i; $O(n \log n)$
   - set $y_i := 1$ for the first $K - |R|$ vertices in $V \setminus R$;
   - if $y_r = 0$ then set $y_r := 1$ and $y_h := 0$, with $h = (K-|R|)$-th vertex in $V \setminus R$.
4) If $\sum_{i=1}^{n} p_i y_i \geq a$ then a feasible solution (y_i) for F-P exists
   else F-P has no solution.

Exercise 7

Given n "jobs" and m "machines". The "cost" per processing job j ($j = 1, \dots, n$) on machine i ($i = 1, \dots, m$) is given by $c_{i,j}$, while the corresponding "processing time" is given by $t_{i,j}$. If machine i ($i = 1, \dots, m$) is used, there is an additional cost equal to b_i (this cost is equal to zero if machine i is not used). The values $c_{i,j}$, $t_{i,j}$ and b_i are positive integers.

Assign each job to one and only one machine so that:

a) the global processing time for each machine i ($i = 1, \dots, m$) is not greater than a given value a_i (with a_i positive and integer);

b) the "global" cost is minimum.

1)- Define the complexity of the problem for determining a feasible solution for the considered problem.

2)- Prove that the considered problem is NP-hard.

3)- Define a Linear Integer Programming model for the considered problem, so as to minimize the number of constraints.

4)- Define additional Linear Integer Programming models which, by using a larger number of constraints, can produce "lower bounds" (obtained with the continuous relaxation of the model) better than those which can be obtained with the continuous relaxation of the model defined at point 3).

EXERCISE 7

7.1) • Size of F-P: n, m, $(t_{i,j})$, (a_i): $2 + m*n + m \Rightarrow m*n$

• F-P $\in$ NP: decision tree with n levels (one level for each job) and m descendent nodes (one node for each machine).

• F-GAP $\propto$ F-P: size of F-GAP: $\bar{n}$, $\bar{m}$, $(\bar{t}_{i,j})$, $(\bar{b}_i)$: $\bar{m}*\bar{n}$

$$O(\bar{m}*\bar{n}) \left\{ \begin{array}{l} n := \bar{n};\ m := \bar{m};\ t_{i,j} := \bar{t}_{i,j} \quad i = 1,\dots,m;\ j = 1,\dots,n \\ a_i := \bar{b}_i \quad i = 1,\dots,m \end{array} \right.$$

• F-GAP is feasible if and only if F-P is feasible.

7.2) • Size of P: n, m, $(c_{i,j})$, $(t_{i,j})$, (a_i), (b_i): $n*m$

• P $\in$ NP : same decision tree of F-P

• P is NP-Hard since F-P is NP-Hard.

7.3)
$$x_{i,j} = \begin{cases} 1 & \text{is job } j \text{ is assigned to machine } i \\ 0 & \text{other wise} \quad i = 1,\dots,m;\ j = 1,\dots,n \end{cases}$$

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

$$\min\ z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{i,j}\, x_{i,j} + \sum_{i=1}^{m} b_i\, y_i \qquad (1)$$

s.t.
$$\sum_{i=1}^{m} x_{i,j} = 1 \qquad j = 1,\dots,n \qquad (2)$$

$$\sum_{i=1}^{m} \sum_{j=1}^{n} t_{i,j}\, x_{i,j} \le a_i\, y_i \qquad i = 1,\dots,n \qquad (3)$$

$$x_{i,j} \in \{0,1\} \quad i = 1,\dots,m;\ j = 1,\dots,n;\ y_i \in \{0,1\} \quad i = 1,\dots,m \quad (4)$$

7.4) a) Replace (3) with:
$$\sum_{j=1}^{n} t_{i,j}\, x_{i,j} \le a_i \qquad i = 1,\dots,m \qquad (5)$$

$$x_{i,j} \le y_i \qquad i = 1,\dots,m;\ j = 1,\dots,n \qquad (6)$$

b) Add (6) to (1)–(4).

Exercise 8

Given a "complete directed graph" $G = (V, A)$, with $| V | = n$: a "weight" $p_{i,j}$ and a non-negative "time" $t_{i,j}$ are associated with each arc (i, j) of A. Two disjoint subsets S and T are also given (with S and T contained in A).

Determine a "Hamiltonian circuit" H of G so that:

a) the sum of the weights of the arcs of H is maximum;

b) the sum of the times of the arcs of H is not greater than a given value d;

c) the number of arcs of H belonging to subset S is not smaller than the number of arcs of H belonging to subset T.

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

2)- Prove that the considered problem is NP-hard.

3) Define the complexity of the problem for determining a feasible solution for the considered problem.

4)- Define the complexity of the problem for determining a feasible solution for the problem in the case in which constraint b) is not imposed.

EXERCISE 8

8.1)
$$x_{ij} = \begin{cases} 1 & \text{if arc } (i,j) \text{ belongs to } H \\ 0 & \text{otherwise} \quad i=1,\dots,n;\ j=1,\dots,n \end{cases}$$

$$Z = \max \sum_{i=1}^{n} \sum_{j=1}^{n} p_{ij} x_{ij} \qquad (1)$$

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad i=1,\dots,n \qquad (2)$$

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad j=1,\dots,n \qquad (3)$$

$$\sum_{i\in R} \sum_{j\in R} x_{ij} \le |R|-1 \quad \forall R \subset V: |R| \ge 2 \qquad (4)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} t_{ij} x_{ij} \le d \qquad (5)$$

$$\sum_{(i,j)\in S} x_{ij} \ge \sum_{(i,j)\in T} x_{ij} \qquad (6)$$

$$x_{ij} \in \{0,1\} \qquad i=1,\dots,n;\ j=1,\dots,n \qquad (7)$$

8.2)
- Size of P: $n, (p_{ij}), (t_{ij}), d, S, T$: $2+4n^2 \Rightarrow n^2$
- $P \in NP$: decision tree with $(n-1)$ levels (one for each arc of the circuit) and at most $(n-1)$ descendent nodes.
- ATSP-Max $\propto P$: size of ATSP-Max: $\bar{n}, (\bar{p}_{ij})$: $\bar{n}^2$

$$O(\bar{n}^2) \begin{cases} n := \bar{n};\ p_{ij} := \bar{p}_{ij} \quad i=1,\dots,n;\ j=1,\dots,n \\ t_{ij} := 0 \quad i=1,\dots,n;\ j=1,\dots,n;\ d := 0; \\ S := \emptyset,\ T := \emptyset \end{cases}$$

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

8.4) • Size of F-P1: n, S, T.

• $F\text{-}P1 \in NP$: as done in 8.2.

• $HC \propto F\text{-}P1$:

size of HC: $\bar{n}, \bar{A} \Rightarrow \bar{n}^2$

$$O(\bar{n}^2) \left\{ \begin{array}{l} n := \bar{n} \\ S := \emptyset \\ T := \{(i,j) : (i,j) \notin \bar{A}\} \end{array} \right.$$

(check if a Hamiltonian circuit using only the arcs in set $\bar{A}$ exists).

• HC is feasible if and only if F-P1 is feasible.

8.3) • Size of F-P: n, (t_{ij}), d, (S), $(T) \Rightarrow n^2$

• $F\text{-}P \in NP$: as done in 8.2.

• F-P is NP-Hard since it is a generalization of F-P2 which is NP-Hard.

8.5) ADDITIONAL QUESTION

"Define the complexity of the problem for determining a feasible solution for the problem in the case in which constraint c) is not imposed".

• Size of F-P2: n, (t_{ij}), $d \Rightarrow n^2$

• $F\text{-}P2 \in NP$: as done in 8.2.

• $HC \propto F\text{-}P2$: size of HC: $\bar{n}, \bar{A} \Rightarrow \bar{n}^2$

$n := \bar{n}$;

<u>for</u> $i = 1, \dots, n$ <u>and</u> $j = 1, \dots, n$ <u>do</u> $t_{ij} := 0$ <u>if</u> $(i,j) \in \bar{A}$

$t_{ij} := 1$ <u>if</u> $(i,j) \notin \bar{A}$

$d := 0$.

• HC is feasible if and only if F-P2 is feasible.

Exercise 9

Given n "depots" and m "customers": each customer i ($i = 1, \ldots, m$) has a non-negative "potential profit" p_i. Each depot j ($j = 1, \ldots, n$) has a non-negative "cost" c_j and is able to "serve" a subset of the m customers. In particular, a binary matrix $(a_{i,j})$ is given, such that for each pair [depot j, customer i] (with $j = 1, \ldots, n$ and $i = 1, \ldots, m$) $a_{i,j} = 1$ if depot j is able to serve customer i, and $a_{i,j} = 0$ otherwise.

For each subset S of the n depots, the corresponding "global profit" is given by the difference: (sum of the profits of the customers which can be served by the depots of S) - (sum of the costs of the depots of S).

Determine a subset S^* of the n depots so that:

a) S^* contains at most d depots (with d given value greater than 0 and smaller or equal to n);

b) the global profit of S^* is maximum;

c) the total cost of the depots of S^* is not smaller than a given non-negative value b.

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

2)- Prove that the considered problem is NP-hard.

3) Define the complexity of the problem for determining a feasible solution for the considered problem.

4)- Define the complexity of the problem for determining a feasible solution for the problem in the following cases:

4.1) subset S^* must contain exactly d depots;

4.2) the global cost of the depots of subset S^* is equal to b.

EXERCISE 9

9.1)

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

$$y_i = \begin{cases} 1 & \text{if customer } i \text{ is served by at least one depot in } S^* \\ 0 & \text{otherwise} \qquad i = 1, \dots, m \end{cases}$$

$$z = \max \sum_{i=1}^{m} p_i y_i - \sum_{j=1}^{n} c_j x_j \qquad (1)$$

s.t.

$$\sum_{j=1}^{n} a_{ij} x_j \geq y_i \qquad i = 1, \dots, m \qquad (2)$$

$$\sum_{j=1}^{n} x_j \leq d \qquad (3)$$

$$\sum_{j=1}^{n} c_j x_j \geq b \qquad (4)$$

$$x_j \in \{0, 1\} \qquad j = 1, \dots, n \qquad (5)$$

$$y_i \in \{0, 1\} \qquad i = 1, \dots, m \qquad (6)$$

9.3) $F\text{-}P \in \mathcal{P}$

$O(n)$ • in constraint (2) set $y_i := 0$: (2) always satisfied;

• sort the n depots according to non increasing values

$O(n \log n)$ of the costs (c_j);

$O(n)$ • $x_j := 1$ for $j = 1, \dots, d$; $x_j := 0$ for $j = d+1, \dots, n$

$O(n)$ • If $\sum_{j=1}^{n} c_j x_j \geq b$ then a feasible solution (x_j) exists

else no feasible solution exists for F-P.

9.4.1) As for question 9.3).

9.2) Size of P: $n, m, b, d, (p_i), (c_j), (a_{ij})$:

$4+m+n+m*n \Rightarrow m*n$

- $P \in NP$: binary decision tree with n levels (one for each depot).

* SSP-Min:

$$z' = \min \sum_{j=1}^{\bar{n}} \bar{w}_j x_j \quad s.t. \sum_{j=1}^{\bar{n}} \bar{w}_j x_j \geq \bar{c}$$

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

- SSP-Min $\propto$ P:

size of SSP-Min: $\bar{n}, \bar{c}, (\bar{w}_j)$: $2+\bar{n} \Rightarrow \bar{n}$

$$O(\bar{n}) \left\{ \begin{array}{l} n := \bar{n};\ m := 1;\ b := \bar{c};\ d := n; \\ p_1 := 0;\ c_j := \bar{w}_j \quad j=1,\dots,n; \\ a_{1j} := 1 \quad j=1,\dots,n \end{array} \right.$$

* constraint (2) always satisfied ($y_1 = 0$ or $y_1 = 1$);

* constraint (3) always satisfied;

* the Objective Function of P is $\min z = \sum_{j=1}^{n} c_j x_j$ with constraints (4) and (5)

* The optimal solution of P is also optimal for SSP-Min.

9.4.2) F-P1 is NP-Hard

- F-P1 $\in$ NP (as for 9.2)
- F-EKP01 $\propto$ F-P1 (as for 9.2)

Exercise 10

Given a "complete directed graph" $G = (V, A)$, with $|V| = n$. A non-negative "time" $t_{i,j}$ (with $t_{i,i}$ = infinity for each vertex i of V) is associated with each arc (i, j) in A (with the times satisfying the triangularity condition, i.e.: $t_{i,k} + t_{k,j} <= t_{i,j}$ for each triple (i, j, k) of vertices of V). A positive profit a_i is associated with each vertex i of V.

Given a vertex h of V, determine an "elementary circuit" of G visiting h and such that:

a) the number of arcs of the circuit is not greater than a given value r (with r between 2 and n);

b) the sum of the times of the arcs of the circuit is not greater than a given value d;

c) the sum of the profits associated with the vertices of the circuit is maximum.

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

2)- Prove that the considered problem is NP-hard.

3) Define the complexity of the problem for determining a feasible solution for the considered problem.

4) Define a Linear Integer Programming model for the variant of the considered problem in which the objective function (to be maximized) is given by:

alpha * (sum of the profits associated with the vertices of the circuit) –

beta * (maximum time of the arcs of the circuit)

with alpha and beta non-negative given values.

(10.1)

10.1)

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

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

$$z = \max \sum_{j=1}^{n} a_i y_i \qquad (1)$$

$$y_i = \sum_{j=1}^{n} x_{ij} \qquad i=1,\dots,n \qquad (2)$$

$$\sum_{j=1}^{n} x_{ij} = \sum_{j=1}^{n} x_{ji} \qquad (3)$$

$$y_h = 1 \qquad (4)$$

$$\sum_{i\in S} \sum_{j\in S} x_{ij} \leq |S| - 1 \qquad \forall S \subseteq V \setminus \{h\},\ S \neq \emptyset \qquad (5)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} x_{ij} \leq k \qquad (6)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} t_{ij} x_{ij} \leq d \qquad (7)$$

$$x_{ij} \in \{0,1\} \quad i=1,\dots,n;\ j=1,\dots,n \qquad (8)$$

$$y_i \in \{0,1\} \quad i=1,\dots,n \qquad (9)$$

10.4)

$$z = \max \left\{ \alpha \sum_{i=1}^{n} a_i y_i - \beta V \right\}$$

s.t. $$t_{ij} x_{ij} \leq V \qquad i=1,\dots,n;\ j=1,\dots,n \qquad (10)$$

(2) - (9)

10.2

10.2) Size of P: $n, h, r, d, (t_{ij}), (a_i)$: $4 + n^2 + n \Rightarrow n^2$

- $P \in NP$: decision tree with $(r-1)$ levels (one for each arc of the circuit and at most $(n-1)$ descendent nodes.

* KP01 $\propto P$

- size of KP01: $\bar{n}, \bar{c}, (\bar{w}_i), (\bar{p}_i) \Rightarrow \bar{n}$

- $n := \bar{n} + 1$;
  $\bar{w}_n := 1$; $\bar{p}_n := 1$;
  $t_{ij} := \bar{w}_i$ $\quad i = 1, \dots, n$; $j = 1, \dots, n$;
  $a_i := \bar{p}_i$ $\quad i = 1, \dots, n$;
  $d := \bar{c} + 1$;
  $r := n$;
  $h := n$.

the optimal solution of P is also optimal for KP01.

10.3) $F\text{-}P \in \mathcal{P}$

* circuit starting from h and returning to h within the minimum time:

- determine vertex k such that:

$O(n)$ $\{$ $\quad t_{hk} + t_{kh} = \min\{t_{hi} + t_{ih} : i \in V \setminus \{h\}\}$

- If $t_{hk} + t_{kh} \le d$ then a feasible solution exists: circuit $h \to k \to h$
  else no feasible solution exists