M. Minoux

A class of combinatorial problems with polynomially solvable large scale set covering/partitioning relaxations

A CLASS OF COMBINATORIAL PROBLEMS WITH POLYNOMIALLY SOLVABLE LARGE SCALE SET COVERING/PARTITIONING RELAXATIONS (*)

by M. Minoux ([1])

———————

Abstract. – *We introduce and study a whole class of combinatorial problems which can be formulated as very large size set covering/set partioning problems whose linear relaxations are either solvable in polynomial time via the Ellipsoid Algorithm, or simply efficiently solvable by using generalized linear programming techniques. An extension to set packing problems is also considered. This class is shown to contain both already well-known combinatorial problems as well as a number of apparently new ones, stated here for the first time. Moreover it is shown that, for all these problems, proving they belong to this class automatically provides new attractive computational approaches. Two recent important practical applications of these methods are mentionned: one concerns an optimization problem arising in satellite communications; the other concerns crew scheduling problems in airline companies.*

Keywords : Combinatorial optimization; set covering, set partitioning; generalized linear programming; column generation.

Résumé. – *On introduit ici toute une classe de problèmes combinatoires qui peuvent se formuler comme des problèmes de recouvrement/partitionnement de très grandes dimensions dont les relaxations continues : ou bien sont solubles en temps polynomial par l'algorithme des ellipsoïdes; ou bien peuvent être résolues de façon efficace par des techniques de programmation linéaire généralisée. On montre que cette classe contient non seulement des problèmes combinatoires bien connus, mais aussi un certain nombre de problèmes apparemment nouveaux et décrits ici pour la première fois. De plus, on montre que pour tous ces problèmes, le fait de montrer l'appartenance à la classe fournit automatiquement de nouvelles méthodes de résolution a priori intéressantes. Deux récentes applications pratiques importantes de ces méthodes sont mentionnées : l'une concerne un problème d'optimisation de système de Télécommunications par satellite; l'autre le problème des rotations d'équipages dans les compagnies aériennes.*

Mots clés : Optimisation combinatoire; problèmes de recouvrement et de partitionnement; programmation linéaire généralisée; génération de colonnes.

———————

1. INTRODUCTION

The purpose of this paper is to exhibit and study a large class of combinatorial optimization problems, all of which enjoy the following nice properties:

(i) they can be reformulated as very large size *set covering* or *set partitioning problems* (with possibly one additional constraint); the "very large size" attribute will be made more precise in the paper. Also, an extension to *set packing problems* will be suggested in section 4.6.2.);

(ii) due to some special structure of the constraint matrix, the *linear programming relaxations* of these large size set covering set/partitioning problems can be solved *exactly* by using *generalized linear programming techniques* (column generation). Moreover, the *polynomial solvability* of these linear programming relaxations can be proved by using a property of the Ellipsoid Algorithm (Shor 1977, Khachian 1979) first exhibited by Grötschel *et al.* (1981).

It will be shown that this class contains a number of well-known problems, or problems already treated by other approaches, such as:

— simple plant location problems;

— partitioning the edges of a graph into forests;

— the so-called graph partitioning problem;

— the chromatic index problem;

— clustering problems arising in such contexts as locating (TV or radio) broadcasting stations, or services (fire, medical etc.).

For all the above-mentioned problems, the general solution method suggested in this paper provides a new alternative computational approach.

However, an additional interesting feature of this work is that it led us to *identify quite a few new problems*, stated here for the first time to our knowledge, some of which might have important potential practical applications; for instance:

— the k-center-sum problem (*see* § 4.1.3);

— minimum weighted partitioning of a matroïd with bases or independent subsets (*see* § 4.4.3);

— minimum weighted partitioning with subsets belonging to the intersection of two matroïds (*see* § 4.5);

— packing the edge set or arc set of a graph into cutsets (weighted and unweighted cases) (*see* § 4.7).

2. FIRST EXAMPLE: THE TV BROADCASTING STATION PROBLEM REVISITED AND NEW EXTENDED MODEL

As an introductory example, we first focus on a very classical textbook-example which can be found almost everywhere as a typical illustration for set covering problems.

Suppose that, on a given geographical area, we have to locate TV broadcasting centers so that the TV signal is appropriatedly received by any of p given cities with known geographical position. Suppose that n possible sites for the broadcasting centers are given, and for each site j ($1 \leq j \leq n$) we know:

— the subset V_j of cities "covered" by a center installed on site j;

— the corresponding cost c_j of installing a center on site j. With each site j, associate a binary $(0-1)$ variable x_j (with the meaning that $x_j = 1$ iff site j is actually selected in the solution) and denote by A the $p \times n$ $0-1$ matrix where, for $j = 1, 2, \ldots, n$, A_j, the j-th column of A, is the characteristic vector of subset V_j. Then the problem of "covering" all the cities at minimum cost can be simply stated as the *set covering problem*:

$$\text{Minimize } \sum_{j=1}^{n} c_j x_j \text{ subject to}$$

$$\text{(TV 1)} \qquad \sum_{j=1}^{n} A_j x_j \geq 1$$

$$x \in \{0, 1\}^n$$

where **1**, at the right-hand side of the set covering constraints, is the p-vector with all components equal to 1. (Observe here that it doesn't matter if some city is covered more than once in the solution, so the relevant model is indeed set covering rather than set partitioning).

Typical values of p and n for practical problems ranging from a few tens to about 100-200, the above can be coped with via existing integer programming techniques such as:

— cutting planes: *see* e. g. Delorme 1974;

— branch-and-bound techniques: *see* e. g. Lemke, salkin and Spielberg 1971;

— more involved techniques combining cutting-planes, branch-and-bound, and subgradient optimization as suggested in Balas and Ho (1980).

However, if we take a more realistic point of view, it could be objected that the applicability of currently available integer programming tools, as

those mentioned above, only results from the fact that we restricted indeed to a very simplified (if not simplistic) model. As a matter-of-fact, in real situations, broadcasting stations featuring various possible power levels (not only a single power level) can be installed on any site j, resulting in a whole family of (cost, covered subset) pairs say: (c_{j1}, v_{j1}); (c_{j2}, v_{j2}), . . . , (c_{jk_j}, v_{jk_j}). Observe that it is natural to assume that the cost is a monotone nondecreasing set function, i. e. it increases in the weak sense when extra elements are included in the subset.

We thus introduce and discuss here a more general and realistic model, in which it will be assumed that each broadcasting center on any site j can be assigned a continuum of power levels ranging from 0 to some maximum value P_{max} (the value which would be necessary to reach all the p given cities from j).

For any fixed j, define Γ_{ij} as the minimum cost of a broadcasting station located at j capable of reaching city i (i. e. the cost corresponding to the assignment of the minimum power level required at j to reach i). Note that each Γ_{ij} value can take account of the possible perturbations in the propagation of the radio waves between the specific pair (i, j) of locations, due to geographical accidents (mountains etc.), so that the model is able to fit very accurately any given practical situation.

With the above data at hand, the exact minimum cost of a broadcasting station located at j and covering all the cities contained in any subset $V \subset \{1, 2, \ldots, p\}$ is readily obtained as:

$$\gamma_j(V) = \underset{i \in V}{\text{Max}} \{\Gamma_{ij}\}. \tag{1}$$

Observe here that, for any fixed j, at most p distinct subsets (let us call them *active subsets*) have to be considered. This is because, if i_0 is the element in V for which the maximum in (1) is obtained, then the subset $V'_{i_0 j} = \{i / \Gamma_{ij} \leqq \Gamma_{i_0 j}\}$ has same cost as V but (usually) contains more elements. Since we do not care about covering some city more than once, it is clear that optimality is not lost in the set covering model by restricting to the active subsets such as V'.

Note that there are at most p of these, each one corresponding to an element i_0 in $\{1, 2, \ldots, p\}$.

As a whole we get a new set covering model, call it (TV 2), with p constraints but $q = n \times p$ columns (p distinct subsets for each possible site j):

(TV 2)

$$\text{Minimize } \sum_{i=1}^{p} \sum_{j=1}^{n} \Gamma_{ij} \, x_{ij} \text{ subject to:}$$

$$\sum_{i=1}^{P} \sum_{j=1}^{n} A_{ij} \, x_{ij} \geqq 1$$

$$x_{ij} = 0 \quad \text{or} \quad 1 \quad (i = 1, \ldots, p; j = 1, \ldots, n),$$

where, for all $j = 1, \ldots, n$, for all $i = 1, \ldots, p$ the double subscript (i, j) relates to the active subset:

$$V' = \{l / \Gamma_{lj} \leqq \Gamma_{ij}\}$$

with cost Γ_{ij}, and incidence vector A_{ij}

Example: We illustrate the above model by means of the following small example with 5 cities and 5 possible locations (in the cities themselves). The inter-city distance matrix (in kilometers) is:

	1	2	3	4	5
1	0	30	110	50	20
2	30	0	80	70	30
3	110	80	0	90	60
4	50	70	90	0	40
5	20	30	60	40	0

We assume that the cost $c(d)$ of installing a broadcasting station at any place with enough transmission power to reach any other city within distance d has a simple analytical form such as:

$$\begin{cases} c(d) = 0.1\, d & \text{for } d \geqq 20 \\ c(d) = 2 & \text{for } d \leqq 20 \end{cases}$$

(d expressed in kilometers).

Thus, for a station placed in city 1, the minimum cost required to reach city 4 ($d = 50$) is $0.1 \times 50 = 5 = \Gamma_{4,1}$. The corresponding subset of cities covered is in this case $\{1, 2, 4, 5\}$. Similarly, covering only cities 4 and 5 by means of a station located in city 4 would cost: $0.1 \times 40 = 4 = \Gamma_{5,4}$.

The resulting set covering problem (TV 2) would be in this case

	Γ_{11}	Γ_{21}	Γ_{31}	Γ_{41}	Γ_{51}	Γ_{22}	Γ_{32}	Γ_{42}	Γ_{52}	Γ_{33}	Γ_{43}	Γ_{53}	Γ_{44}	Γ_{54}	Γ_{55}	
	2	3	11	5	2	2	8	7	3	2	9	6	2	4	2	
1	1	1	1	1	1	0	1	1	1	0	0	0	0	0	0	≥ 1
2	0	1	1	1	0	1	1	1	1	0	1	0	0	0	0	≥ 1
3	0	0	1	0	0	0	1	0	0	1	1	1	0	0	0	≥ 1
4	0	0	1	1	0	0	1	1	0	0	1	0	1	1	0	≥ 1
5	0	1	1	1	1	0	1	1	1	0	1	1	0	1	1	≥ 1

(end of example).

If p and n are large, say $p = 500$ and $n = 200$ we are then confronted with a large size set covering problem, unsolvable by the current integer programming techniques (10^5 binary variables) However, what we now intend to show is that, in spite of its large size, (TV 2) has some structure which can be exploited from a computational point of view.

Consider the linear relaxation of (TV 2) obtained by dropping the integrality requirements on the variables :

$$\text{Minimize} \sum_{i=1}^{p} \sum_{j=1}^{n} \Gamma_{ij} x_{ij} \text{ subject to:}$$

$(\overline{\text{TV 2}})$

$$\sum_{i=1}^{p} \sum_{j=1}^{n} A_{ij} x_{ij} \geq 1$$

$$0 \leq x_{ij}, \qquad i = 1, \ldots, p; \quad j = 1, \ldots, n.$$

Even though $(\overline{\text{TV 2}})$ is recognized as an ordinary linear program, the actual possibility of solving it by the simplex method for large n and p is not obvious (just observe that for $p = 500$ and $n = 200$, the average number of nonzero elements in the constraint matrix would be $> 10^7$, so just storing the constraint matrix would be out of range of most available computers).

However, the important observation is that $(\overline{\text{TV 2}})$ can be efficiently solved by *generalized linear programming* (column generation or constraint generation) techniques (*see* Dantzig 1963). This is because there is, in this case, an efficient (polynomial time) *column generator algorithm* (*see* e. g. Minoux 1983) which works as follows:

at any stage of the column generation process, a partial set-covering problem, restricted to only a small number of columns out of $(\overline{\text{TV 2}})$, is solved to optimality via the simplex algorithm. Let $\pi_1, \pi_2, \ldots \pi_p$ the optimal dual variables (simplex multipliers) obtained. The next step is to find the minimum reduced cost column. This is easily done for fixed j because, for

any active subset V' corresponding to city i ($V' = \{l/\Gamma_{lj} \leq \Gamma_{ij}\}$) the reduced cost of the corresponding column is:

$$\bar{\Gamma}_{ij} = \Gamma_{ij} - \sum_{i \in V'} \pi_i.$$

Note here that the computation of the p values $\bar{\Gamma}_{ij}$ can be carried out in time complexity $\mathcal{O}(p)$ since, sorting the Γ_{ij} values according to nondecreasing values in such a way that:

$$\Gamma_{i_1 j} \leq \Gamma_{i_2 j} \leq \cdots \leq \Gamma_{i_p j}$$

the p active subsets corresponding to j are:

$$\{i_1\}, \{i_1, i_2\}, \{i_1, i_2, i_3\}, \ldots, \{i_1, i_2, \ldots, i_p\}$$

The minimum reduced cost column

$$\bar{\Gamma}_{i_0 j_0} = \min_{\substack{i = 1, \ldots, p \\ j = 1, \ldots, n}} \{\bar{\Gamma}_{ij}\}$$

in thus obtained in time complexity $\mathcal{O}(np)$.

If $\bar{\Gamma}_{i_0 j_0} \geq 0$ then global optimality has been reached and the whole column generation procedure terminates. Otherwise, the new column is added to the current restricted set covering problem and the process is repeated with this new augmented restricted problem.

The (generalized) TV broadcasting location problem is thus an example of a combinatorial problem for which it is possible to define a large scale set covering linear relaxation amenable to efficient solution through generalized linear programming.

Indeed, it will be shown is section 3 that, as a direct consequence of the existence of a *polynomial generator algorithm*, the polynomial solvability of the large scale set covering linear relaxation is obtained.

Finally section 4 will be devoted to show that there is a large number of interesting combinatorial problems which share these properties.

3. THE GENERAL SET COVERING/SET PARTITIONING MODELS AND COMPLEXITY OF THEIR LINEAR RELAXATIONS

3.1. Definitions and notations

Let $E = \{1, 2, \ldots, p\}$ be a given set of p elements (ground set) and let $\mathcal{F} = \{S_1, S_2, \ldots, S_q\}$ be a finite family of q subsets of E.

We will consider later on a huge variety of specializations for the pair $(E, \mathscr{F})$. For instance, we may think of $\mathscr{F}$ as $\mathscr{P}(E)$, the power set of E. Or, else, we may think of E as being the edge set of some given unoriented graph G, $\mathscr{F}$ being the set of all subtrees (connected cycle-free partial subgraphs) in G. In what follows, we will be especially interested in the cases where $\mathscr{F}$ is allowed to contain a very large number q of elements (e. g. growing exponentially in p, the cardinality of the ground set), while requiring that $\mathscr{F}$ is a whole family of *well-identified combinatorial objects* (e. g. all the subsets of E; all the subtrees of a graph; all the paths between two vertices in a graph, etc.). The TV broadcasting location problem presented in section 2 was a first example of this.

The *incidence matrix* of $\mathscr{F}$ in E is defined, in the classical way, as the $p \times q$ 0-1 matrix

$$A = (a_{ij})_{\substack{i=1, \ldots, p, \\ j=1, \ldots, q}}$$

where $a_{ij} = 1$ if and only if S_j, the j-th member of the family $\mathscr{F}$, contains element i of E.

Suppose now that, to each element S_j of $\mathscr{F}$, we attach a real number $c_j = \gamma(S_j)$ called the *cost* of member S_j in $\mathscr{F}$. Let us associate with each S_j in $\mathscr{F}$ a binary (0-1) variable x_j, and consider the following 0-1 integer programming problem (set covering):

$$SC[E, \mathscr{F}, \gamma] \qquad \begin{array}{c} \text{Minimize } \sum_{j=1}^{q} c_j x_j \text{ subject to:} \\[2mm] \sum_{j=1}^{q} A_j x_j \geq 1 \\[2mm] \forall j, \quad x_j = 0 \quad \text{or} \quad 1, \quad j = 1, \ldots, q. \end{array} \qquad (2)$$

where, for all $j = 1, 2, \ldots, q$, A_j denotes the j-th column of A (characteristic vector of S_j in E) and $\mathbf{1}$ the p-component vector of all 1's.

This problem is denoted by $SC[E, \mathscr{F}, \gamma]$ to indicate that it is a set covering problem perfectly determined by specifying the family $\mathscr{F}$ on E, and the cost set function γ.

The linear relaxation of the above set covering problem is denoted by $\overline{SC}[E, \mathscr{F}, \gamma]$ and is clearly obtained by replacing the integrality requirements on the variables ($x_j = 0$ or 1) by

$$0 \leqq x_j, \qquad j = 1, \ldots, q.$$

In a natural way, when the inequality constraints (2) are replaced by equality constraints, we obtain *the set partitioning problem:*

$$\text{Minimize } \sum_{j=1}^{q} c_j x_j \text{ subject to}$$

$$SP[E, \mathscr{F}, \gamma] \qquad \sum_{j=1}^{q} A_j x_j = \mathbf{1} \qquad (3)$$

$$x_j = 0 \quad \text{or} \quad 1, \qquad j = 1, \ldots, q.$$

The continuous relaxation of the above problem is denoted: $\overline{SP}[E, \mathscr{F}, \gamma]$.

As will be seen, the above general formulations will already allow us to model many combinatorial problems. However the class of problems can still be significantly enlarged by considering a slightly more general model in which the x vector is restricted to belong to some polytope $X \subset [0, 1]^q$ represented by linear equality or inequality constraints.

In most cases of practical interest (*see* § 4) these constraints will assume a very simple form such as:

$$\sum_{j=1}^{q} x_j = k \qquad (4)$$

(k is a given positive integer).

It will then be convenient to use the notation $SC[E, \mathscr{F}, \gamma, X]$ for the problem:

$$\text{Minimize } \sum_{j=1}^{q} c_j x_j \text{ subject to}$$

$$SC[E, \mathscr{F}, \gamma, X] \qquad \sum_{j=1}^{q} A_j x_j \geqq \mathbf{1}$$

$$x \in X \subset [0, 1]^q$$

$$x_j = 0 \quad \text{or} \quad 1, \qquad j = 1, \ldots, q.$$

and $SP[E, \mathscr{F}, \gamma, X]$ for the corresponding set partitioning problem. Of course, the linear relaxations of these problems will be respectively denoted by $\overline{SC}[E, \mathscr{F}, \gamma, X]$ and $\overline{SP}[E, \mathscr{F}, \gamma, X]$.

In a natural way, problems $SC[E, \mathscr{F}, \gamma]$ or $SP[E, \mathscr{F}, \gamma]$ in which no extra constraint is imposed, will often be identified with special instances of $SC[E, \mathscr{F}, \gamma, X]$ or $SP[E, \mathscr{F}, \gamma, X]$ in which $X = H = [0, 1]^q$, the whole unit hypercube in q-dimensional space.

We note here that it would also be possible to define *set packing problems* associated with specified triples $[E, \mathscr{F}, \gamma]$: such an extension will be considered in section 4.6 in connection with path-constrained network flows, and in section 4.7 in connection with cutset packing problems in graphs.

3.2. Efficient solvability and the polynomial column generation assumption

Now, what we are basically interested in, here, are combinatorial problems which, when stated into the form of $SC[E, \mathscr{F}, \gamma, X]$ or $SP[E, \mathscr{F}, \gamma, X]$ for some specific choice of E, $\mathscr{F}$, γ and X, *result into efficiently solvable linear relaxations* $\overline{SC}[E, \mathscr{F}, \gamma, X]$ or $\overline{SP}[E, \mathscr{F}, \gamma, X]$.

Of course, the motivation behind the requirement of efficient solvability of the linear set covering/set partitioning relaxations is to eventually use them to provide *lower bounds* which, if not computationally too expensive to obtain, can subsequently be exploited within branch-and-bound procedures *to get exact optimal integer solutions* to the problems under consideration.

Since we have in mind possibly large values of p (the size of the ground set, a mesure of the size of the original problem) and we want to allow for families $\mathscr{F}$ having cardinalities growing exponentially with p, the only known pratical ways ([2]) of solving linear programs such as $\overline{SC}[E, \mathscr{F}, \gamma, X]$ (or $\overline{SP}[E, \mathscr{F}, \gamma, X]$) rely on *generalized linear programming* (column generation) techniques.

Moreover, it is clear that the overall generalized linear programming procedure cannot be efficient if the column generation subproblems cannot be solved efficiently at each generation step.

In view of this, we will focus here on problems such that, once formulated in terms of $SC[E, \mathscr{F}, \gamma, X]$ or $SP[E, \mathscr{F}, \gamma, X]$, the following *polynomial*

([2]) Apart from very special cases such as *submodular cost functions* for which a specific efficient algorithm exists (the greedy algorithm). See § 4.1.6 below.

generation assumption (PGA) holds:

Given any set of real numbers π_1, π_2, ... π_p associated with the elements of the ground set E, there exists a polynomial algorithm (column generator) for finding a member S^* of $\mathscr{F}$ such that:

(PGA)
$$\bar{\gamma}(S^*) = \gamma(S^*) - \sum_{i \in S^*} \pi_i = \underset{S \in \mathscr{F}}{\text{Min}} \{ \gamma(S) - \sum_{i \in S} \pi_i \} \qquad (5)$$

($\bar{\gamma}(S^*)$ above is the reduced cost of the minimum reduced cost column).

(Here *polynomial* means *polynomial* in p, the cardinality of the ground set).

We show in section 4 that (PGA) indeed holds for a huge variety of interesting combinatorial problems, when reformulated in the above framework. For all these problems, of course, generalized linear programming will provide an efficient practical way:

— of getting good lower bounds (see the few computational results mentioned in section 4 for some problems to which the approach was already applied) leading to search trees of reduced dimensions;

— of solving sometimes the integer problem itself (either because the continuous solution obtained happens to be integral, or because the family $\mathscr{F}$ and the cost function γ have properties guaranteeing the integrality of the solution).

However, we first end up this section by stating a nice result showing that the class of combinatorial problems for which (PGA) holds is interesting, not only from a practical and computational point of view, but also from a theoretical point of view.

3.3. Polynomial solvability of the linear relaxations under the (PGA) assumption

For convenience, we restrict here to consider the set covering case, since an identical proof would follow, step by step, for the set partitioning case.

Denoting by $\pi = (\pi_1, \pi_2, \ldots, \pi_p)$ the vector of dual variables, the linear program dual to $\overline{SC}[E, \mathscr{F}, \gamma]$ reads:

$$\overline{DSC}[E, \mathscr{F}, \gamma] \qquad \begin{array}{c} \text{Maximize} \sum_{i=1}^{p} \pi_i \text{ subject to:} \\[1em] \sum_{i \in S_j} \pi_i \leqq \gamma(S_j) \qquad \text{(for all } S_j \text{ in } \mathscr{F}) \\[1em] \pi_i \geqq 0, \qquad i = 1, \ldots, p. \end{array}$$

Though, in cases of interest to us, this problem has a very large number of constraints, it can be solved efficiently by a generalized linear programming simplex algorithm: this leads to a *constraint generation scheme* which is just the dual to a column generation scheme.

However, another alternate way of solving a structured linear program with many constraints such as $\overline{DSC}[E, \mathscr{F}, \gamma]$ is to apply the *ellipsoïd algorithm* (Shor 1977, Khachian 1979). In this approach, a sequence of ellipsoïds $\mathscr{E}^t$ with geometrically decreasing volumes is generated, each of which contains at least one optimal point.

Moreover, the reduction factor in the volume between $\mathscr{E}^t$ and $\mathscr{E}^{t+1}$ is, in the worst case, equal to a constant (< 1) depending only on the dimension p of the space (the number of variables of the problem) and thus it can be shown that the number of necessary iterations to get an ellipsoïd containing *at most one extreme point* of the polytope Q of feasible points is bounded by a polynomial in p (the dimension of the space) and $\log_2 T$ where T is the *rationality constant* of the polytope Q. More specifily, assuming that the extreme points of Q have rational coordinates, T is the largest absolute value of any integer appearing as a numerator or a denominator in the coordinates of any extreme point of Q.

Since all the coefficients in the constraints of $\overline{DSC}[E, \mathscr{F}, \gamma]$ are 0 or 1, it can easily be shown that the largest possible determinant of a basis is bounded by p^p, hence that (provided that the γ values are integers, which is not a restrictive assumption here) in this case, $\log_2 T = p \log_2 p$ which is indeed a polynomial in p. So, applied to $\overline{DSC}[E, \mathscr{F}, \gamma]$, the ellipsoïd algorithm will require a number of iterations polynomial in p.

Thus we can apply to our problem the argument, first suggested by Grötschel, Lovász and Schrijver (1981) in a somewhat different context ([3]) namely that the polynomial solvability of $\overline{DSC}[E, \mathscr{F}, \gamma]$ only depends on the polynomial solvability of the so-called *separation problem* (SEP) stated below:

(SEP)
> Given any $\bar{\pi} = \{\bar{\pi}_1, \ldots \bar{\pi}_p\}$, find whether $\bar{\pi} \in Q$ or else produce $S_j \in \mathscr{F}$ such that the corresponding constraint is violated, i. e.:

$$\sum_{i \in S_j} \bar{\pi}_i > \gamma(S_j)$$

([3]) The class of problems considered in Grötschel, Lovász and Schrijver paper were essentially polynomially solvable integer problems, whereas we are concerned here with a new class of problems containing mainly *NP*-complete problems.

Clearly, under (PGA) the above separation problem is polynomially solvable (*see* § 3.2) hence $\overline{DSC}[E, \mathscr{F}, \gamma]$ *is polynomially solvable.*

Also it is easily checked that exactly the same kind of arguments apply to $\overline{SC}[E, \mathscr{F}, \gamma, X]$ and $\overline{SP}[E, \mathscr{F}, \gamma, X]$ when the set $X \subset [0, 1]^q$ is defined by a single constraint such as (4) (*see* § 3.1). We have thus obtained:

PROPERTY 1: Under (PGA) the linear relaxations of $SC[E, \mathscr{F}, \gamma]$ and $SP[E, \mathscr{F}, \gamma]$ are polynomial-time solvable. This extends to $SC[E, \mathscr{F}, \gamma, X]$ and $SP[E, \mathscr{F}, \gamma, X]$ e. g. when $X \subset [0, 1]^q$ is defined by a single linear constraint such as (4).

4. A LIST OF A FEW TYPICAL EXAMPLES

We now turn to describe a number of typical problems belonging to the new class of combinatorial optimization problems studied in sections 2-3 above. Unless otherwise specified, each problem listed below meets the (PGA) assumption.

In each case, a proper statement of the column generation subproblem is provided, together with the complexity of solution algorithms usually expressed in terms of the cardinality of the ground set E.

For clarity, we found it convenient to organize them into a few main subclasses, each subclass corresponding to a specific choice of the pair $(E, \mathscr{F})$.

4.1. Cases where $\mathscr{F}$ is the power set of some ground set E

Consider $E = \{1, 2, \ldots, p\}$ and $\mathscr{F} = \mathscr{P}(E)$.

We first assume here that, for any unordered pair (i, j) of elements in E, a real number d_{ij} is given (and, for all i, $d_{ii} = 0$). d_{ij} may be interpreted as a "distance" or "dissimilarity index" or "cost" between i and j.

4.1.1. *The extended broadcasting center problem*

Consider the case where the cost $\gamma(S)$ of any S in $\mathscr{F}$ (S is thus a subset of E) is given by

$$\gamma(S) = = \underset{\iota \in S}{Min} \ \underset{j \in S}{Max} \{d_{ij}\}. \tag{6}$$

In the case of a set-covering problem (SC) with no extra constraint (i. e. $X = H = [0, 1]^p$) we find again the generalized broadcasting center problem introduced in section 2.

Associate with each element $i \in E$ a binary (0-1) variable y_i ($y_i = 1$ if element i belongs to the optimal — minimum reduced cost — subset looked for). The column generation subproblem can be stated as minimizing:

$$g(y) = \underset{i \in E}{\text{Min}} \ \underset{j \in E}{\text{Max}} \ d_{ij} y_i y_j - \sum_{j \in E} \pi_j y_j \qquad (7)$$

over all $y \in \{0, 1\}^p$.

This problem is polynomially solvable as explained in section 2.

Suppose that the index i_0 achieving the minimum in the first (Min-Max) term in (7) for the optimum solution is known. Then the problem reduces to minimize:

$$\underset{j \in E}{\text{Max}} \ d_{i_0 j} y_j - \sum_{j \in E} \pi_j y_j$$

which can be done in time $\mathcal{O}(p)$ (exactly as explained in section 2) by sorting the $d_{i_0 j}$ values according to nondecreasing order

$$d_{i_0 j_1} \leqq d_{i_0 j_2} \leqq \ldots \leqq d_{i_0 j_p}$$

and computing the minimum reduced cost among the p active subsets

$$\{j_1\}, \{j_1, j_2\}, \ldots, \{j_1, j_2, \ldots, j_p\}.$$

Hence the minimization of (7) can be carried out in time complexity $\mathcal{O}(p^2)$ (we do not take into account here the sorting operation which can be done once for all in a "preprocessing" step before starting the column generation process).

4.1.2. Extended broadcasting center problem with cardinality constraint

When in problem 4.1.1, an extra constraint of the form (4) is imposed $\left(\sum_{j=1}^{q} x_j = k \right)$ then we find an extension of the generalized broadcasting center problem where the number k of centers is imposed. The column generation subproblem remains essentially unchanged in this case, and the resulting complexity for column generation is still $\mathcal{O}(p^2)$.

The same remark would apply to the case of extra constraints of the form $\sum_{j=1}^{q} x_j \geqq k$ or $\leqq k$.

4.1.3. A new location problem: center-sum and k-center-sum problem

E, $\mathcal{F}$ and γ being as in section 4.1.1, consider now the case of set partitioning (instead of set covering). The problem is then recognized as partitioning the ground set E into subsets (classes) $S_1, S_2, \ldots, S_h$, the cost of any subset S in the partition being given by (6), in such a way as to minimize total cost. In location theory (see e. g. Hansen et al. 1985) the element i_0 achieving the minimum in (6) is called the center of the class (or subset) S, and $\gamma(S)$ which is the maximum "distance" from i_0 to any other element of S is the radius of S. We thus want to determine a partition minimizing the sum of the radii. Since the only difference between this problem and the ones in sections 4.1.1 and 4.1.2 is that set covering is replaced by set partitioning, the column generation subproblem is essentially unchanged, hence solvable in the same way in time $\mathcal{O}(p^2)$. In accordance with the common terminology in use in Location Theory, P. Hansen (1986) suggested the term of center-sum problem.

The cardinality constrained version [in which an extra condition of type (4) is imposed] is referred to as the k-center-sum problem. For this problem, which doesn't seem to have been studied before, the present paper provides a readily available computational approach for getting good lower bounds. Computational results will be presented in a forthcoming paper.

4.1.4. Median and K-median problems

E and $\mathcal{F}$ being as above, we now define the cost of any subset $S \in \mathcal{F}$ by:

$$\gamma(S) = \underset{i \in S}{\text{Min}} \left\{ \sum_{j \in S} d_{ij} \right\}. \tag{8}$$

In location theory, the element i_0 achieving the minimum in (8) is called the median element of the class (subset) S.

In case of set partitioning with cardinality constraint of type (4), the problem is thus recognized as the well-known K-median problem (Hakimi 1964, Maranzana 1964).

Given simplex multipliers $\pi_1, \pi_2, \ldots, \pi_p$ associated with the elements of E, the column generation subproblem consists in minimizing, for each i in E:

$$\sum_{i \in E} d_{ij} y_j - \sum_{j \in E} \pi_j y_j = \sum_{j \in E} (d_{ij} - \pi_j) y_j \tag{9}$$

and then taking the minimum of the resulting quantities over all i in E. Clearly, for each given i in E, the minimum in (9) is readily obtained by:

$$d_{ij} - \pi_j > 0 \quad \Rightarrow \quad y_j = 0$$

$$d_{ij} - \pi_j \leqq 0 \quad \Rightarrow \quad y_j = 1$$

and the required computational effort is $\mathcal{O}(p)$.

As a whole, the column generation process has complexity $\mathcal{O}(p^2)$.

4.1.5. The simple plant location problem or uncapacitated facility location

This a slight variation of the previous problem in which there is a fixed cost f_i to be paid whenever element i is chosen as the median element in the subset S.

Hence, in this case, the cost $\gamma(S)$ of any subset S should be replaced by:

$$\gamma(S) = \underset{i \in S}{\text{Min}} \Big\{ f_i + \sum_{j \in S} d_{ij} \Big\}. \tag{10}$$

It is easy to check that the column generation subproblem remains essentially unchanged in this case, therefore being solvable in time $\mathcal{O}(p^2)$. For a bibliography on the simple plant location problem *see* e. g. Hansen *et al.* (1985).

4.1.6. Polymatroïd optimization problems

Again, we consider $\mathscr{F}$ as $\mathscr{P}(E)$ with $E = \{1, 2, \ldots, p\}$ a set of p elements (ground set). In addition, we assume that $\gamma(S)$, the set function giving the cost of any subset S in $\mathscr{F}$ is a *nondecreasing submodular set function* (*see* Edmonds 1970). Consider, first, the case of a set covering problem without extra constraint $SC[E, \mathscr{F}, \gamma]$.

It is easy to chek that the *dual* to $\overline{SC}[E, \mathscr{F}, \gamma]$ is nothing but a standard *polymatroïd optimization problem* (for an introduction to polymatroïd optimization, *see* e. g. Gondran and Minoux 1983 chapter 9). This problem is known to be efficiently (polynomially) solvable by the so-called *greedy algorithm* (Edmonds 1971).

We note that, in this case, our generalized linear programming approach is unlikely to be competitive, since the column generation subproblem would involve the minimization of the submodular set function:

$$\gamma(S) - \sum_{j \in S} \pi_j. \tag{11}$$

Though the above problem has been shown to be polynomially solvable by means of the Ellipsoïd algorithm (Grötschel, Lovász and Schrijver 1981) no efficient purely combinatorial algorithm is known at present for solving the general case of an arbitrary submodular set function. However, it should be observed that there are various interesting special cases of submodular

functions for which efficient combinatorial algorithms exist for solving (11); e. g. pseudoboolean functions with the Rhys-Balinski property (Rhys 1970), submodular quadratic and cubic pseudoboolean functions (Billionnet and Minoux 1985). Consider now the case of a set covering problem with an extra constraint of type (4): $SC [E, \mathscr{F}, \gamma, X]$.

The dual to the relaxed problem $\overline{SC} [E, \mathscr{F}, \gamma, X]$ is no longer a standard polymatroïd optimization problem (due to an extra column with all entries equal to 1 in the constraint matrix). However this "extended" polymatroïd optimization problem is still solvable by the greedy algorithm as shown by Billionnet and Minoux (1984).

For this problem, too, the column generation subproblem is essentially the same as (11).

4.2. Cases where E is the set of nodes of a given unoriented graph and $\mathscr{F}$ is the power set of E

Let $E = \{ 1, 2, \ldots, p \}$ be the vertex set of a p-node unoriented graph G, and $\mathscr{F} = \mathscr{P}(E)$ the family of all possible subsets of nodes.

4.2.1. The graph partitioning problem

For any unordered pair of vertices (i, j) we assume that there is a number d_{ij} associated such that:

— $d_{ij} = 1$ if (i, j) is an edge in G;

— $d_{ij} = 0$ if (i, j) is not an edge in G.

Associate with any subset of nodes $S \in \mathscr{F}$ a cost equal to:

$$\gamma(S) = - \sum_{i \in S} \sum_{j \in S, \, j > i} d_{ij} \qquad (12)$$

i.e., up to the "−" sign, $\gamma(S)$ is the number of edges of G having both endpoints in S. X being defined by a cardinality constraint of type (4) the set partitioning problem $SP [E, \mathscr{F}, \gamma, X]$ is then recognized as the problem of finding an optimal partition of the nodes of G into k subsets such that the sum of all the edges *within the subsets be maximized*. Accordingly, it can be viewed as minimizing the number of edges joining distinct subsets in the partition: this is the so-called *Graph partitioning problem*.

The column generation subproblem in this case can be stated as minimizing the quadratic pseudoboolean function:

$$g(y) = - \sum_{i \in E} \sum_{j \in E, \, j > i} d_{ij} y_i y_j - \sum_{j \in E} \pi_j y_j \qquad (13)$$

over all $y \in \{0, 1\}^p$.

Since all d_{ij} are nonnegative, $g(y)$ given by (13) is known to be *submodular* (*see* e. g. Fischer, Nemhauser and Wolsey 1978) and we are in the case where minimizing (13) is equivalent to solving a max-flow-min-cut problem (*see* e. g. Billionnet and Minoux 1985) on a graph with $p+2$ nodes.

This can be carried out in time $\mathcal{O}(p^3)$ e. g. by using the algorithms by Dinic (1970), Karzanov (1974) or the simplified version of these described in Tarjan (1984).

4.2.2. *The weighted graph partitioning problem*

This is the weighted version of the graph partitioning problem in which, with each unordered pair (i, j) of vertices, corresponding to an edge in the graph, one associates a nonnegative number W_{ij} [the weight of edge (i, j)]. Moreover, it is agreed that $W_{ij} = 0$ if (i, j) is not an edge. It is required to partition the vertex set of G into k subsets such that the weighted sum of the edges within the subsets be maximized (equivalently, the weighted sum of edges joining distinct subsets be minimized).

All that has been said in section 4.2.1. above readily extends to this weighted version, the column generation subproblem being in this case, to minimize:

$$g(y) = - \sum_{i \in E} \sum_{j \in E, \, j > i} W_{ij} y_i y_j - \sum_{j \in E} \pi_j y_j \qquad (14)$$

which is, again, a quadratic submodular pseudoboolean function, whose minimization can be performed in time $\mathcal{O}(p^3)$ through a maximun network flow algorithm. Computational experiments about this approach to the Graph Partitioning problem are under way, and will be presented in a forth coming paper.

4.3. Cases where the elements in $\mathscr{F}$ are matchings in a given graph

We now consider the case where E is the edge set of some given unoriented graph G and $\mathscr{F}$ is the whole family of matchings in G.

4.3.1. *Minimum edge coloring of an arbitrary graph (chromatic index)*

Take the cost $\gamma(S)$ of any matching $S \in \mathscr{F}$ as being equal to 1. The problem $SP[E, \mathscr{F}, \gamma]$ can then be interpreted as follows: find a partition of the edge set of G into a minimum number of matchings. This problem is well-known in graph theory as *finding the chromatic index* of G. Though polynomially solvable for some special classes of graphs (such as bipartite graphs, *see* e. g.

Berge 1970, chapter 12) it is known to be *NP*-complete in the case of an arbitrary graph (*see* Holyer 1981).

Given simplex multipliers π_1, π_2, ..., π_p associated with the edges of G, the column generation subproblem which can be stated as:

$$\underset{S \in \mathscr{F}}{\text{Minimize}} \left\{ 1 - \sum_{i \in S} \pi_i \right\} \tag{15}$$

is recognized as looking for a *maximum weight matching* in G, when the edges $i \in E$ are assigned weights π_i. The problem can be solved efficiently in time $\mathcal{O}(n^3)$ (n being the number of nodes in G) by using efficient implementations of Edmonds' algorithm (Edmonds 1965) such as those described by Lawler and Gabow (*see* Lawler 1976, Gabow 1973).

4.3.2. *Minimimum weighted edge-coloring of a graph and application to a class of scheduling problems*

As in section 4.3.1 above, E is the edge set of some given unoriented graph G and $\mathscr{F}$ is the family of matchings in G. Now, assuming that with each edge i in E a *weight* W_i has been attached, the cost of any matching $S \in \mathscr{F}$ is defined as:

$$\gamma(S) = \underset{i \in S}{\text{Max}} \{ W_i \} \tag{16}$$

The problem $SP[E, \mathscr{F}, \gamma]$ then consists in looking for a partition of the edge set of G into matchings, such that the sum of the costs of the matchings in the partition be minimized. We note that, if S is a matching and $S' \subset S$, then S' is also a matching and $\gamma(S') \leq \gamma(S)$. As a consequence, an optimal solution to $SP[E, \mathscr{F}, \gamma]$ can be readily deduced from an optimal solution to $SC[E, \mathscr{F}, \gamma]$.

It is worth observing here that the above problems $SP[E, \mathscr{F}, \gamma]$ or $SC[E, \mathscr{F}, \gamma]$ correspond to an interesting class of *scheduling problems with only disjunctive constraints* (no precedence constraint). Indeed, the edges of the given graph G can be viewed as corresponding to a set of specified tasks 1, 2, ..., p the weights W_1, W_2, ..., W_p being the *durations of these tasks*. It is assumed that any number of tasks can be carried out simultaneously provided that the corresponding edges in the associated graph do not have a common endpoint (i.e. form a matching in G). Determining a schedule of minimum duration for the whole set of tasks thus amounts to divide the tasks into subsets of independant tasks (to partition the edges of G into matchings) so that the sum of processing times for the various subsets of tasks is minimized.

But the time needed for processing a subset S of tasks corresponds to the longest processing time for the tasks in the subset: thus it is exactly equal to $\gamma(S)$ given by (16). As an application of the above model, it is interesting here to point out the connections with the *traffic matrix decomposition problem* arising in Satellite Communication Systems optimization. (Ito *et al.* 1977, Inukai 1978). In this case, the underlying graph G is bipartite, its nodes corresponding to the rows and columns of some given *traffic matrix*. The weights associated with the edges correspond to the traffic values attached to the various row-column pairs, expressed in time units. The minimization of the total switching time (the "length of the frame") then amounts to determining a minimum weighted edge coloring of the (bipartite) graph G as described above. Solving this problem as a large size set covering (or set partitioning) problem via a generalized linear programming approach was first suggested by Minoux (1984 *a*).

The column generation subproblem in this case is a follows. If π_1, π_2, ..., π_p are the simplex multipliers associated with the edges of G, then we have to determine a matching S^* such that:

$$\bar{\gamma}(S^*) = \underset{S \in \mathscr{F}}{\mathrm{Min}} \{ \bar{\gamma}(S) \}$$

where, for all $S \subset E$:

$$\bar{\gamma}(S) = \underset{i \in S}{\mathrm{Max}} \{ W_i \} - \sum_{i \in S} \pi_i. \tag{17}$$

The problem of finding, in a combinatorial family $\mathscr{F}$, an optimum element S^* with respect to a combined Min-Max-Min-Sum objective such as (17) seems to have been stated an solved for the first time in Minoux (1984 *a*) in the context of the traffic matrix decomposition problem mentioned above. The idea is to convert the problem into a sequence of standard (Min-Sum) weighted matching problems via a *threshold approach*. Let the elements of E be ordered according to nondecreasing weights $W_{i_1} \leqq W_{i_2} \leqq \ldots \leqq W_{i_p}$ and, for each $t \, (1 \leqq t \leqq p)$ let S^t be the matching such that $\sum_{i \in S^t} \pi_i$ is maximum over all the matchings of the partial graph of G induced by the subset of edges $(i_1, i_2, \ldots, i_t)$. Then $S^* \in \mathscr{F}$ minimizing (17) is given by:

$$\bar{\gamma}(S^*) = \underset{t=1, \ldots, p}{\mathrm{Min}} \{ \bar{\gamma}(S^t) \}$$

where:

$$\bar{\gamma}(S^t) = W_{i_t} - \sum_{i \in S^t} \pi_i.$$

As shown by Minoux (1984 a), in the case of a bipartite graph G with $|E| = p$ edges and n vertices, the whole computation above can be carried out in time $\mathcal{O}(pn^2)$.

The general problem of minimizing combined Min-Max-Min-Sum objectives over many classes of combinatorial families $\mathscr{F}$ is studied in Minoux (1986 b) and leads, in particular, to the following result:

THEOREM 1: *Suppose that the combinatorial family $\mathscr{F}$ on E is such that finding an optimal min-sum member, with respect to arbitrary weights π_i on the elements of E can be performed in polynomial time $T(E, \mathscr{F})$.*

Then an optimal member S^* with respect to a combined Min-Max-Min-Sum *objective such as* (17) *can be found in polynomial time $\mathcal{O}(p . T(E, \mathscr{F}))$.*

The traffic matrix decomposition problem provides a good illustration of the relevance of the large scale set covering/set partitioning reformulation of a difficult combinatorial problem (it is *NP*-hard as shown by Rendl 1984) in view of building *tight relaxations* for deriving strong lower bounds, eventually useful within Branch-and-Bound schemes. Over all the test-problems treated in Minoux (1984 a) and Ribeiro, Minoux and Penna (1986) the generalized linear programming bounds *always provided the exact integer optimum value.* This resulted in Branch-and-Bound trees of very small size, less than 10 nodes say, even for the largest problems considered: problems of size up to 15×15 were solved to optimality, whereas previous approaches (based on much weaker lower bounds) did not succeed in providing guaranteed optimal integer solutions to problems of size greater than about 6×6 (*see* e. g. Vismara 1982).

4.4. Cases where the elements in $\mathscr{F}$ are bases or independent sets of a matroïd

Let $\mathscr{F}$ be the family of independent subsets in a matroïd on

$$E = \{1, 2, \ldots, p\}.$$

4.4.1. *Partition or covering into a minimum number of independent subsets*

The problem $SP[E, \mathscr{F}, \gamma]$ where $\gamma(S) = 1$ for any independent subset $S \in \mathscr{F}$ is recognized as partitioning the ground set E into a minimum number of independent subsets.

The column generation subproblem in this case simply reduces to determining a maximum weight independent subset, where the elements of E are given weights π_i equal to the current simplex multipliers. However, solving the generalized linear programming relaxation $\overline{SP}[E, \mathcal{F}', \gamma]$ in this case does not seem of much interest since the problem has already been shown to be polynomially solvable by matroïdal techniques (*see* Nash-Williams 1966).

Note here that an other equivalent representation of the same problem would be to consider the family $\mathcal{F}'$ of *bases* (maximal independent subsets in the matroïd) and to solve the set covering problem $SC[E, \mathcal{F}', \gamma]$ (again assuming $\gamma(S) = 1$, $\forall S \in \mathcal{F}'$). Indeed, an optimal solution to $SP[E, \mathcal{F}, \gamma]$ can be deduced from an optimal solution to $SC[E, \mathcal{F}', \gamma]$ just by deleting in the appropriate subsets the elements which are covered more than once (because if S is an independent subset, then $S' \subset S$ is also independent).

4.4.2. *Special case of graphic matroïds. Minimum covering of the edge set of a connected graph with spanning trees*

We only mention this problem here which is a special case of the above problem $SC[E, \mathcal{F}', \gamma]$ where $\mathcal{F}'$ is the family of all spanning trees in a given graph G. The corresponding partitioning problem $SP[E, \mathcal{F}, \gamma]$ was studied and shown to be polynomially solvable by Nash-Wiliams (1964).

4.4.3. *Minimum weighted partitioning or covering of a matroïd with bases or independent subsets*

Examples of sections 4.4.1 and 4.4.2 have been known for a long time in matroïd theory. However, in connection with these, we would like to suggest here an interesting extension which nicely fits into our general model, and, as far as we know, is stated here for the first time (together with a readily available computational approach, *see* § 2).

Assume that every element i in E is given a weight W_i (no sign assumption is necessary on the W_i's) and define the cost $\gamma(S)$ of any independent subset S in $\mathcal{F}$ as in (16) by:

$$\gamma(S) = \underset{i \in S}{\text{Max}} \{ W_i \}.$$

The problem $SP[E, \mathcal{F}, \gamma]$ then consists in finding a minimum weight partition of E into independent subsets (the weight of a partition being the sum of the weights of the subsets which it contains).

Given simplex multipliers π_1, π_2, ..., π_p associated with the elements of E, the column generation subproblem is then to determine an independent

subset S^* minimizing

$$\bar{\gamma}(S) = \underset{i \in S}{\mathrm{Max}} \{ W_i \} - \sum_{i \in S} \pi_i. \tag{18}$$

This is a combinatorial problem with combined Min-Max-Min-Sum objective, analogous to the one discussed in section 4.3.2. Thus, according to theorem 1, it can be solved in polynomial time provided that the standard maximum weight (Max-Sum) independent subset problem can be solved in polynomial time (by means of the *greedy algorithm*, *see* Edmonds 1971).

An interesting special case is of course the one involving *graphic matroïds* in which E is the edge set and $\mathscr{F}$ is the set of all forests of a given graph G. In that case, minimizing (18) can be done by applying at most p times a minimum weighted tree algorithm (such as Prim's algorithm 1957), leading to a worst-case complexity $\mathcal{O}(pn^2)$ where n is the number of nodes in G (*see* also Minoux 1986 b).

Even in the case of graphic matroïds mentioned above, there does not seem to exist any polynomial algorithm for the minimum weighted partitioning problem when the weights are arbitrary. It is conjectured here that this problem is *NP*-hard.

4.5 – Cases where elements in $\mathscr{F}$ are common independent subsets of two matroïds on the same ground set E

Considering such cases leads to problems much alike those studied in section 4.4 above.

4.5.1. Partitioning or covering into a minimum number of subsets, for which the column generation subproblem requires the search for a *maximum weight intersection of two matroïds* (efficient algorithms have been described e.g. in Lawler 1976).

4.5.2. Weighted partitioning or covering with common independent subsets of two matroïds [the weight of the subsets being defined as in (16)] for which the column generation subproblem can be reduced to solving a sequence of standard weighted matroïd intersection problems (*see* Minoux 1986 b).

The traffic matrix decomposition problem studied in section 4.3.2 can be viewed as a special case of section 4.5.2 above since matchings in a bipartite graph are nothing but the independent sets in the intersection of two *partition matroïds* (*see* e.g. Lawler 1976 or Gondran and Minoux 1979, chapt. 9). Since, as already mentioned in section 4.5.2, even in this partitular situation,

the problem is *NP*-hard, it is apparent that problems such as in section 4.5.2 will in general be difficult to solve for arbitrary matroïds. However, the general computational approach described earlier (§ 2) can be applied to such problems.

Another interesting special case of sections 4.5.1 and 4.5.2 is the one where the two matroïds define the set of all *directed trees* (also called "branchings") of a given oriented graph *G*. As a matter of fact, it is well known that the directed trees or partial subtrees in a graph are common independent sets of a partition matroïd and the associated graphic matroïd.

Minimum cardinality partitioning or covering gives rise to a column generation subproblem consisting in the search for a maximum weight directed tree in the graph. In the weighted version, the problem involves a combined Min-Max-Min-Sum objective which can be solved by the general technique presented in section 4.5.2.

4.6. Cases where the elements of $\mathscr{F}$ are paths or circuits (or chains or cycles) in a graph

Many combinatorial problems can be formulated as large scale set covering or set partitioning problems such as $SC[E, \mathscr{F}, \gamma]$ or $SP[E, \mathscr{F}, \gamma]$ for which E is either the node set or the arc set of some given directed graph *G* and $\mathscr{F}$ is the family of all the paths or circuits in *G*. Similarly when *E* is the node set or the edge set of some unoriented graph *G* and $\mathscr{F}$ the family of chains or cycles.

We will thus describe here only two very typical examples of such problems.

4.6.1. *The m-salesman vehicle routing problem*

This corresponds to the case where each arc *u* of *G* is given a length l_u and $\mathscr{F}$ is the family of all the subsets of nodes corresponding to elementary circuits containing a specific node i_0 of *G*. Let μ be any elementary circuit containing i_0 and let S_μ denote the subset of nodes other than i_0 on the circuit. The cost $\gamma(S_\mu)$ of $S_\mu \in \mathscr{F}$ is then defined as the sum of the lengths l_u of the arcs on the circuit μ.

In addition, let *X* be the subset of $[0,1]^q$ defined by an additional constraint of type (4) (*see* § 3)

$$\sum_{j=1}^{q} x_j = m \qquad\qquad (19)$$

where m is a given positive integer (in accordance with section 3, q denotes the number of elements in the family $\mathscr{F}$, here the number of elementary circuits containing i_0 in the graph). Then the problem $SP\ [E,\ \mathscr{F},\ \gamma,\ X]$ can be interpreted as finding a set of m circuits with *minimum total length* meeting each node of the graph (other than i_0) once and exactly once. In vehicle routing, this is exactly what is referred to as *the m-salesman problem*. The column generation subproblem, in this case is as follows. Given simplex multipliers $\pi_1, \pi_2, \ldots, \pi_p$ associated with the nodes of the graph other than i_0 (thus the graph has $p+1$ vertices) and π_{i_0} associated with constraint (19), find an elementary circuit containing i_0 with minimum reduced cost where the reduced cost of any circuit μ containing i_0 is

$$\overline{\gamma}(S_\mu) = \gamma(S_\mu) - \sum_{i \in S_\mu} \pi_i - \pi_{i_0} \tag{20}$$

The above problem is a difficult one but it is well-known that, in order to have the generalized linear programming approach work, it is enough to be able to test for the existence of a negative reduced cost column. In our problem, this exactly amounts to test for the existence of a negative reduced cost circuit in the given graph with lengths l_u on the arcs u and $-\pi_i$ on the nodes i. This latter problem can be solved in polynomial time $\mathcal{O}(p^3)$ (*see* e. g. Gondran and Minoux 1979, Chapt. 2).

4.6.2. *Path-constrained network flows*

A typical example of such a problem is to find a maximum flow between two specified vertices s and t in a capacitated network (i. e. a directed graph $G = (X, U)$ together with upper capacity bounds on the arcs), when additional restrictions or constraints are imposed on the paths which can be used for sending flow from s to t.

Examples of such additional constraints are:

— given numbers w_u attached to the arcs of the graph (weights) and an upper limit $\overline{w}$, consider only those paths of total weight not greater than $\overline{w}$. Of course the weights may have many possible interpretations, depending on the specific application considered: time, variance of a probability distribution etc. (for a survey of such applications, *see* Minoux 1975),

— cardinality constraint (this is a special case of the above in which all the weights are equal),

— time constraints with time windows (*see* e. g. Halpern and Priess 1974, Desrosiers *et al.* 1984),

— more complicated constraints such as those imposed to crew pairings (or crew schedules) in some crew scheduling problems arising e. g. in airline companies (*see* Minoux 1984 *b*, Lavoie Minoux & Odier 1985).

Contrasting with the simple case of maximum flows without extra constraints, path-constrained network flows with constraints of the above types are usually *NP*-hard (*see* Garey and Johnson 1979). By using an arc-path formulation (*see* e. g. Gondran & Minoux 1979, chapt. 6) these problems can be stated as large scale (generalized) *set packing problems* very similar to the problems $SP[E, \mathscr{F}, \gamma]]$ or $SC[E, \mathscr{F}, \gamma]$ of section 2. Typically:

$$\text{Maximize} \sum_{j=1}^{q} c_j x_j \text{ subject to:}$$

$$SPK[E, \mathscr{F}, \gamma] \qquad \sum_{j=1}^{q} A_j x_j \leqq b$$

$$\forall j, \quad x_j \text{ integral}$$

where $b = (b_1, b_2, \ldots, b_p)^T$ is a right handside vector whose components can be viewed as *capacities* on the arcs of the graph (the ordinary set packing problem corresponds to the special case $b_i = 1$, $i = 1, \ldots, p$).

The column generation subproblem, in such models, consists in finding a minimum or maximum length $s - t$ path subject to the extra constraints under consideration.

In the cases where the constrained shortest or longest paths obtained are polynomially solvable, all the developments presented earlier in this paper are applicable. An interesting special instance of such problems seems to be when one imposes a cardinality constraint and the underlying graph is *acyclic*.

4.7. Cases where the elements of $\mathscr{F}$ are cutsets in a graph

Let $G = [X, U]$ be an oriented graph with node set X and arc set U. For any subset of nodes $A \subset X$, $\omega^+(A)$ denotes the associated cutset, i. e. the subset of arcs having initial endpoint in A and terminal endpoint in $X - A$. Now, we take $E = U =$ the arc set of G, and $\mathscr{F} =$ the family of cutsets of G of the form $\omega^+(A)(A \subset X)$.

4.7.1. *Maximum packing with cutsets*

We first consider the case where the cost $\gamma(S)$ of any cutset $S \in \mathscr{F}$ is equal to 1. Then the packing problem $SPK[E, \mathscr{F}, \gamma]$ introduced in section 4.6.2 can be interpreted as packing the arc set of G with a maximum number of

cutsets. Solving the linear relaxation of this problem by generalized linear programming leads to a column generation subproblem which is as follows. At any current step, the simplex multipliers are nonnegative values π_u associated with the arcs $u \in U$ of G, and we have to look for a cutset S^* having maximum reduced cost i. e.:

$$\bar{\gamma}(S^*) = 1 - \sum_{u \in S^*} \pi_u = \underset{S \in \mathscr{F}}{\text{Max}} \{ 1 - \sum_{u \in S} \pi_u \}.$$

The π_u values being nonnegative, they can be interpreted as *capacities* assigned to the arcs of G, and the above problem amounts to finding a cutset with minimum capacity in G. Obviously, this problem can be solved in polynomial time through maximum network flow computations.

We note here that, contrary to the set packing problem above, (PGA) does not hold in general for problems $SP[E, \mathscr{F}, \gamma]$ (partitioning the arc set into a minimum number of cutsets) or $SC[E, \mathscr{F}, \gamma]$ (covering the arc set into a minimum number of cutsets): in the case of set covering, the column generation subproblem would be to find a maximum capacity cutset in G w. r. t. the capacities π_u, a NP-hard problem for a general graph G; similarly for the case of set partitioning where one would have to find a minimum capacity cutset with respect to capacities π_u possibly taking negative values.

4.7.2. Relation to a linear programming formulation of the steiner tree problem in a graph

We point out here the close connection between the set packing problem $SPK[E, \mathscr{F}, \gamma]$ in section 4.7.1 and the *steiner tree problem* in a directed graph G.

Suppose that we partition the set X of vertices of G into two subsets:

X_c the set of "compulsory vertices";

X_S the set of "optional" vertices or "steiner vertices".

Moreover, we associate with each edge $u \in U$ a (nonnegative) number l_u called the length of arc u. The steiner tree problem in G is then to find a directed tree covering each compulsory vertex exactly once, having some specified vertex $i_0 \in X_c$ as a *root* and such that total length be minimized. For any node j in $X_c - \{ i_0 \}$, denote by $\mathscr{C}_{i_0 j}$ the set of all cutsets of the form $\omega^+(A)$ where $A \subset X$ and $i_0 \in A$, $j \notin A$. If, to each arc $u \in U$, we let correspond a binary variable y_u ($y_u = 1$ iff arc u is selected in the optimal steiner tree

looked for) the steiner tree problem above can be formulated as:

$$\text{Minimize} \sum_{u \in U} l_u y_u \text{ subject to}$$

$$(STP) \quad \sum_{u \in C} y_u \geq 1 \, (\forall j \in X_c - \{ i_0 \}, \, \forall C \in \mathscr{C}_{i_0 j})$$

$$y_u = 0 \quad \text{or} \quad 1 \qquad (u \in U)$$

The above formulation is due to Aneja (1980). Also, when all the vertices in G are compulsory, one finds again the integer linear programming formulation of the directed spanning tree problem due to Edmonds (1967). A survey on linear programming formulations of the steiner problem in graphs will be found in Maculan 1987.

It is easy to see that the linear program dual to $(\overline{STP})$ [the linear relaxation of (STP)] is a problem very similar to the set packing problem of section 4.7.1, the only differences being that:

— the right handsides are arbitrary nonnegative numbers l_u (instead of being 1);

— the set of columns do not correspond to the whole set of cutsets in G but is restricted to: $\bigcup_{j \in X_c - \{ i_0 \}} \mathscr{C}_{i_0 j}$.

4.7.3. *Maximum weight packings with cutsets*

The problem introduced in section 4.7.1 can be easily extended to some more general cost functions on $\mathscr{F}$. For instance, weights w_u can be assigned to the arcs $u \in U$ of G and the cost $\gamma(S)$ of a cutset S defined by:

$$\gamma(S) = \underset{u \in S}{\text{Min}} \{ w_u \}.$$

In a way similar to what occured in sections 4.3.2, 4.4.3, 4.5.2, the column generation subproblem in that case would give rise to a combinatorial problem with combined Max Min-Max Sum objective, reducible to a sequence of minimum cut problems, hence to a sequence of maximum network flow computations. This extension of the cutset packing problem seems to be new.

5. CONCLUSIONS

Though far from being exhaustive, the list of examples given in section 4 above is enough to show the great number and variety of combinatorial problems related to the new class studied here. For a few problems in the

class (e. g. the one presented in section 4.3.2) the numerical results obtained have already confirmed the relevance of the generalized linear programming approach from a computational point of view.

From a theoretical point of view, it was shown in section 3 that the common fundamental feature to all the problems dealt with here was the polynomial solvability of the large scale set covering/set partitioning linear relaxations. It is our feeling that this could be a useful step towards getting better insights into the class of NP-complete problems by providing a theoretically consistent distinction (or "hierarchy of difficulty") among problems which, up to now, were all considered in a sense as "indistinguishably difficult".

To be more specific, the distinction suggested by the above developments between "easier NP-complete or NP-hard problems" and "harder NP-complete or NP-hard problems" could be based on the fact that, once formulated as large size set covering/set partitioning problems as in section 2, the (PGA) assumption holds or does not hold.

According to this criterion, one could say, for instance, that the (one dimensional) *cutting stock problem* belongs to the "harder class" in view of the generalized linear programming approach proposed in their pioneering work by Gilmore and Gomory (1963): this is because, in this case, the generation subproblem involves the solution of an integer knapsack problem.

It should be noted, here, that numerous examples of the "harder class" could easily be deduced as variants of some of the examples stated in section 4. For instance, in the weighted graph partitioning model of section 4.2.2, if the weights W_{ij} are allowed to be negative, then a hard quadratic 0-1 optimization problem is found.

Another instance of this is provided by the scheduling problems of section 4.3.2 if minimum weighted edge-coloring is replaced by minimum weighted vertex-coloring (in which case, the generation subproblem involves stable sets instead of matchings). Though such examples correspond to situations where column generation may be computationally infeasible in bad cases, it seems that the models and algorithms presented here should not be hastily ruled out, since practically efficient algorithms are known anyway for some NP-hard problems (e. g. knapsack problems, travelling salesman, etc.).

REFERENCES

Y. P. ANEJA, *An Integer Linear Programming Approach to the Steiner Problem in Graphs*, Networks, Vol. 10, 1980, pp. 167-178.

E. BALAS and M. W. PADBERG, *On the Set Covering Problem, II. An Algorithm for Set Partitioning*, Operations Research, Vol. 23, 1975, pp. 74-90.

E. BALAS and M. W. PADBERG, *Set Partitioning: a Survey* in *Combinatorial Optimization*, N. CHRISTOFIDES, A. MINGOZZI, P. TOTH and C. SANDI Eds., J. Wiley and Sons, 1979, pp. 151-210.

E. BALAS and HO, *Set Covering Algorithms Using Cutting Planes, Heuristics, and Subgradient Optimization: A Computational Study*, Mathematical Programming Study, Vol. 12, 1980, pp. 37-60.

C. BERGE, *Graphes et Hypergraphes*, Dunod, Paris, 1970.

A. BILLIONNET and M. MINOUX, *Maximizing a Supermodular Pseudoboolean Function: a Polynomial Algorithm for Supermodular Cubic Functions*, Discrete Applied Mathematics, Vol. 12, 1985, pp. 1-11.

A. BILLIONNET and M. MINOUX, *Bounds and Efficient Algorithms for Submodular Pseudoboolean Function Minimization*, 1984, To appear under the title: *Duality Results and related Algorithms for Submodular function minimization*.

G. B. DANTZIG, *Linear Programming and Extensions*, Princeton University Press, Princeton, 1963.

G. B. DANTZIG and P. WOLFE, *The Decomposition Algorithm for Linear Programming*, Econometrica, Vol. 29, No. 4, 1961, pp. 767-778.

J. DELORME, *Contribution à la résolution du problème de recouvrement : méthodes de troncatures*, Thèse de Docteur Ingénieur, Université Paris-VI, 1974.

J. DESROSIERS, F. SOUMIS and M. DESROCHERS, *Routing with Time Windows by Column Generation*, Networks, Vol. 14, 1984, pp. 545-565.

E. A. DINIC, *Algorithm for Solution of a Problem of Maximum Flow in a Network with Power Estimation*, Soviet Math. Dokl., Vol. 11, 1970, pp. 1277-1280.

J. EDMONDS, *Maximum Matching and a Polyhedron with 0-1 Vertices*, Journal Res. Nat. Bureau Standards, Vol. 69-B, Nos. 1-2, 1965, pp. 125-130.

J. EDMONDS, *Optimum Branchings*, J. Res. Nat. Bur. Stand. B, Vol. 71, 1967, pp. 233-240.

J. EDMONDS, *Submodular Functions, Matroïds, and Certain Polyhedra* in *Combinatorial Structures and their Applications*, R. GUY Ed., Gordon and Breach, New York, 1970, pp. 69-87.

J. EDMONDS, *Matroïds and the Greedy Algorithm*, Mathematical Programming, Vol. 1, 1971, pp. 127-136.

M. L. FISCHER, G. L. NEMHAUSER and L. A. WOLSEY, *An Analysis of Approximations for Maximizing Submodular Set Functions I*, Math. Programming, Vol. 14, 1978, pp. 265-294.

H. N. GABOW, *Implementations of Algorithms for Maximum Matchings on Non-Bipartite Graphs*, Ph. D. Dessertation, Computer Sc. Department, Stanford University, California, 1973.

M. R. GAREY and D. S. JOHNSON, *Computers and Intractability. An Introduction to the Theory of NP-Completeness*, W. H. Freeman and Co., San Francisco, 1979.

R. S. GARFINKEL and G. L. NEMHAUSER, *The Set Partitioning Problem: Set Covering with Equality Constraints*, Operations Research, Vol. 17, 1969, pp. 848-856.

P. C. GILMORE and R. E. GOMORY, *A Linear Programming Approach to the Cutting-Stock Problem. Part II*, Operations Research, Vol. 11, 1963, pp. 863-888.

R. E. GOMORY and T. C. HU, *An Application of Generalized Linear Programming to Network Flows*, Journal S.I.A.M., Vol. 10, No. 2, 1962, pp. 260-283.

M. GONDRAN and M. MINOUX, *Graphes et Algorithmes*, Eyrolles, Paris, 1979, English translation, J. Wiley and Sons, 1983.

M. GRÖTSCHEL, L. LOVÁSZ and SCHRIJVER, *The Ellipsoïd Method and its Consequences in Combinatorial Optimization*, Combinatorica, Vol. 1, No. 2, 1981, pp. 169-197.

S. L. HAKIMI, *Optimum Location of Switching Centers and the Absolute Centers and Medians of a Graph*, Operations Research, Vol. 12, 1964, pp. 450-459.

J. HALPERN and I. PRIESS, Shortest Path with Time Constraints on Movement and Parking, *Networks*, Vol. 4, 1974, pp. 241-253.

P. HANSEN, D. PEETERS and J. F. THISSE, *Facility Location Analysis*, Rutcor Research Report 4-85, Rutgers University (NJ), 1985, Encyclopedia of Economics (to appear).

P. HANSEN, Private communication, 1986.

I. HOLYER, *The NP-Completeness of Edge-Coloring*, S.I.A.M. J. Comput., Vol. 10, No. 4, 1981, pp. 718-720.

T. INUKAI, *Comments on Analysis of a Switch Matrix for an SS/TDMA System*, Proceedings of the I.E.E.E., Vol. 66, No. 12, 1978, pp. 1669-1670.

Y. ITO, Y. URANO, T. MURATANI and M. YAMAGUCHI, *Analysis of a Switch Matrix for an SS/TDMA System*, Proceedings of the I.E.E.E., Vol. 65, No. 3, 1977, pp. 411-419.

A. V. KARZANOV, *Determining the Maximal Flow in a Network by the Method of Preflows*, Soviet Math. Dokl., Vol. 15, 1974, pp. 434-437.

L. G. KHACHIAN, *A Polynomial Algorithm in Linear Programming*, Soviet Math Dokl., Vol. 20, No. 1, 1979, pp. 191-194.

L. S. LASDON, *Optimization Theory for Large Systems*, Macmillan series for Operations Research, 1970.

S. LAVOIE, M. MINOUX and E. ODIER, *A new Approach of Crew Pairing Problems by Column Generation and Application to Air Transports*, 1985 (to Appear).

E. LAWLER, *Combinatorial Optimization. Networks and Matroïds*, Holt, Rinehart and Winston, 1976.

C. E. H. LEMKE, M. SALKIN and K. SPIELBERG, *Set Covering by Single Branch Enumeration with Linear Programming Subproblems*, Operations Research, Vol. 19, 1971, pp. 998-1022.

N. MACULAN, *The Steiner Problem in Graphs*, in *Surveys in Combinatorial Optimization*, S. MARTELLO, G. LAPORTE, M. MINOUX and C. RIBEIRO Eds., North Holland, 1987.

V. M. MALHOTRA, M. P. KUMAR and S. N. MAHESHWARI, *An $O(|V|^3)$ Algorithm for finding Maximum Flows in Networks*, Information Processing Letters, Vol. 7, No. 6, 1978, pp. 277-278.

F. E. MARANZANA, *On the Location of Supply Points to Minimize Transport Costs*, Operational Research Quarterly, Vol. 15, 1964, pp. 261-270.

R. E. MARSTEN, *An Algorithm for Large Set Partitioning Problems*, Management Science, Vol. 20, 1974, pp. 779-787.

M. MINOUX, *Plus court chemin avec contraintes, algorithmes et applications*, Annales des Télécommunications, Vol. 30, Nos. 11-12, 1975, pp. 383-394.

M. MINOUX, *Programmation Mathématique : Théorie et Algorithmes*, Dunod Paris, 1983, English Translation, J. Wiley and Sons, 1986.

M. MINOUX, *Optimal Traffic Assignment in a SS/TDMA Frame: A New Approach by Set Covering and Column Generation*, O.R.S.A./T.I.M.S. meeting, Dallas, Texas, 1984 a, R.A.I.R.O., Vol. 20, No. 4, 1986, pp. 1-13.

M. MINOUX, *Column Generation Techniques in Combinatorial Optimization: A New Application to Crew-Pairing Problems*, XXIVth AGIFORS Symposium, 1984 b, Strasbourg, France, september 1984.

M. MINOUX, *New Lower Bounds to the Graph Partitioning Problem through Generalized Linear Programming and Network Flows*, 1986 a (to be published).

M. MINOUX, *Solving Combinatorial Problems with Combined Min-Max-Min-Sum objective*, 1986 a (under preparation).

C. St. J. A. NASH-WILLIAMS, *Decomposition of Finite Graphs into Forests*, Journal London Math. Soc., Vol. 39, 1964, p. 12.

C. St. J. A. NASH-WILLIAMS, *An Application of Matroïds to Graph Theory* in *Théorie des Graphes*, Proceedings Internat. Symposium, Rome, Italy, 1966, Dunod, Paris, 1967.

R. C. PRIM, *Shortest Connection Networks and Some Generalizations*, Bell Syst. Techn. J., Vol. 36, 1957, p. 1389.

F. RENDL, *On the Complexity of Decomposing Matrices Arising in Satellite Communications*, Bericht 84-47, Technische Universität, Graz, Austria, 1984.

J. M. W. RHYS, *A Selection Problem of Shared Fixed Costs and Network Flows*, Management Science, Vol. 17, No. 3, 1970, pp. 200-207.

C. RIBEIRO, M. MINOUX and C. PENNA, *A Hybrid Branch and Bound/Column Generation Approach to Very Large Scale Set Covering Problems with Special Structure*, O.R.S.A./T.I.M.S. meeting, Miami, Florida, November 1986 (to Appear).

N. Z. SHOR, *Cut-off Methods with Space Extension in Convex Programming Problems*, Kibernetika, Vol. 13, No. 1, 1977, pp. 94-95, English Translation in Cybernetics, Vol. 13, No. 1, 1977, pp. 94-96.

R. E. TARJAN, *A simple Version of Karzanov's Blocking Flow Algorithm*, Operations Research Letters, Vol. 2, No. 6, 1984, pp. 265-268.

L. VISMARA, *Piano di accesso dei messaggi in sistemi SS/TDMA: un metodo di enumerazione implicita per minimizzare il tempo di transmissione*, Tesi di laurea, politechnico di Milano, Departamento di Elettronica (Paolo Camerini i Francesco Maffioli Relatori), 1982.