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) Determine "good" Lagrangian Lower Bounds which can be computed through procedures having time complexity $O(n \log(n))$, and describe the corresponding subgradient optimization procedures.

2) Determine a "good" Surrogate Lower Bound which can be computed through a procedure having time complexity $O(n)$, and describe the corresponding subgradient optimization procedure.

EXERCISE 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: $\bar{n}, \bar{b}, (\bar{P}_j), (\bar{W}_j)$ size $\bar{n}$) $O(\bar{n})$

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

$\underline{b := 0}$

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

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) $\underline{O(n)}$

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)

$\underline{O(n \log n)}$

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

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

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

1.1.a) Lagrangian Relaxation of constraint (a) : ($\lambda \geq 0$)

$$L(\lambda) = \min \sum_{j=1}^{n} c_j x_j + \lambda \left(a - \sum_{j=1}^{n} p_j x_j\right)$$

$$L(\lambda) = \lambda a + \min \sum_{j=1}^{n} \hat{c}_j x_j$$

(with $\hat{c}_j = c_j - \lambda p_j$; $j = 1, \dots, n$) $\quad O(n)$

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

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

* Solution:

1) Sort the n items according to non decreasing values of $\hat{c}_j$ $\quad O(n \log n)$

2) $L(\lambda) := \lambda a$, $\bar{b} := 0$

3) **for** $j = 1, \dots, n$ **do** $\quad O(n)$

   if ($\hat{c}_j \leq 0$ or $\bar{b} < b$) **then** $x_j(\lambda) := 1$

   $L(\lambda) := L(\lambda) + \hat{c}_j$

   $\bar{b} := \bar{b} + 1$

   else $x_j(\lambda) := 0$

* Subgradient Procedure (initially : $\lambda := 0$, $LB := 0$)

$O(n)$ $\begin{cases} s(\lambda) := a - \sum_{j=1}^{n} p_j x_j(\lambda) ; \; LB := \max\{LB, \lceil LB(\lambda) \rceil\} ; \\ \lambda := \max\{0, \lambda + t\, s(\lambda)\} \quad (t = \text{step size}) \end{cases}$

** The relaxed problem could be solved in $O(n)$ time by considering the problem as a KP01-Min (with "unitary" weights), and solving its continuous relaxation by applying the Balas-Zemel algorithm (see the solution of 1.1.b)

1.1.b) Lagrangian Relaxation of constraint (b) ($\sigma \geq 0$)

$$L(\sigma) = \min \sum_{j=1}^{n} c_j x_j + \sigma \left(b - \sum_{j=1}^{n} x_j\right)$$

$$L(\sigma) = \sigma b + \min \sum_{j=1}^{n} \tilde{c}_j x_j \quad (\tilde{c}_j := c_j - \sigma;\ j = 1, \dots, n)$$

$O(n)$

s.t.

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

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

* The Relaxed Problem is a KP01-Min (possibly with negative costs). Solution of the relaxed problem:

$N := \{1, \dots, n\}$

$O(n)$ (1) for $j \in N$ do

if $\tilde{c}_j \leq 0$ then $x_j(\sigma) := 1$ else $x_j(\sigma) := 0$.

2) $S := \{j \in N : x_j(\sigma) = 0\}$; $\bar{a} := a - \sum_{j \in N \setminus S} p_j$; $O(n)$

if $\bar{a} \leq 0$ then go to step 3

$O(n)$ Solve the Continuous Relaxation of the KP01-Min corresponding to the items in S (having "costs" $\tilde{c}_j$ and "weights" p_j) with "threshold" $\bar{a}$, by applying the Balas-Zemel algorithm.

Let $x_j(\sigma)$ the corresponding optimal solution.

3) $L(\sigma) := \sigma b + \sum_{j=1}^{n} \tilde{c}_j x_j(\sigma)$

* Subgradient Procedure (initially $\sigma := 0$, $LB := 0$).

$s(\sigma) := b - \sum_{j=1}^{n} x_j(\sigma)$; $LB := \max\{LB, \lceil L(\sigma) \rceil\}$;

$\sigma := \max\{0, \sigma + t\, s(\sigma)\}$

1.1.c) Lagrangian Relaxation of (a) and (b) ...

1.2) Surrogate Relaxation of constraints (a), (b)

(1) $L(\alpha, \gamma) = \min \sum_{j=1}^{n} c_j x_j$

s.t. $(\alpha \geq 0, \gamma \geq 0, \alpha + \gamma > 0)$

$$\alpha \sum_{j=1}^{n} p_j x_j + \gamma \sum_{j=1}^{n} x_j \geq \alpha a + \gamma b \qquad (*)$$

(2) $\sum_{j=1}^{n} \tilde{p}_j x_j \geq d$ (with $\tilde{p}_j := \alpha p_j + \gamma (\geq 0)$ $j = 1, \dots, n$; $d := \alpha a + \gamma b$) $O(n)$

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

The Relaxed Problem (1), (2) and (3) is a KP01-Min.

* Consider the Continuous Relaxation of the Relaxed Problem and solve it by applying the Balas-Zemel algorithm ($O(n)$ time): optimal solution $x_j(\alpha, \gamma)$

* Subgradient Procedure (Initially: $\alpha := 1$, $\gamma := 1$, $LB := 0$)

$s(\alpha) := a - \sum_{j=1}^{n} p_j x_j (\alpha, \gamma)$

$s(\gamma) := b - \sum_{j=1}^{n} x_j (\alpha, \beta)$; $LB := \max\{LB, \lceil L(\alpha, \gamma) \rceil\}$

$\alpha := \max\{0, \alpha + t\, s(\alpha)\}$

$\gamma := \max\{0, \gamma + t\, s(\gamma)\}$

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) Determine "good" Lagrangian Lower Bounds which can be computed through procedures having time complexity $O(r + n)$, with $r = |V_1| + |V_2| + \ldots + |V_m|$, and describe the corresponding subgradient optimization procedures.

2) Determine a "good" Surrogate Lower Bound which can be computed through a procedure having time complexity $O(r + n)$, and describe the corresponding subgradient optimization procedure.

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.

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

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

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

Constraints (a) can also be written as:

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

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

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

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

$O(\bar{n})\{$ $n := \bar{n}$; $d := \bar{b}$; $(c_j) := (\bar{c}_j)$; $(t_j) := (\bar{w}_j)$; $m := 1$; $V_1 = \{1, \dots, \bar{n}\}$

The optimal solution of P is also optimal for KP01-Min.

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

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

$O(\bar{n})\{$ • $PP \propto F\text{-}P2$; ... ; $m := 1$, $V_1 := \{1, \dots, \bar{n}\}$;

PP is feasible if and only if $F\text{-}P$ has a solution

$\Rightarrow F\text{-}P2$ is NP-Hard

<u>NOTE</u> In the mathematical model it is not allowed to use matrix (a_{ij}), since its definition requires a time $O(mn) > O(z+n)$.

* Given the subsets V_i $(i=1,\dots,m)$, define, for each route j $(j=1,\dots,n)$, the subset:

$$B_j := \{ i : j \in V_i,\ i=1,\dots,m \}$$

containing the customers which can be served by j:

1) <u>for</u> $j=1,\dots,n$ <u>do</u> $B_j := \emptyset$; $\qquad O(n)$

2) <u>for</u> $i=1,\dots,m$ <u>do</u>
  <u>for</u> $j \in V_i$ <u>do</u> $B_j := B_j \cup \{i\}$ $\qquad \Big\} O(z)$

We can assume $n \le z$, $m \le z$.

3.1.2) <u>Lagrangian Relaxation</u> of <u>constraints (a)</u> $(u_i \ge 0,\ i=1,\dots,m)$

$$L(u) = \min \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} u_i \Big(1 - \sum_{j\in V_i} x_j\Big)$$

$$\Big(\sum_{i=1}^{m} u_i \sum_{j\in V_i} x_j = \sum_{i=1}^{m}\sum_{j=1}^{n} u_i a_{ij} x_j = \sum_{j=1}^{n}\sum_{i=1}^{m} u_i a_{ij} x_j =$$

$$= \sum_{j=1}^{n} x_j \sum_{i=1}^{m} a_{ij} u_i = \sum_{j=1}^{n} x_j \sum_{i\in B_j} u_i \Big)$$

$$L(u) = \sum_{i=1}^{m} u_i + \min \sum_{j=1}^{n} \bar{c}_j x_j \qquad \Big(\text{with } \bar{c}_j := c_j - \sum_{i\in B_j} u_i \quad j=1,\dots,n$$

<u>$O(z)$ time</u>

$$\text{s.t.} \quad \sum_{j=1}^{n} t_j x_j \ge d$$

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

<u>Solution of the relaxed problem</u>:

Similar to the solution of 1.1.b: time $O(n)$

<u>Subgradient Procedure</u>

Similar to the Subgradient Procedure used for the Lagrangian Relaxation of the Set Covering Problem.

3.1.b) Lagrangian Relaxation of constraint (b) $(\lambda \geq 0)$

$$L(\lambda) = \min \sum_{j=1}^{n} c_j x_j + \lambda \left(d - \sum_{j=1}^{n} t_j x_j \right)$$

$$L(\lambda) = \lambda d + \min \sum_{j=1}^{n} \tilde{c}_j x_j \quad (\text{with } \tilde{c}_j := c_j - \lambda t_j \; ; \; j = 1, \dots, n)$$

$O(n)$

s.t.

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

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

* Solution of the Relaxed Problem

1) for $j = 1, \dots, n$ do if $\tilde{c}_j \leq 0$ then $x_j := 1$ else $x_j := 0$

$O(n)$

2) With respect to the variables x_j such that $\tilde{c}_j > 0$, and the constraints (a) not satisfied by the variables x_j such that $\tilde{c}_j \leq 0$, the Relaxed Problem is a "Set Covering Problem" with $O(n)$ columns and $O(z)$ elements equal to 1 in the coefficient matrix.

Any Relaxation of the Set Covering Problem having time complexity $O(z+n)$ can be used:

- Lagrangian Relaxation of constraints (a);
- Continuous Relaxation of the Surrogate Relaxation of constraints (a) (by applying the B-Z algorithm).

* Subgradient Optimization Procedure

similar to that considered for 1.1.a, by taking into account the relaxation utilized for constraints (a).

3.1.c) Lagrangian Relaxation of constraints (a) and (b).

3.2 Surrogate Relaxation of constraints (a) and (b)

$(u_i \geq 0 \quad i = 1,\dots, m; \; \lambda \geq 0)$

$$L(u, \lambda) = \min \sum_{j=1}^{n} c_j x_j \qquad (c)$$

s.t.

$$\sum_{i=1}^{m} u_i \sum_{j \in V_i} x_j + \lambda \sum_{j=1}^{n} t_j x_j \geq \sum_{i=1}^{m} u_i + \lambda d$$

$$\Downarrow$$

$$\sum_{j=1}^{n} \Big(\sum_{i \in B_j} u_i + \lambda t_j \Big) x_j \geq \sum_{i=1}^{m} u_i + \lambda d$$

$$\Downarrow$$

$$\sum_{j=1}^{n} \tilde{p}_j x_j \geq \tilde{d} \qquad (a\text{-}b)$$

with $\tilde{p}_j := \sum_{i \in B_j} u_i + \lambda t_j \quad j = 1,\dots, n; \; \tilde{d} := \sum_{i=1}^{m} u_i + \lambda d$

$O(\geq)$

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

* The Relaxed Problem (c), (a-b), (d) define a KP01-Min.
* Continuous Relaxation of this KP01-Min by applying the Balas-Zemel algorithm.
* Subgradient Optimization Procedure similar to that considered in 1.2.

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) Consider first the mathematical model corresponding to the surrogate relaxation of the constraints associated with point a) with surrogate multipliers all equal to 1. Then, starting from this surrogate relaxation, determine a "good" Lagrangian Upper Bound which can be computed through a procedure having time complexity O(n + m), and describe the corresponding subgradient optimization procedure.

2) Determine a "good" Lagrangian Upper Bound which can be computed through a procedure having time complexity O(m * n), and describe the corresponding subgradient optimization procedure. Assume log(m) <= n.

Exercise 4

4.1)

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

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

s.t.

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

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

$$\sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij} < K \Rightarrow \sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij} \le \overline{K} \qquad (c)$$
(with $\overline{K} = K-1$)

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

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: $\bar{n}, (w_j), \bar{C} \Rightarrow \bar{n}$

$O(\bar{n})$ $\{$ $n := 1$; $m := \bar{n}$; $p_j := w_j$ $(j=1,\dots,m)$; $a_1 := \bar{C}$; $K := m+1$

The optimal solution of P is also optimal for SSP.

4.3.1) $F\text{-}P \in P$: 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

4.1) Surrogate Relaxation of constraints (a) with surrogate multipliers equal to 1.

The n constraints (a) are replaced by

$$\sum_{i=1}^{n}\sum_{j=1}^{m} p_j x_{i,j} \le \sum_{i=1}^{n} a_i$$

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

time $O(n)$

Set $y_j = \sum_{i=1}^{n} x_{i,j}$ $(j = 1, \dots, m)$

The mathematical model becomes:

$$\bar{z} = \max \sum_{j=1}^{m} p_j y_j \qquad (d')$$

s.t.

$$\sum_{j=1}^{m} p_j y_j \le C \qquad (a')$$

$$\sum_{j=1}^{m} y_j \le \bar{K} \qquad (\text{with } \bar{K} = K-1) \qquad (c')$$

$$y_j \in \{0, 1\} \qquad j = 1, \dots, m \qquad (e')$$

The problem (d'), (a'), (c') and (e') is "equivalent" to the problem considered in Exercise 1. As a consequence, it is possible to utilize the Lagrangian Relaxations considered in 1.1.a or 1.1.b (time $O(m)$ by applying the Balas-Zemel algorithm) and the corresponding Subgradient Optimization procedures.

4.2) Lagrangian Relaxation of constraints (b) and (c)

$(\lambda_j \geq 0 \quad j=1,\dots,m; \; \sigma \geq 0)$

$$U(\lambda,\sigma) = \max \sum_{i=1}^{n}\sum_{j=1}^{m} p_j x_{ij} + \sum_{j=1}^{m} \lambda_j \left(1 - \sum_{i=1}^{n} x_{ij}\right) + \sigma \left(\bar{k} - \sum_{i=1}^{n}\sum_{j=1}^{m} x_{ij}\right)$$

$$U(\lambda,\sigma) = \sum_{j=1}^{m} \lambda_j + \sigma\bar{k} + \max \sum_{i=1}^{n}\sum_{j=1}^{n} \bar{p}_j x_{ij} \qquad (d'')$$

(with $\bar{p}_j = p_j - \lambda_j - \sigma \quad j=1,\dots,m$) time $O(m)$

s.t. (a) and (e).

Solution

1) $\bar{M} := \emptyset$

for $j=1,\dots,m$ **if** $\bar{p}_j \leq 0$ **then** $x_{ij} := 0$ for $i=1,\dots,n$

else $\bar{M} := \bar{M} \cup \{j\}$

2) The problem defined by (d''), (a) and (e) corresponding to the items $j \in \bar{M}$ asks for the solution of n independent KP01s, one for each vehicle i $(i=1,\dots,n)$ of the form:

$$\bar{z}_i = \max \sum_{j\in\bar{M}} \bar{p}_j \bar{x}_j^{(i)}$$

s.t.

$$\sum_{j\in\bar{M}} p_j \bar{x}_j^{(i)} \leq a_i$$

$$\bar{x}_j^{(i)} \in \{0,1\} \qquad j \in \bar{M} \quad (\text{with } |\bar{M}| \leq m)$$

3) $$U(\lambda,\sigma) = \sum_{j=1}^{m} \lambda_j + \sigma\bar{k} + \sum_{i=1}^{n} \bar{z}_i$$

* By considering the Continuous Relaxations of the n KP01s, and by applying for their solutions the Balas-Zemel algorithm, the global time is $O(m*n)$.

* Subgradient Optimization Procedure for 4.2.

- **for** $j = 1, \dots, m$ **do** $\lambda_j := 0$; $\sigma := 0$; $UB := +\infty$.

- $s(\lambda_j) := 1 - \sum_{i=1}^{n} x_{ij}$ $\quad j = 1, \dots, m$

  $s(\sigma) := k - \sum_{i=1}^{n} \sum_{j=1}^{m} x_{ij}$

- $UB := \min \{UB, U(\lambda, \sigma)\}$;

- $\lambda_j := \max \{0, \lambda_j - t\, s(\lambda_j)\}$ $\quad j = 1, \dots, m$

- $\sigma := \max \{0, \sigma - t\, s(\sigma)\}$

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) Determine "good" Lagrangian Lower Bounds which can be computed through procedures having time complexity $O(n * n)$ (some constraints could be eliminated), and describe the corresponding subgradient optimization procedures.

2) As at point 1) in the case where it is imposed that the elementary circuit visits exactly one vertex of each of the K regions.

EXERCISE 5

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

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

* $ATSP \propto P$

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

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

• 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 $c_{ij} := \infty$

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

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

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

s.t.

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

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

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

$$\sum_{i \in S} \sum_{j \in S} x_{ij} \le |S| - 1 \quad \forall S \subseteq V - \{1\}, S \ne \emptyset \quad (e)$$

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

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

5.1.2) Lagrangian Relaxation of constraints (c) with elimination of constraints (e).

(λ_i can take any value, $i=1,\dots,n$)

$$L(\lambda) = \min \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}\,x_{ij} + \sum_{i=1}^{n}\lambda_i\left(\sum_{j=1}^{n} x_{ij} - \sum_{j=1}^{n} x_{ji}\right)$$

$$L(\lambda) = \min \sum_{i=1}^{n}\sum_{j=1}^{n} \tilde{c}_{ij}\,x_{ij} \qquad (a')$$

with $\tilde{c}_{ij} := c_{ij} + \lambda_i - \lambda_j \quad i=1,\dots,n;\ j=1,\dots,n \quad O(n^2)$

$$\left(\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\,x_{ji} = \underbrace{\sum_{j=1}^{n}\sum_{i=1}^{n}\lambda_j\,x_{ij}}_{\text{exchange "i" and "j"}} = \sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_j\,x_{ij}\right)$$

s.t.

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

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

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

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

* Because of constraint (b), for each vertex i ($i=1,\dots,n$) at most one arc (i,j) ($j=1,\dots,n$) can be used:

for $i=1,\dots,n$ do $\bar{c}_i := \tilde{c}_{i,j(i)} := \min_{j=1,\dots,n}\{\tilde{c}_{ij}\}$ $\quad O(n^2)$

$$L(\lambda) = \min \sum_{i=1}^{n}\sum_{j=1}^{n} \bar{c}_i\,x_{ij}$$

$$L(\lambda) = \min \sum_{i=1}^{n} \bar{c}_i \sum_{j=1}^{n} x_{ij}$$

*
$$L(\lambda) = \min \sum_{i=1}^{n} \bar{c}_i\,y_i$$

s.t.

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

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

* Solution of the relaxed problem (5.1.2)

- **for** $i = 1, \dots, n$ **do**
  $y_i := 0$ $\qquad O(n^2)$
  for $j = 1, \dots, n$ **do**
  $x_{ij} := 0$;
- **for** $i = 1, \dots, n$ **do**
  if $\bar{c}_i \le 0$ **then** $y_i := 1$, $x_{i,j(i)} := 1$ $\qquad O(n)$
- **for** $h = 1, \dots, k$ **do**
  if $\sum_{i \in R_h} y_i = 0$ **then**
  let ℓ such that $\bar{c}_\ell = \min\{\bar{c}_i : i \in R_h\}$
  $y_\ell := 1$; $x_{\ell, j(\ell)} := 1$ $\qquad O(n)$

* Subgradient Optimization Procedure (5.1.3)

for $i = 1, \dots, n$ **do** $\lambda_i := 0$; $LB := 0$

- $\delta(\lambda_i) := \sum_{j=1}^{n} x_{i,j} - \sum_{j=1}^{n} x_{j,i}$ $\qquad i = 1, \dots, n$ $\qquad O(n^2)$
- $LB := \max\{LB, L(\lambda)\}$
- $\lambda_i := \lambda_i + t\, \delta(\lambda_i)$ $\qquad i = 1, \dots, n$

If $\delta(\lambda_i) < 0$, we have $\sum_{j=1}^{n} x_{i,j} < \sum_{j=1}^{n} x_{j,i}$

(number of arcs leaving vertex i < number of arcs entering vertex i)
then λ_i decreases $\Rightarrow$ the cost $\bar{c}_{ij}$ of the arcs leaving vertex i decreases ($\bar{c}_{ij} = c_{ij} + \lambda_i - \lambda_j$)

$\Downarrow$

a larger number of arcs leaving vertex i will be used

$\Downarrow$

constraint (C) for vertex i less violated

5.1.b) Lagrangian Relaxation of constraints (b) and (c) with elimination of constraints (e)

(α_i and λ_i can take any value, $i=1,\dots,n$)

$$L(\alpha,\lambda)=\sum_{i=1}^{n}\sum_{j=1}^{n}c_{i,j}x_{i,j}+\sum_{i=1}^{n}\alpha_i\left(\sum_{j=1}^{n}x_{i,j}-y_i\right)+$$
$$+\sum_{i=1}^{n}\lambda_i\left(\sum_{j=1}^{n}x_{i,j}-\sum_{j=1}^{n}x_{j,i}\right)$$

$$L(\alpha,\lambda)=\min\sum_{i=1}^{n}\sum_{j=1}^{n}\hat{c}_{i,j}x_{i,j}-\sum_{i=1}^{n}\alpha_i y_i \qquad (a'')$$

(with $\hat{c}_{i,j}=c_{i,j}+\alpha_i+\lambda_i-\lambda_j \quad i=1,\dots,n;\ j=1,\dots,n \quad O(n^2)$

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

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

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

* 2 independent problems: (y_i), $(x_{i,j})$. $L(\alpha,\lambda)=L(x)-L(y)$

- variables (y_i):

$$L(y)=\max\sum_{i=1}^{n}\alpha_i y_i$$

s.t. $\sum_{i\in R_h} y_i \geq 1 \quad h=1,\dots,K\ ;\ y_i\in\{0,1\}\ i=1,\dots,n$

<u>for</u> $i=1,\dots,n$ <u>do</u>

  <u>if</u> $\alpha_i\geq 0$ <u>then</u> $y_i:=1$ <u>else</u> $y_i:=0$;

<u>for</u> $h=1,\dots,K$ do

  <u>if</u> $\sum_{i\in R_h} y_i=0$ <u>then</u> let ℓ such that $\alpha_\ell=\max\{\alpha_i : i\in R_h\}$

  $y_\ell:=1$ $\qquad O(n)$

- variables $(x_{i,j})$:

<u>for</u> $i=1,\dots,n$ <u>do</u>

  <u>for</u> $j=1,\dots,n$ <u>do</u>

    <u>if</u> $\hat{c}_{i,j}\leq 0$ <u>then</u> $x_{i,j}:=1$ <u>else</u> $x_{i,j}:=0$ $\qquad O(n^2)$

* Subgradient Optimization Procedure (5.1.3)

as for 5.1.2 with

$$s(\alpha_i) := \sum_{j=1}^{n} x_{ij} - y_i \qquad i = 1,\dots,n$$

$$\alpha_i := \alpha_i + t\, s(\alpha_i) \qquad i = 1,\dots,n$$

$$LB := \max\{LB,\ L(\alpha,\lambda)\}$$

5.2) Constraints (d) become:

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

* Consider the Lagrangian Relaxation of constraints (c) and elimination of constraints (e)
- The Relaxed Problem is defined by (a), (d'), (e), (f)

* Solution of the Relaxed Problem

- **for** $i = 1,\dots,n$ **do** $\qquad O(n^2)$
  - $y_i := 0$
  - **for** $j = 1,\dots,n$ **do**
    - $x_{i,j} := 0$

- **for** $h = 1,\dots,K$ **do**
  - let ℓ such that $\bar{c}_\ell = \min\{\bar{c}_i : i \in R_h\}$
  - $y_\ell := 1$; $x_{\ell,j(\ell)} := 1$ $\qquad O(n)$

* Subgradient Optimization Procedure similar to that considered for 5.1.2.

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) Determine a "good" Upper Bound obtained through a Lagrangian relaxation of the constraints b) and c), and which can be computed through a procedure having time complexity $O(n * n * n)$.

2) Describe the subgradient optimization procedure corresponding to the Upper Bound defined at point 1) and having time complexity $O(n * n * n)$.

3) Determine an additional "good" Lagrangian Upper Bound obtained through a Lagrangian relaxation of the constraints b) and c) (and possibly of other constraints), and which can be computed through a procedure having time complexity $O(n * n)$.

4) Describe the subgradient optimization procedure corresponding to the Upper Bound defined at point 3) and having time complexity $O(n * n * n)$.

EXERCISE 8

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

ATSP_{max}

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

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

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

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

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

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

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

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 $(n-1)$ descendant nodes.

• ATSP-Max $\propto P$: size of ATSP-Max: n, (p_{ij}): n^2

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

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

8.1) <u>Lagrangian Relaxation of constraints (5) and (6)</u>

$(\lambda \geq 0, \sigma \geq 0)$

$$U(\lambda, \sigma) = \max \sum_{i=1}^{n} \sum_{j=1}^{n} p_{ij} x_{ij} + \lambda \left(d - \sum_{i=1}^{n} \sum_{j=1}^{n} t_{ij} x_{ij} \right) +$$

$$+ \sigma \left(\sum_{(i,j) \in S} x_{ij} - \sum_{(i,j) \in T} x_{ij} \right)$$

$$U(\lambda, \sigma) = \lambda d + \max \sum_{i=1}^{n} \sum_{j=1}^{n} (p_{ij} - \lambda t_{ij}) x_{ij} + \sigma \sum_{(i,j) \in S} x_{ij} - \sigma \sum_{(i,j) \in T} x_{ij}$$

$$U(\lambda, \sigma) = \lambda d + \max \sum_{i=1}^{n} \sum_{j=1}^{n} \hat{p}_{ij} x_{ij} \qquad (1'')$$

with

$$O(n^2) \begin{cases} \hat{p}_{ij} = p_{ij} - \lambda t_{ij} + \sigma & (i,j) \in S \\ \hat{p}_{ij} = p_{ij} - \lambda t_{ij} - \sigma & (i,j) \in T \\ \hat{p}_{ij} = p_{ij} - \lambda t_{ij} & (i,j) \in A \setminus (S \cup T) \end{cases}$$

s.t. (2), (3), (4) and (7).

The Relaxed Problem is an ATSP (<u>maximization version</u>) with "cost" of arc (i,j) equal to $\hat{p}_{ij}$ for each $(i,j) \in A$.

8.2) <u>Elimination of constraints</u> (4):

The Relaxed Problem is an Assignment Problem (maximization version): time $O(n^3)$.

8.3) <u>Elimination (or Lagrangian Relaxation) of constraints (2)</u>:

The Relaxed Problem is an "r-Longest Spanning Arborescence Relaxation" (with any vertex of V as vertex r): time $O(n^2)$.

8.2) <u>Subgradient Optimization Procedure</u>: 8.1

- $$\mathcal{S}(\lambda) = d - \sum_{i=1}^{n}\sum_{j=1}^{n} t_{ij}\, x_{ij}$$

  $$\mathcal{S}(G) = \sum_{(i,j)\in S} x_{ij} - \sum_{(i,j)\in T} x_{ij}$$

- Initialization: $\lambda := 0$, $G := 0$, $UB = +\infty$.
- $UB := \min\{UB, U(\lambda, G)\}$
- $\lambda := \max\{0, \lambda - t\,\mathcal{S}(\lambda)\}$
- $G := \max\{0, G - t\,\mathcal{S}(G)\}$

Maximum number of iteration = constant.

8.4) <u>Subgradient Optimization Procedure</u>: 8.3

- If constraints (2) have been eliminated: see 8.2.
- If constraints (2) have been relaxed in a Lagrangian way, the corresponding Lagrangian Multipliers are updated as done for the "2-SSA Relaxation" considered for the ATSP.

* Maximum number of iterations = $O(n)$.

Exercise 9

Given n "depots" and m "customers": each customer i (i = 1, ..., m) has a non-negative "potential profit" p_i. Each depot j (j = 1, ..., n) has a non-negative "cost" c_j and is able to "serve" a subset R_j 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, ..., n and i = 1, ..., 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.

Let h denote the number of elements of the matrix $(a_{i,j})$ having value equal to 1 (with h >= n, h >= m). $h = \sum_{j=1}^{n} |R_j|$

Determine "good" Upper Bounds based on the following relaxations:

1) Three different Lagrangian relaxations which can be computed through procedures having time complexity O(h).

2) A Surrogate relaxation for the particular case in which all the customers must be served, and which can be computed through a procedure having time complexity O(h).

3) For at least two of the relaxations considered at point 1), describe the corresponding subgradient optimization procedures.

Exercise 9

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

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

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

s.t.

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

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

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

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

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

$$* \; B_i = \{ j : i \in R_j \} \qquad i = 1, \dots, m.$$

$$\sum_{i=1}^{m} |B_i| = h$$

$O(h)$

See Exercise 3

9.1.2) Lagrangian Relaxation of constraints (2') and (3).

($\lambda_i \geq 0 \quad i = 1, \dots, m$; $\sigma \geq 0$)

$$U(\lambda, \sigma) = \max\left(\sum_{i=1}^{m} p_i y_i - \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} \lambda_i \left(\sum_{j \in B_i} x_j - y_i\right) + \sigma\left(d - \sum_{j=1}^{n} x_j\right)\right)$$

- $U(\lambda, \sigma) = \sigma d + \max\left(\sum_{i=1}^{m} \hat{p}_i y_i - \sum_{j=1}^{n} \hat{c}_j x_j\right)$

with: $\hat{p}_i := p_i - \lambda_i \qquad i = 1, \dots, m \qquad O(m)$

$\hat{c}_j := c_j - \sum_{i \in R_j} \lambda_i + \sigma \qquad j = 1, \dots, n \qquad O(n)$

- s.t. (4), (5) and (6).

* The Relaxed Problem can be decomposed into two independent subproblems: (y_i), (x_j). $U(\lambda, \sigma) = \sigma d + U1 + U2$

- Variables (y_i):

$U1 = \max \sum_{i=1}^{m} \hat{p}_i y_i$

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

Solution:

for $i = 1, \dots, m$ **do** **if** $\hat{p}_i > 0$ **then** $y_i = 1$ **else** $y_i = 0$ $\}$ $O(m)$

- Variables (x_j):

$U2 = \max\left(-\sum_{j=1}^{n} \hat{c}_j x_j\right) = -\min \sum_{j=1}^{n} \hat{c}_j x_j$

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

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

The problem to be solved is similar to that considered in 1.1. Time $O(n)$

9.1.b) Lagrangian Relaxation of constraints (2') and (4)

($\lambda_i \geq 0 \quad i=1,\dots,m; \ \alpha \geq 0$)

$$U(\lambda,\alpha) = \max\left(\sum_{i=1}^{m} p_i y_i - \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} \lambda_i \left(\sum_{j=1}^{n} x_j - y_i\right) + \alpha\left(\sum_{j=1}^{n} c_j x_j - b\right)\right)$$

$$U(\lambda,\alpha) = -\alpha b + \max\left(\sum_{i=1}^{m} \hat{p}_i y_i - \sum_{j=1}^{n} \hat{c}_j x_j\right)$$

with: $\hat{p}_i := p_i - \lambda \qquad i=1,\dots,m \qquad O(m)$

$\hat{c}_j := c_j - \sum_{i \in R_j} \lambda_i - \alpha c_j \qquad j=1,\dots,n \qquad O(n)$

- s.t. (3), (5), (6)

* The Relaxed Problem can be decomposed into two independent subproblems: (y_i), (x_j). $U(\lambda,\alpha) = -\alpha b + U1 + U2$

- Variables (y_i): see 9.1.a $\qquad O(m)$
- Variables (x_j)

$$U2 = \max\left(-\sum_{j=1}^{n} \hat{c}_j x_j\right)$$

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

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

Solution

$\bar{N} := \emptyset$; for $j=1,\dots,n$ do if $\hat{c}_j \geq 0$ then $x_j := 0$ else $\bar{N} := \bar{N} \cup \{j\}$.

solve, by applying the Balas-Zemel algorithm, the continuous relaxation of the KP01 given by:

$$U2 = \max \sum_{j \in \bar{N}} (-\hat{c}_j) x_j$$

s.t. $\sum_{j \in \bar{N}} x_j \leq d$; $x_j \in \{0,1\} \quad j=1,\dots,n \qquad \} O(n)$ time

9.1.c) Lagrangian Relaxation of constraints (2'), (3), (4)

$(\lambda_i \geq 0 \quad i=1,\dots,m;\ \sigma \geq 0;\ \alpha \geq 0)$

$$U(\lambda, \alpha, \sigma) = \max\left(\sum_{i=1}^{m} p_i y_i - \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} \lambda_i \left(\sum_{j \in E_i} x_j - y_i\right) + \right.$$
$$\left. + \sigma\left(d - \sum_{j=1}^{n} x_j\right) + \alpha\left(\sum_{j=1}^{n} c_j x_j - b\right)\right)$$

$$U(\lambda, \alpha, \sigma) = \sigma d - \alpha b + \max\left(\sum_{i=1}^{m} \tilde{p}_i y_i - \sum_{j=1}^{n} \tilde{c}_j x_j\right)$$

with: $\tilde{p}_i := p_i - \lambda \qquad i=1,\dots,m \qquad O(m)$

$\tilde{c}_j := c_j - \sum_{i \in R_j} \lambda_i - \alpha c_j + \sigma \qquad j=1,\dots,n \qquad O(n)$

s.t. (5) and (6)

* The Relaxed Problem can be decomposed into two independent subproblems: (y_i), (x_j). $U(\lambda, \alpha, \sigma) = \sigma d - \alpha b + U1 + U2$.

- Variables (y_i): see 9.1.a). $\qquad O(m)$
- Variables (x_j):

for $j=1,\dots,n$ do if $\tilde{c}_j \leq 0$ then $x_j := 1$ else $x_j := 0$ $\}$ $O(n)$

9.2) Surrogate Relaxation with $y_i = 1 \quad i = 1, \ldots, m$

$$U(\alpha, \beta, \gamma) = \max\left(-\sum_{j=1}^{n} c_j x_j\right) = -\min \sum_{j=1}^{n} c_j x_j \qquad (1')$$

s.t.

$$\sum_{j \in B_i} x_j \geq 1 \qquad i = 1, \ldots, m \qquad (2'')$$

$$\sum_{j=1}^{n} x_j \leq d \quad (2) \qquad (3)$$

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

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

- Replace (3) with

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

* Surrogate Relaxation of $(2'')$, $(3')$, (4)

$(\alpha_i \geq 0 \;\; i = 1, \ldots, m;\; \beta \geq 0;\; \gamma \geq 0)$

$$\sum_{i=1}^{m} \alpha_i \sum_{j \in B_i} x_j - \beta \sum_{j=1}^{n} x_j + \gamma \sum_{j=1}^{n} c_j x_j \geq \sum_{i=1}^{m} \alpha_i - \beta d + \gamma b$$

$$\Downarrow$$

$$\sum_{j=1}^{n} \tilde{w}_j x_j \geq \tilde{c} \qquad (7)$$

with $\tilde{w}_j := \sum_{i \in R_j} \alpha_i - \beta + \gamma c_j$; $\tilde{c} := \sum_{i=1}^{m} \alpha_i - \beta d + \gamma b$ $\{O(n)$

- The Relaxed Problem $(1')$, (7) and (5) is a KP01-Min (possibly with negative $(\tilde{w}_j)$ and $\tilde{c}$).
- If $\tilde{c} \leq 0$ then for $j = 1, \ldots, n$ do $x_j := 0$, STOP.
- $\bar{N} := \emptyset$; for $j = 1, \ldots, n$ do if $\tilde{w}_j \leq 0$ then $x_j := 0$ else $\bar{N} := \bar{N} \cup \{j\}$
- Replace (7) with $\sum_{j \in \bar{N}} \tilde{w}_j x_j \geq \tilde{c}$.
- Continuous Relaxation of the KP01-Min with the B-Z algor. then