Investigação Operacional, Vol. 8, Nº 2 (Dezembro/1988) 3-11

PRIMAL AND DUAL GREEDY HEURISTICS FOR THE GENERALIZED SET COVERING PROBLEM

José Paixão
Faculdade de Ciências, Universidade de Lisboa
Av. 24 de Julho, 134-5º, 1300 Lisboa

Margarida Pato
Instituto Superior de Economia, Universidade Técnica de Lisboa
R. Miguel Lupi, 20, 1200 Lisboa

Abstract
This paper reports on the development of primal and dual greedy heuristics for the generalized set covering problem (GSCP).
The primal heuristics provide a feasible solution and, consequently, an upper bound on the optimum for the GSCP. Dual based heuristics are used for obtaining and improving lower bounds at optimal value. Both, the primal and dual procedures are described in this paper and the corresponding computational complexity is studied.
Moreover, we present empirical results, obtained from computational experience with 34 instances of the GSCP. The test problems are related to the scheduling of bus drivers at Rodoviária Nacional a large transport operator in Portugal. In fact, this specific integer program, GSCP, has been widely used in crew scheduling applications.
These computational tests show that a combined primal-dual greedy heuristic procedure is a reasonably accurate and fast tool at least to tackle with bus driver scheduling GSCP instances. Taking this into account, another procedure, embedding the primal-dual greedy heuristics in a lagrangean relaxation based method, has been deviced for the GSCP and is presented in a different paper.

Keywords: Heuristics, generalized set covering.

Research for this paper was supported by Instituto Nacional de Investigação Científica and Junta Nacional de Investigação Científica e Tecnológica.

1. INTRODUCTION

The generalized set covering problem can be stated as the following integer program:

$$\text{(GSCP)} \quad \min \sum_{j \in N} c_j x_j$$

$$\text{s.t.} \quad \sum_{j \in N} a_{ij} x_j \geq b_i \qquad (i \in M) \tag{1.1}$$

$$0 \leq x_j \leq h_j \qquad (j \in N) \tag{1.2}$$

$$x_j \text{ integer} \qquad (j \in N), \tag{1.3}$$

where a_{ij}, for $i \in M$ and $j \in N$, are equal to 0 or 1. When the integer variable-bounds h_j are all equal to 1 the GSCP has been referred in the literature as the multiple set covering problem or the multicovering problem ([3]). When, on top of this, the integer values b_i are all equal to 1, the GSCP becomes the well known set covering model ([1], [2], [4]) where constraints (1.2) and (1.3) are replaced by $x_j \in \{0, 1\}$ $(j \in N)$.

Like the classical set covering, the generalized and the multiple covering models are closely related to some real life situations. Multicovering problems arise in communication or distribution problems where reliability is important. The generalized set covering problem is mainly related to personnel scheduling ([5] and [9]).

According to the usual, $\mathcal{V}$(GSCP) denotes the optimal value for GSCP and $\mathcal{F}$(GSCP) stands for the set of feasible solutions for GSCP.

The GSCP is well known as a NP-complete problem ([10]) and so, heuristic methods come as a reasonable way of obtaining an approximation interval for $\mathcal{V}$(GSCP).

In this paper, we present several heuristic techniques developed in order to produce bounds, both from above and from below, on the optimal value for the GSCP. Hence, the next section is devoted to describing primal greedy heuristics followed by local search for the GSCP which provide feasible solutions and an upper bound on $\mathcal{V}$(GSCP). In section 3, dual greedy heuristics are studied for the purpose of obtaining lower bounds on $\mathcal{V}$(GSCP) and, possibly, improving the upper bounds.

Results on the computational complexity for each one of the algorithms are also presented.

At last, in section 4, an empirical analysis obtained with a composed procedure embedding all the heuristics is reported.

The GSCP instances considered for the experience belong to a particular type arising in bus driver scheduling applications. For these problems, N is the index set for the feasible driver shifts or workdays, each one of them being assigned a cost, c_j. The set M corresponds to time periods requiring a minimal number of drivers defined through vector $b = (b_i)_{i \in M}$. An element a_{ij} is equal to 1 if period i is a working period of shift j and 0 otherwise. The particularity of the bus driver scheduling problem comes from the fact that each column $(a_{ij})_{i \in M}$ consists of one or two strings of consecutive ones.

The composed procedure proves reasonable efficiency for those real life applications. In fact it produced good upper bounds for the most of the test problems and even the optimum for some of them. However, the lower bounds were not so accurate, requiring further attempts to obtain improvements namely using lagrangean relaxation techniques as described in [8]. Besides the procedure is relatively fast when the algorithms, implemented in FORTRAN, run on a VAX 11/750 VMS (with FPA). Moreover, the primal-dual greedy procedure can easily be implemented on a PC as reported in [5].

2. PRIMAL GREEDY HEURISTICS WITH LOCAL SEARCH

Greedy heuristics have been studied in deep detail for the classic set covering model ([1]) and the multicovering version ([3]). An extension of those techniques for a generalized covering model related to scheduling problems has been presented in a previous paper ([9]).

As described in there, and following the general pattern, the greedy heuristics construct feasible solutions, step by step, selecting a row to be covered and, then, a variable to cover the unsatisfied demand for that row. Several row and column selection criteria can be combined for building up feasible solutions ([7], [8] and [9]) and consequently, producing upper bounds on the optimal value $\mathcal{V}$(GSCP). Relatively to the value given to the selected variable, different options can be taken too.

The primal greedy procedure is summarized below.

<u>Procedure</u> <u>PRIMAL GREEDY</u>
input: M, N, $(c_j)_{j\in N}$, $(h_j)_{j\in N}$, $(b_i)_{i\in M}$, $(a_{ij})_{i\in M,\ j\in N}$
output: $(x_j)_{j\in N}$, z_u [feasible solution for GSCP and upper bound on $\mathcal{V}$(GSCP)]

1. [Initializing]

$x_j := 0$ $(j\in N)$
$z_u := 0$
$N_i := \{j\in N : a_{ij} = 1\}$ $(i\in M)$ [indices for columns to cover row i]
$M_j := \{i\in M : a_{ij} = 1\}$ $(j\in N)$ [indices for rows to be covered by column j]

$p_j := \sum_{i\in M_j} b_i$ $(j\in N)$ [weight for variable j]

2. [Selecting a row index i* according to criterion g(i)]
$i^* := \arg \underset{i\in M}{\text{opt}}\ g(i)$

3. [Selecting a column index j* according to criterion f (c_j, p_j)]
$j^* := \arg \min_{j\in N_{i^*}} f(c_j, p_j)$

4. [Updating sets and values]
$\Delta_{j^*} := \min \{h_{j^*} - x_{j^*}, v(b_i, i\in M_{j^*})\}$ [the increase on the variable j* depends on its upper bound h_{j^*} and on the criterion $v(b_i)$]

$x_{j^*} := x_{j^*} + \Delta_{j^*}$; $z_u := z_u + c_{j^*} \Delta_{j^*}$
update M, [rows to cover]
N, [columns not fixed at their upper bounds]
$(b_i)_{i\in M}$, [uncovered demand for row i]
$(N_i)_{i\in M}$, $(M_j)_{j\in N}$ and $(p_j)_{j\in N}$

5. [Checking feasibility]
if $M \neq \Phi$ **then goto 2 endif**
stop

The computational complexity for this algorithm, considering any one of the criteria explained in previous works, is $O(\max(|M|, |N|) \sum_{i\in M, j\in N} a_{ij})$.

After obtaining a greedy feasible solution, local improvements on the upper bound can be attempted, amongst others, through a search procedure based on three main steps:

- decreasing redundant columns and producing a prime solution;
- replacing one column in the current solution by a cheaper one;
- replacing a pair of columns in the current solution by a single column which covers the same at less cost.

Follows the description of the local search procedure.

<u>Procedure</u> SEARCH
input: M, N, $(c_j, h_j, M_j)_{j\in N}$, $(b_i, N_i)_{i\in M}$, $(x_j)_{j\in N}$, z_u
output: $(x_j)_{j\in N}$, z_u [improved feasible solution and upper bound on $\mathcal{V}$(GSCP)]

1. [Obtaining a prime solution]
calculate $(s_i)_{i\in M}$ [overcovered demand for row i]

$z^\circ := z_u$

$\tilde{S} := \{j : \Delta_j := \min\{x_j, s_i \text{ over } i \in M_j\} \geq 0\}$ [indices of redundant columns]
while $\tilde{S} \neq \Phi$ **do**
  choose $\bar{j} \in \tilde{S}$ [$\bar{j}$ is selected for decrease]
  $x_{\bar{j}} := x_{\bar{j}} - \Delta_{\bar{j}}$; $z_u := z_u - c_{\bar{j}} \Delta_{\bar{j}}$
  update $(s_i)_{i \in M_{\bar{j}}}$, $(\Delta_j)_{j \in \tilde{S}}$ and $\tilde{S}$
  enddo

2. [Search for an improved solution changing one variable by another]
for $j := 1, \ldots, |N|$ such that $x_j > 0$ **do**
  find a variable $\bar{j}$ such that $c_{\bar{j}} < c_j$ and covers binding constraints of M_j
  calculate Δ
  $x_j := x_j - \Delta$; $x_{\bar{j}} := x_{\bar{j}} + \Delta$; $z_u := z_u - \Delta(c_j - c_{\bar{j}})$
  update $(s_i)_{i \in M}$
  enddo

3. [Search for an improved solution changing a pair of variables by another]
for $j_1 := 1, \ldots, (|N| - 1)$ such that $x_{j_1} > 0$ **do**
  for $j_2 := (j_1 + 1), \ldots, |N|$ **do**
    if $x_{j_2} > 0$ **and** $M_{j_1} \cap M_{j_2}$ has no binding constraints **then**
    find variable $\bar{j}$ such that, $c_{\bar{j}} < c_{j_1} + c_{j_2}$ and covers binding constraints of $M_{j_1} \cup M_{j_2}$
      calculate Δ
      $x_{j_\ell} := x_{j_\ell} - \Delta$ $(\ell = 1, 2)$; $x_{\bar{j}} := x_{\bar{j}} + \Delta$; $z_u := z_u - \Delta(c_{j_1} + c_{j_2} - c_{\bar{j}})$
      update $(s_i)_{i \in M}$
      endif
    enddo
  enddo

4. [Checking local optimization]
if $z_u \neq z^o$ **then** $(z^o := z_u$ **and goto 1) endif**
stop

As shown in [6], the algorithm is polynomial on |M| and |N| provided that the number of iterations for the local search phase is previously fixed. In fact, there one proves that the complexity of that local search procedure is given by $O(|N|^2 \sum_{i \in M, j \in N} a_{ij})$.

3. DUAL GREEDY HEURISTICS

Now, let us consider the linear program resulting from removing the integrality conditions, that is, the linear relaxation for GSCP:

$$(\overline{GSCP}) \quad \min \sum_{j \in N} c_j x_j$$

$$\text{s.t.} \quad \sum_{j \in N_i} x_j \geq b_i \qquad (i \in M) \tag{3.1}$$

$$0 \leq x_j \leq h_j \qquad (j \in N).$$

We denote by $v(\overline{GSCP})$ the optimal value for $\overline{GSCP}$ which has the following linear dual problem:

$$(\overline{DGSCP}) \quad \max \sum_{i \in M} b_i u_i - \sum_{j \in N} h_j v_j$$

$$\text{s.t.} \quad \sum_{i \in M_j} u_i - v_j \le c_j \qquad (j \in N) \tag{3.2}$$

$$v_j \ge 0 \qquad (j \in N)$$

$$u_i \ge 0 \qquad (i \in M).$$

The constraints (3.2) can be rewritten, as

$$c_j - \sum_{i \in M_j} u_i + v_j \ge 0 \qquad (j \in N) \tag{3.3}$$

with the left-hand-side being the linear reduced cost for the variable x_j.

It is well known that, if GSCP is feasible, then $\mathcal{V}(\overline{GSCP}) = \mathcal{V}(\overline{DGSCP})$ is a lower bound on $\mathcal{V}$(GSCP).

However, optimally solving $\overline{GSCP}$ (or $\overline{DGSCP}$) can be expensive from a computational point of view and a reasonably good lower bound may be obtained with less effort using heuristics to compute feasible solutions for $\overline{DGSCP}$.

Such heuristics can be developed for the GSCP following a similar pattern to the dual greedy heuristics developed by Balas and Ho [1] for the set covering problem. Hence the dual greedy heuristic that we developed for the GSCP selects, in each iteration, a particular element, say index i*, from a subset row $R \subseteq M$ and assigns a feasible value to the corresponding dual variable, u_i*. The row index i* is then removed from R and the process repeats until R becomes empty. If the initial R is strictly contained in M, the rows of M-R can be considered in order to improve the lower bound.

Two possibilities were considered for initializing R:

$$R := M \tag{DUAL 0}$$

and

$$R := \{i \in M : \sum_{j \in N_i} \bar{x} = b_i, \qquad \text{with } \bar{x} \in \mathcal{F}\text{(GSCP)}\}. \tag{DUAL 1}$$

The version DUAL 1 explicitly requires a feasible solution for GSCP which can be obtained from the procedure presented in the previous section.

For selecting a row from R, several different criteria were considered in [8] and [9]. Concerning to the dual variables v_j ($j \in N$), although they can be given positive values, our own experience has shown that they seldom differ from zero in the best heuristic solutions for the problems such that the variable-bounds h_j are not very tight. In fact, this is the case for the real life bus crew scheduling problems that we considered and, therefore, one forces the v_j to be always at zero level.

Both DUAL 0 and DUAL 1, are synthetized in the following:

<u>**Procedure** DUAL</u>

input: M, N, $(c_j, h_j, M_j)_{j \in N}$, $(b_i, N_i)_{i \in M}$,
$(\bar{x}_j)_{j \in N}$ (DUAL 1) [feasible solution for GSCP]

output: $((u_i)_{i \in M}, \mathbf{0})$, z [feasible solution for $\overline{DGSCP}$ and lower bound on $\mathcal{V}$(GSCP)]

1. [Initializing]

$u_i := 0 \quad (i \in M)$ [dual variable associated to row i]

$$R := \begin{cases} M & \text{(DUAL 0)} \\ \{i \in M : \sum_{j \in N_i} \bar{x}_j = b_i ,\ \bar{x} \in \mathcal{F}(GSCP)\} & \text{(DUAL 1)} \end{cases}$$

$z_\ell := 0$

2. [Selecting a row]

if $R = \Phi$ **then** **goto 4**
 else choose $i^* \in R$
 endif

3. [Assigning a value to the dual variable and updating]

$u_{i^*} := \min_{j \in N_{i^*}} c_j$

$z_\ell := z_\ell + u_{i^*} b_{i^*}$

$$c_j := \begin{cases} c_j - u_{i^*} & (j \in N_{i^*}) \\ c_j & (j \in N - N_{i^*}) \end{cases}$$ [update j-th reduced cost]

$R := R - \{i^*\}$

goto 2

4. [Checking the terminal conditions]
 if (step 4 is entered for the first time) **then** ($R := M - R$ **and goto** 2) **endif**
 stop

In a rather straightforward implementation this procedure DUAL is $O(|M| \max\{|M|, |N|\})$.

An attempt to improve the bounds, both from below and from above, produced by the heuristics PRIMAL and DUAL, can be made by trying to impose the linear complementary conditions on the corresponding solutions. That is, if $[\bar{x}, (\bar{u},0)]$ is a pair of primal and dual feasible solutions for $\overline{GSCP}$ one aims to impose:

$$\sum_{i \in M} \bar{u}_i \left(\sum_{j \in N_i} \bar{x}_j - b_i \right) = 0 \quad (3.4)$$

$$\sum_{j \in N} \bar{x}_j \left(\sum_{i \in M_j} \bar{u}_i - c_j \right) = 0 . \quad (3.5)$$

Obviously, if $\bar{x}$ and $(\bar{u},0)$ satisfy (3.4) and (3.5) then they are optimal solutions, respectively for $\overline{GSCP}$ and $\overline{DGSCP}$. If $\bar{x}$ is feasible for GSCP then $v(GSCP) = \sum_{j \in N} c_j \bar{x}_j$.

The procedure IMPROVE 1, which is briefly described next, performs linear complementary improvement tests for a pair of dual and primal solutions.

Firstly, the primal solution is modified in order to satisfy (3.5) by setting $\bar{x}_j = 0$ if $r_j > 0$. The resulting solution is unlike to be feasible and a primal greedy heuristic technique is used to bring it into feasibility. Then, the row complementary relations (3.4) are imposed to the dual solution relatively to the new primal solution in the following way:

- set, all the dual variables associated with positive surplus equal to zero;
- keeping dual feasibility, increase first the dual variables associated with zero surplus and then increase the ones which are associated with positive surplus.

The process repeats itself until no changes in both solutions are produced or the number of iterations exceeds a maximum value which, in our case, was fixed at 3.

The procedure IMPROVE 1 has the same complexity that the primal heuristic procedure called as a subroutine.

Another bound improving technique, that we named IMPROVE 2, was deviced as follows:

1. initialize $\bar{N}$ as the index set of columns for a primal feasible solution;

2. compute both primal and dual feasible solutions for the problem restricted to $\bar{N}$;

3. find, among N-$\bar{N}$, the variable with the most negative reduced cost for the restricted dual solution; if no negative reduced cost variable exists, stop;

   otherwise, include the selected variable in $\bar{N}$ and resume to 2.

The algorithm IMPROVE 2 has a computational complexity equal to |N| times the complexity of the primal-dual procedure called in step 2.

4. COMPUTATIONAL RESULTS

As we have seen in the previous sections of this paper the primal and dual heuristics are polinomial algorithms on the dimensions of the GSCP, although, for some of the heuristics, the number of computing steps is bounded above by a polinomial in |M| |N| with high degree.

Besides, this worst-case computational study, an analysis of the heuristics from an empirical point of view was carried out. The test problems that we considered are of the bus crew scheduling type and so, exhibit the particularities mentioned in section 1.

All the test problems have the same number of covering constraints, |M| =36, corresponding to the working periods from 6 a.m. to midnight. The set of test problems, consists of 30 instances generated according to the driver's contract rules in Rodoviária Nacional (15 with 100 columns - R1 to R15 - and 15 with 865 - G1 to G15). The right-hand-side numbers were randomly generated in order to produce 3 different type of distributions - unimodal, bimodal and irregular - for the demand pattern. The last 4 test problems correspond to real bus crew scheduling situations at Rodoviária Nacional (RN1 to RN4 also with 865 columns). The cost of each working shift is defined according to the rules in use at Rodoviária Nacional and consists of a fixed cost added by extra-costs related either to the period of the day or overtime working periods.

For a more detailed description of these problems see [7].

The combined procedure includes all heuristics described in this paper and gives a feasible solution for the GSCP and simultaneously a lower bound that yields a measure by excess of the error associated to the solution.

After performing extensive tests, we opted by a final version consisting of a combination of procedure DUAL 0, DUAL 1 (both with two different selection criteria), GREEDY PRIMAL (several row and column selection criteria) with SEARCH, IMPROVE 1 and IMPROVE 2. Also dual linear penalties ([8]) were calculated in order to tighten the variable-bounds, hj, or eventually eliminating the variables.

This experience is reported in Table I where the column (1) identifies the problem which dimensions (|M|, |N| and density - number of ones over |M| |N|) and optimal value are shown, respectively, in columns (2) and (3).

TABLE I
Computing times and quality of the bounds obtained with the primal-dual heuristic procedure

Problem (1)	Dimension (2)	Optimal value (3)	Lower % (4)	Upper % (5)	Time sec (6)
R1	|M| = 36	51726	90.5	3.2	6.0
R2	|N| = 100	52268	96.3	0.7	6.3
R3	55%	84184	99.7	2.2	7.1
R4		85080	93.8	4.4	9.7
R5		120930	99.5	0.1	6.8
R6		48990	100.0	0.0	2.3
R7		27062	98.4	1.3	4.8
R8		55236	96.6	2.9	6.4
R9		47470	98.9	1.0	5.8
R10		26612	87.5	1.1	4.2
R11		98930	95.8	1.3	5.5
R12		51248	89.3	0.0	4.3
R13		49096	93.2	0.0	6.1
R14		40862	97.1	4.2	4.8
R15		188598	94.4	3.6	6.6
G1	|M| = 36	49972	92.2	5.2	31.6
G2	|N| = 865	49326	96.3	0.8	28.6
G3	56%	80248	99.0	1.0	32.2
G4		79670	92.1	3.0	39.1
G5		115540	100.0	0.4	36.2
G6		48434	100.0	0.0	16.8
G7		24084	91.2	2.6	21.1
G8		51900	96.3	1.2	30.1
G9		45658	99.8	0.5	21.9
G10		25248	91.3	3.0	26.4
G11		93078	99.9	0.1	22.1
G12		51134	89.2	1.0	28.9
G13		47720	89.6	0.0	17.2
G14		36520	100.0	0.0	9.2
G15		167728	99.99	0.1	23.2
RN1	|M| = 36	70418	96.0	0.1	40.4
RN2	|N| = 865	45952	98.3	4.4	36.8
RN3	56%	33360	83.4	0.5	31.2
RN4		80316	100.0	0.0	29.1
Average			95.5	1.5	
Worst value			83.4	5.2	
Number of optimal values			5	7	

The remaining columns of Table I refer to the computing experience relative to the primal-dual heuristic procedure that was carried out on a VAX 11/750 VMS with FPA and FORTRAN compiler (similar to the one presented in [8]).

Columns (4) and (5) refer the quality for the final lower and upper bounds. Hence, in column (4), the value $(z_\ell \, / \, v(GSCP))10^2$ is shown for each test problem and, for evaluating the upper bounds, we give the value $10^2 \, (z_u - v(GSCP)) \, / \, v(GSCP)$ in column (5).

Finally column (6) shows the computing time in seconds.

From this experience we can see (column (5)) that the final upper bound is very good for most of the test problems. In fact, the worst value (problem G1) was 5.2% above the optimum, and the upper bound is equal to the optimal value for 7 out of 34 test problems. For the large real life problems (RN1 to RN4), the upper bound value proved even better. Concerning to the lower bound values, the procedure didn't prove as good as above. Although, the lower bound value is within 1% of the optimum for 11 test problems, the procedure failed to produce reasonably good results for some cases, in particular for problem RN3.

5. CONCLUSIONS

Finally, we can conclude that the primal-dual greedy procedure described in this paper is a relatively easy method to tackle with the generalized set covering problem. In particular, it proved to be reasonably fast at least for the two-duty period bus driver scheduling instances that we tried out to solve. The primal-dual heuristics implemented on a microcomputer have been in use at Rodoviária Nacional ([5]) with very good results for the real bus crew scheduling problems.

Moreover, this procedure can be used as an initial step for a subgradient optimization based technique to improve a lagrangean bound on the optimum. It also can be embedded, as the bounding tool, in a tree-search-method for optimally solving the GSCP.

REFERENCES

[1] Balas, E. and Ho, A., "Set covering Algorithms Using Cutting Planes, Heuristics and Subgradient Optimization: A Computational Study", Mathematical Programming, vol. 12, pp. 37-70 (1980).

[2] Balas, E. and Padberg, M.W., "On the Set Covering Problem", Operations Research, vol. 20, pp. 1153-1161 (1972).

[3] Hall, N. and Hochbaum, D., "The Multicovering Problem: The Use of Heuristics, Cutting Planes and Subgradient Optimization for a Class of Integer Programs", Working Paper WPS 85-73, College of Administrative Science, The Ohio State University (1985).

[4] Paixão, J.P., "Algorithms for Large Scale Set Covering Problems", Ph.D.Thesis, Department of Management Science, Imperial College, London (1983)

[5] Paixão J.P., Branco, I.M., Captivo, E., Pato, M.V., Eusébio, R. and Amado, L., "Bus and Crew Scheduling on a Microcomputer", in "Or Models on Microcomputers", J.D. Coelho and L. V. Tavares (eds), North Holland Amsterdam (1986).

[6] Paixão, J.P. and Pato, M.V., "Análise de Pior-Caso no Comportamento de Heurísticas Greedy para o Problema de Cobertura Generalizado", nota 9/87 CEA, Faculdade de Ciências da Universidade de Lisboa (1987).

[7] Paixão, J.P. and Pato, M.V., "Greedy Heuristics for Generalized Set CoveringProblems: A Computational Study", nota 8/85 CEA, Faculdade de Ciências da Universidade de Lisboa (1985).

[8] Paixão, J.P. and Pato, M.V., "Heuristics, Lagrangean Relaxation and Penalties for Two-Duty Period Scheduling Problems", nota 1/87 CEA, Faculdade de Ciências da Universidade de Lisboa (1987).

[9] Paixão, J.P. and Pato, M.V., "Heuristics for an Operator Scheduling Problem", nota 10/85 CEA, Faculdade de Ciências da Universidade de Lisboa (1985).

[10] Papadimitriou, C.H. and Steiglitz, K., "Combinatorial Optimization", Prentice Hall, Englewood Cliffs (1982).