

\\ 140 \\

**Exact solution of the SONET Ring
Loading Problem**

di

**Mauro Dell'Amico
Martine Labbé
Francesco Maffioli**

Maggio 1996

**Università degli Studi di Modena
Dipartimento di Economia Politica
Viale Berengario, 51
41100 Modena (Italia)
e-mail: dellamico@unimo.it**

Exact Solution of the SONET Ring Loading Problem

Mauro Dell'Amico^{a1}, Martine Labbé^b, Francesco Maffioli^c

^a *Dipartimento di Economia Politica, Università di Modena,
via Berengario, 51, 41100 Modena, Italy.*

^b *Institut de Statistique et Recherche Opérationnelle and
Service de Mathématiques de la Gestion, Université Libre de Bruxelles,
CP 210/01, Boulevard de Triomphe, B-1050 Bruxelles, Belgium*

^c *Dipartimento di Elettronica e Informazione, Politecnico di Milano,
P.zza L. da Vinci 32, 20133 Milano, Italy*

Abstract

In this paper we address the problem of planning the capacity of the local rings in Synchronous Optical NETWORKS (SONET). We present efficient lower and upper bound procedures and a branch and bound algorithm which is able to find the exact solution of large instances within short computing times.

Keywords: Communications. Programming, linear: algorithms, Programming, integer: branch-and-bound.

1 Introduction

One of the most successful ways to exploit the great potential of fiber optics technology is offered by the recent introduction of the Synchronous Digital Hierarchy (SDH) in Europe and Synchronous Optical NETWORK (SONET) in the US. This has suggested a number of new optimization problems, most of them in the field of Combinatorial Optimization. We address here the problem of planning the capacity of a single bi-directional ring of n fiber links connecting n Add-and-Drop Multiplexers (ADM) corresponding to n customer (or set of customers) locations.

This problem has been first studied in [1] where it is called the SONET Ring Loading Problem. To the author's knowledge no exact approach has yet been proposed for this problem, which is known to be NP-hard [1], but not yet satisfactorily classified with respect to its approximability status: in particular it is still unknown if it can have a Polynomial Time Approximation Scheme. Only when the demands are all equal to one the problem is known to be polynomially solvable [2].

After having formulated the problem and presented a simple heuristic having a factor of 2 approximability guarantee in Section 2, we present in Section 3 a strongly polynomial algorithm for the continuous relaxation of the integer formulation. This is exploited in

¹Corresponding author: E-mail: dellamic@elet.polimi.it

Section 4 within an exact branch-and-bound approach, whose performances are extensively tested as shown by the computational results in Section 5.

2 Formulation and heuristic solution

We are given a *ring* graph $G = (V, E)$ that is a simple circuit with node set $V = \{1, \dots, n\}$ and edge set $E = \{1, \dots, n\}$. We also consider a set of m demands $D = \{1, \dots, m\}$. Each demand i is defined by an *origin* node o_i , a *destination* node d_i and a *weight* w_i which represent the amount of traffic which must be routed from o_i to d_i along the ring. The entire traffic of a demand must be routed either clockwise or counter-clockwise and it cannot be splitted to be transmitted on both directions contemporarily. Each edge $\ell \in E$ may be initially loaded with a fixed traffic $t_\ell \geq 0$, which is independent of the demands in D (this traffic is due to pre-routed demands). The problem we consider is to determine the routing of each demand in D , in order to minimize the total load of an edge.

Without loss of generality we can assume that all the input data are integers.

In order to present a mathematical model we need to introduce some notation. For each demand i let Π_i^+ denote the path from node o_i to node d_i , clockwise, and Π_i^- the path from o_i to d_i , counter-clockwise. For each demand $i \in D$ let x_i be a boolean variable which takes value one if the traffic w_i is routed clockwise, value zero otherwise. We also use the continuous variable z to identify the maximum load of an edge. The model is as follows.

$$(P) \quad \min z \tag{1}$$

$$\text{s.t. } z \geq \sum_{k:\ell \in \Pi_k^+} w_k x_k + \sum_{k:\ell \in \Pi_k^-} w_k (1 - x_k) + t_\ell, \quad \forall \ell \in E, \tag{2}$$

$$x_k \in \{0, 1\}, \quad \forall k \in D. \tag{3}$$

The objective function (1) minimizes the maximum load of an edge. Constraints (2) ensure that the value of variable z is not less than the load of each edge $\ell \in E$. In particular the first sum in (2) gives the total load of edge ℓ due to the demands routed clockwise; the second sum gives the load due to the demands routed counter-clockwise. Constraints (3) impose that the variables are binary.

The continuous relaxation C(P) of problem P is obtained by allowing the variables x_i to assume any real value in $[0, 1]$, i.e. substituting (3) with

$$0 \leq x_k \leq 1, \quad \forall k \in D. \tag{3'}$$

Then a demand can be splitted routing part of the traffic w_i clockwise and the remaining traffic counter-clockwise.

A feasible solution for problem P can be obtained from the solution of the continuous relaxation by rounding each variable with fractional value to the closest integer. This corresponds to route each *fractional* demand k (i.e a demand associated with a value x_k with $0 < x_k < 1$) either clockwise (counter-clockwise) if $x_k \geq 0.5$ ($x_k < 0.5$). We call HR

the heuristic algorithm which determine an approximate solution by applying this simple rounding technique.

Theorem 2.1 *The value of the solution obtained by algorithm HR is at most two times the value of the optimal solution*

Proof. Algorithm HR routes the entire traffic of a demand k along the path Π_k^+ if $x_k \geq 0.5$ and along Π_k^- , otherwise. It follows that the traffic on an edge in the approximate solution is at most two times the traffic routed on that edge in the continuous relaxation. Thus the maximum load of an edge in the heuristic solution is at most $2L$, were L is the value of the continuous relaxation. Since L is a lower bound on the optimum solution value the result follows. \square

Algorithm HR can be improved by introducing a post-processing of the first K demands with largest weight, having fractional values. In particular we start by rounding the values of the variables of all demands, except that of the K fractional selected. Then we enumerate the 2^K solutions obtained with all the possible routing combinations of the K fractional demands and we choose the best integer solution. If K is a constant independent of the input data the procedure can be implemented to run in polynomial time. We call this improved algorithm KHR.

In the next section we show how to solve efficiently problem C(P). The effectiveness of procedure KHR is evaluated in Section 5.

3 A strongly polynomial algorithm for the continuous relaxation

Before giving the details of our efficient algorithm to solve problem C(P), we need to introduce some further notation. For each pair of distinct edges $r \in E, s \in E$ let $C_{r,s}$ denote the set of *crossing demands*, i.e. the set of demands which must be routed exactly on one of the two edges r, s . Formally,

$$C_{r,s} = \{k \in D : r \in \Pi_k^+ \text{ and } s \in \Pi_k^- \text{ or } r \in \Pi_k^- \text{ and } s \in \Pi_k^+\}.$$

Any pair of distinct edges (r, s) partitions the node set into two subsets, and is thus called a *straight cut* of G .

In the following we will speak indifferently of the pair of distinct edges (r, s) or of the cut (r, s) . Let τ denote the set of all possible straight cuts of G .

Given a demand index $i \in D$ and a partial solution $x_k, k = 1, \dots, i$, let

$$F_\ell^i = \sum_{k \leq i: \ell \in \Pi_k^+} w_k x_k + \sum_{k \leq i: \ell \in \Pi_k^-} w_k (1 - x_k), \quad \ell \in E$$

denote the total traffic on edge ℓ , due to the first i demands. In a solution to C(P), the load of an edge ℓ is then $F_\ell^n + t_\ell$.

We start the description of our algorithm by introducing a simple lower bound on the value of the optimal solution to C(P). Given a cut (r, s) one can see that the traffic of each crossing demand must be routed through edge r or edge s , so

$$L1 = \frac{1}{2} \max_{(r,s) \in \tau} \left(\sum_{k \in C_{r,s}} w_k + t_r + t_s \right) \quad (4)$$

is a valid lower bound.

Since the optimal load of an edge $\ell \in E$ is at least t_ℓ , an improved bound is

$$L = \max\{L1, L2\}, \quad (5)$$

with $L2 = \max_{\ell \in E} t_\ell$. The bound L is the optimal solution value for problem C(P). To prove this we give an $O(mn^2)$ algorithm, called CSONET, which determines a feasible solution to C(P).

Our algorithm considers a demand at a time and routes it optimally by determining the maximum traffic which can be routed clockwise (or counter-clockwise) without exceeding the bound L . More precisely we have m iterations, one for each demand. At iteration i the routing of the first $i - 1$ demands is already fixed so that the edges of the ring have preassigned loads and we consider demand i and the path Π_i^+ . For all edges and straight cuts in Π_i^+ we compute two values $C1$ and $C2$, similar to $L1$ and $L2$, which give the total amount of traffic which have to be certainly routed on some edge of Π_i^+ , disregarding demand i . We do the same for path Π_i^- and we determine the gaps Δ^+ and Δ^- between the lower bound L and the above quantities. If $\Delta^+ \geq \Delta^-$ then we route the traffic equal to the minimum of w_i and Δ^+ along Π_i^+ , and the possibly remaining traffic of demand i along Π_i^- . A similar routing is made if $\Delta^- \geq \Delta^+$, but with paths Π_i^+ and Π_i^- exchanged.

The details of procedure CSONET are given in the pseudocode below.

Algorithm CSONET

begin

 compute $L1 = \frac{1}{2} \max_{(r,s) \in \tau} \left(\sum_{k \in C_{r,s}} w_k + t_r + t_s \right)$;

 compute $L = \max(L1, \max_{\ell \in E} t_\ell)$;

for each $\ell \in E$ **do** $F_\ell^0 := 0$;

for $i := 1$ **to** n **do**

$C1 := \frac{1}{2} \max_{(r,s) \in \Pi_i^+} \left(\sum_{\substack{k \geq i \\ k \in C_{r,s}}} w_k + F_r^{i-1} + F_s^{i-1} + t_r + t_s \right)$;

$C2 := \max_{\ell \in \Pi_i^+} (t_\ell + F_\ell^{i-1})$

$\Delta^+ := L - \max(C1, C2)$;

$C1 := \frac{1}{2} \max_{(r,s) \in \Pi_i^-} \left(\sum_{\substack{k \geq i \\ k \in C_{r,s}}} w_k + F_r^{i-1} + F_s^{i-1} + t_r + t_s \right)$;

$C2 := \max_{\ell \in \Pi_i^-} (t_\ell + F_\ell^{i-1})$

```

 $\Delta^- := L - \max(C1, C2);$ 
if  $\Delta^+ \geq \Delta^-$  then
     $x_i := \min(1, \frac{\Delta^+}{w_i});$ 
else
     $x_i := \max(0, 1 - \frac{\Delta^-}{w_i});$ 
endif
endfor
end.

```

The following theorem proves the correctness of algorithm CSONET

Theorem 3.1 *Algorithm CSONET determines an optimal solution of problem C(P) and the optimal solution value is L.*

Proof. We show that at the end of each iteration i of the main loop **for** in CSONET, the following two conditions hold:

$$L \geq \frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{r,s}}} w_k + F_r^i + F_s^i + t_r + t_s \right) \quad \forall (r, s) \in \tau, \quad (6)$$

$$L \geq t_\ell + F_\ell^i, \quad \forall \ell \in E. \quad (7)$$

If condition (7) is satisfied at the end of the n -th iteration, then no edge ℓ has load greater than L . Since value L is a lower bound on the optimal solution value to C(P), then the solution provided by algorithm CSONET is optimal and the result follows.

We now prove, by induction, that conditions (7)–(6) hold at each iteration. We will give the details of the proof for the case $\Delta^+ \geq \Delta^-$, the prove for the other case ($\Delta^+ < \Delta^-$) can be obtained with similar arguments.

The two conditions (7) and (6) are certainly satisfied at the beginning of the procedure (iteration 0), by definition of L . Now suppose they hold at the end of iteration $i - 1$ ($0 < i \leq n$): we will show that they hold also at the end of iteration i .

From the definition of $C1$ and $C2$, and the fact that conditions (7) and (6) are satisfied at the end of iteration $i - 1$, we have that $C1 \leq L, C2 \leq L$, so $\Delta^+ \geq 0$ and the algorithm routes $w_i x_i$ ($= \min(w_i, \Delta^+)$) traffic units of demand i , on each edge $\ell \in \Pi_i^+$.

We first prove that the thesis holds for the edges and the straight cuts in Π_i^+ . From the above considerations we have that $w_i x_i \leq \Delta^+ \leq L - C2$ and $C2 \geq t_\ell + F_\ell^{i-1}$ for each $\ell \in \Pi_i^+$, so that $L \geq t_\ell + F_\ell^{i-1} + w_i x_i$. Since $F_\ell^i = F_\ell^{i-1} + w_i x_i$, condition (7) immediately holds. Similarly, for condition (6) we have

$$w_i x_i \leq \Delta^+ \leq L - C1 \leq L - \frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{r,s}}} w_k + F_r^{i-1} + F_s^{i-1} + t_r + t_s \right)$$

for each $r \in \Pi_i^+$, $s \in \Pi_i^+$. Since demand i is not included in $C_{r,s}$ for all $r \in \Pi_i^+$ and $s \in \Pi_i^+$ we have

$$L \geq \frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{r,s}}} w_k + F_r^{i-1} + F_s^{i-1} + 2w_i x_i + t_r + t_s \right) = \frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{r,s}}} w_k + F_r^i + F_s^i + t_r + t_s \right)$$

and condition (6) holds for any straight cut in Π_i^+ .

We now prove that the two conditions (6), (7) hold for the edges and the straight cuts in Π_i^- . If $x_i = 1$ then no traffic from demand i is routed on path Π_i^- , so the only remaining case is $0 < \Delta^+ < w_i$ (i.e. the algorithm routes Δ^+ traffic units along path Π_i^+ and $w_i - \Delta^+$ along Π_i^-). By definition of Δ^+ we know that after iteration i at least one of the following two cases arises:

- case (i) an edge ℓ in Π_i^+ is *saturated* i.e. $t_\ell + F_\ell^{i-1} + w_i x_i = t_\ell + F_\ell^i = L$,
case (ii) a straight cut (r, s) with $r \in \Pi_i^+$, $s \in \Pi_i^+$ is saturated, i.e.

$$\frac{1}{2} \left(\sum_{\substack{k \geq i \\ k \in \bar{C}_{r,s}}} w_k + F_r^{i-1} + F_s^{i-1} + t_r + t_s + 2\Delta^+ \right) = \frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{r,s}}} w_k + F_r^i + F_s^i + t_r + t_s \right) = L.$$

In the remaining of the proof we refer to four particular edges r, s, u, v (see Figure 1), with $r \in \Pi_i^+$, $s \in \Pi_i^+$, $u \in \Pi_i^-$, $v \in \Pi_i^-$ and such that edge r precedes edge s , if we move clockwise from o_i to d_i , and u precedes v , if we move counter-clockwise from o_i to d_i .

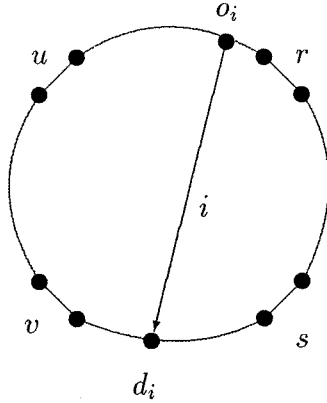


Figure 1: The four particular edges r, s, u and v .

Case (i). Let r denote the edge in Π_i^+ saturated by the routing of demand i , i.e.

$$t_r + F_r^{i-1} + \Delta^+ = L \quad (8)$$

Since condition (6) holds, at the end of iteration $i - 1$, for any cut (r, j) with $r \in \Pi_i^+$ and $j \in \Pi_i^-$, we have that

$$\frac{1}{2} \left(\sum_{\substack{k \geq i \\ k \in \bar{C}_{j,r}}} w_k + F_j^{i-1} + F_r^{i-1} + t_j + t_r \right) = \frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{j,r}}} w_k + w_i + F_j^{i-1} + F_r^{i-1} + t_j + t_r \right) \leq L \quad (9)$$

Multiplying inequality (9) by 2 and subtracting equation (8) we obtain:

$$\sum_{\substack{k > i \\ k \in C_{j,r}}} w_k + w_i + F_j^{i-1} - \Delta^+ + t_j \leq L.$$

Observing that $F_j^i = F_j^{i-1} + w_i - \Delta^+$, for each edge $j \in \Pi_i^-$, we immediately conclude that $t_j + F_j^i \leq L$, and condition (7) holds.

Condition (6) has to be considered for straight cuts in Π_i^- only if Π_i^- contains at least two distinct edges, say u and v . Therefore we can refer to the two cuts (u, r) and (v, r) . Adding two inequalities of the type (9) for $j = v$ and $j = u$, respectively, and subtracting (8) we obtain

$$\frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{u,r}}} w_k + \sum_{\substack{k > i \\ k \in C_{v,r}}} w_k + 2w_i + F_u^{i-1} + F_v^{i-1} + t_u + t_v - 2\Delta^+ \right) \leq L.$$

Reminding again that $F_u^i = F_u^{i-1} + w_i - \Delta^+$ and that $F_v^i = F_v^{i-1} + w_i - \Delta^+$, we obtain

$$\frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{u,r}}} w_k + \sum_{\substack{k > i \\ k \in C_{v,r}}} w_k + F_u^i + F_v^i + t_u + t_v \right) \leq L.$$

It is now sufficient to observe that $C_{u,v} \subseteq C_{u,r} \cup C_{v,r}$ to derive

$$\frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{u,v}}} w_k + F_u^i + F_v^i + t_u + t_v \right) \leq L,$$

which shows that condition (6) holds for any straight cut (u, v) in Π_i^- .

Case(ii). Let (r, s) with $r \in \Pi_i^+$ and $s \in \Pi_i^+$ denote the saturated cut, i.e.

$$\frac{1}{2} \left(\sum_{\substack{k \geq i \\ k \in C_{r,s}}} w_k + F_r^{i-1} + F_s^{i-1} + t_r + t_s \right) + \Delta^+ = L \quad (10)$$

and consider an edge $j \in \Pi_i^-$ and the two cuts (j, r) and (j, s) . Note that inequality (9) can be rewritten with s instead of r , so that adding the two inequalities of type (9) associated with cuts (j, r) and (j, s) , and observing that: $C_{r,s} \subseteq C_{j,r} \cup C_{j,s}$ we obtain

$$t_j + w_i + F_j^{i-1} + \frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{r,s}}} w_k + F_r^{i-1} + F_s^{i-1} + t_r + t_s \right) \leq 2L$$

Subtracting equation (10) and reminding that: (a) demand $i \notin C_{r,s}$; (b) $F_j^i = F_j^{i-1} + w_i - \Delta^+$ we obtain $t_j + F_j^i \leq L$ and condition (7) holds.

We conclude the proof by showing that condition (6) holds also for any cut (u, v) with $u \in \Pi_i^-$ and $v \in \Pi_i^-$. Consider the two cuts (v, r) and (u, s) and observe that $C_{u,v} \cup C_{r,s} \subseteq C_{v,r} \cup C_{u,s}$. Adding the two inequalities of type (9) associated to the cuts (u, s) and (v, r) , and subtracting equation (10) we obtain

$$\frac{1}{2} \left(\sum_{\substack{k > i \\ k \in C_{u,v}}} w_k + F_u^{i-1} + F_v^{i-1} + t_u + t_v + 2(w_i - \Delta^+) \right) \leq L$$

Since $F_\ell^i = F_\ell^{i-1} + w_\ell - \Delta^+$, $\forall \ell \in \Pi_i^-$ condition (6) immediately follows. \square

3.1 Complexity and implementation

Algorithm CSONET performs m iterations, one for each demand. At the generic iteration i we have to compute two times the quantities $C1$ and $C2$. Each evaluation of $C2$ requires $O(n)$ time, whereas to determine the value of $C1$ we have to consider all the possible straight cuts with both edges in Π_i^+ (or in Π_i^-) and for each cut (r, s) we have to compute

$$\sum_{\substack{k \geq i \\ k \in C_{r,s}}} w_k + F_r^{i-1} + F_s^{i-1} + t_r + t_s. \quad (11)$$

A naive implementation requires $O(m)$ time for each evaluation of (11), and $O(mn^2)$ time to determine $C1$. A more efficient implementation can be obtained using a matrix $A = [a_{i,j}]$ to store the $n \times (n-1)/2$ sums $\{\sum w_k : k \geq i, k \in C_{r,s}\}$, for all cuts (r, s) of the ring. With this data structure we need $O(1)$ time to evaluate (11) and $O(n^2)$ time to determine $C1$. Matrix A needs to be initialized at iteration 0 of the algorithm and updated when each iteration is concluded. The construction of the initial matrix can be done in $O(mn)$ time, as we show here below. The updating, instead, requires $O(n^2)$ time since it is enough to consider all the $(n^2 - n)/2$ cuts (r, s) and subtract w_i from $a_{r,s}$ if $i \in C_{r,s}$. Therefore the overall time complexity of algorithm CSONET is $O(mn^2)$.

Moreover we note that if matrix A is given, then the lower bound $L1$ can be computed in $O(n^2)$, and the lower bound $L2$ can be obtained in $O(n)$. Therefore if one is only interested into the value L of the solution to C(P) (and not to the corresponding values of x_i) the entire computation requires $O(mn)$ time against the $O(mn^2)$ time necessary to have the complete solution.

We now show how to compute the initial matrix A in $O(mn)$ time. We select each edge r in turn and we compute the weight $a_{r,s} = \sum_{k \in C_{r,s}} w_k + t_r + t_s$ for all $s \in E, s \neq r$. The computation of the values $a_{r,s}$ starts with s being the first edge following r , clockwise, and continue choosing s as the next edge of the ring, clockwise, until we reach r . At the first step, when r and s are adjacent, the value of $a_{r,s}$ is given by the sum of all demands with origin or destination in the node common to the edges r and s . When we consider a cut (r, s') immediately after the computation of $a_{r,s}$, with s and s' being adjacent edges

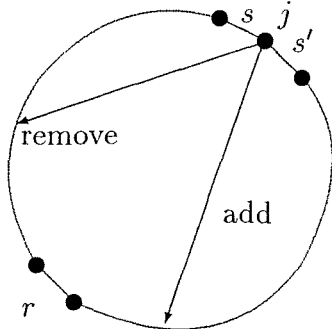


Figure 2: The case of adjacent edges while computing matrix A .

(see Figure 2), we do not need to compute the new value from scratch, but it is enough to set $a_{r,s'} = a_{r,s}$, and then to remove from $a_{r,s'}$ the weights of the demands which have their origin or destination in the node j common to s and s' and destination or origin on the path from r to s , clockwise, and to add to $a_{r,s'}$ the weights of the demands which have origin or destination on node j and destination or origin on the path from s' to r , clockwise. For a given edge r , during the computation of all the values $a_{r,s}$, we consider each demand at most twice, so giving a time complexity of $O(m)$. Repeating this procedure for the $n - 1$ possible choices of r we obtain the overall time complexity of $O(mn)$.

4 An exact algorithm

The results of the previous sections have been used to develop a depth-first branch-and-bound algorithm for the exact solution of problem P. Our basic algorithm is as follows.

We use a binary decision tree. At the each node we solve problem C(P) modified by fixing the routing of a set of demands. Among the demands routed with a fractional traffic, in the solution to C(P), we determine the one having maximum traffic. Two child nodes are generated by assigning this demand either clockwise or counter-clockwise.

At the root node we set $t_\ell = 0$ for each $\ell \in E$, then we apply algorithm CSONET of the previous section to solve problem C(P) and to obtain the lower bound L . The continuous solution is used by algorithm KHR of Section 2 to obtain an heuristic solution of value UB . If $L = UB$ we are done, otherwise we start the exploration of the branching decision tree. When a demand i is routed along a path π (i.e. a forward step is performed) we increase by w_i the values t_ℓ for each $\ell \in \pi$. When a backtracking occurs the value w_i is subtracted from t_ℓ , for all $\ell \in \pi$. Observe that the loads t_ℓ completely describe the traffic due to a partial solution, therefore no modification in the algorithms of Sections 2 and 3 is required in order to compute lower and upper bounds for the intermediate nodes of the tree.

Preliminary computational experiments performed with instances randomly generated

as in [1] show that the time required by our algorithm to find an optimal solution is one order of magnitude less than the time required by CPLEX to find an integer solution which is within 0.5% of the optimum (for more details on the experiments with CPLEX see [1]).

Stimulated by these results we tried to improve the performances of our basic algorithm with several techniques.

Presorting of the demands. We observed experimentally that the value of the heuristic solution obtained with KHR is very close to the optimum value. Therefore we decided to change the branching rule. We do not select the fractional variable with maximum traffic, but the first fractional variable in a presorted list. This list is defined in such a way that the first descent in the branch-decision-tree, from the root node to a leaf, builds exactly the heuristic solution. Our guess is that a limited number of backtrackings (i.e. routing changes) are required to obtain the optimal solution. We also tried a presorting by nonincreasing weights.

Improvement of the lower bound. Given the solution to C(P), we consider the K largest demands with fractional value (where K is the same parameter introduced in Section 2 to obtain the improved heuristic algorithm). Similarly to procedure KHR, we enumerate the 2^K partial solutions obtained with all the possible routing combinations of these K demands and we apply the lower bound procedure to the remaining $m - K$ demands of each partial solution. The minimum of these bounds is a valid lower bound which improves bound L . We apply this technique to all the nodes of the tree.

Second lower bound. Given a straight cut (r, s) and the crossing demands $C_{r,s}$, we know that each of these demands must be routed through edge r or s . Thus minimizing the maximum load on r and s , with an integer assignment of the demands in $C_{r,s}$, gives a valid lower bound on the optimum solution value. This problem can be transformed into a *subset-sum problem* in which a set of q items is given, each of which has an associated weight, and a subset S of the q items which maximizes the sum of the weights of the item in S without exceeding a given capacity c is sought. Given a straight cut (r, s) , we define an instance of subset-sum with capacity $c = \frac{1}{2} \sum_{i \in C_{r,s}} w_i$ and with an item i , having weight w_i , associated to each demand $i \in C_{r,s}$. The optimal solution to the subset-sum problem minimizes the difference between c and the sum of the weights of the items selected, so it also minimizes the difference between the items selected and discarded. In terms of the SONET problem this solution minimizes the difference of the traffic loads on r and s and thus the maximum of the two. For each of the $n(n-1)/2$ straight cuts we solve the corresponding subset-sum problem with a C translation of the FORTRAN routine contained in the book by Martello and Toth [3]. We apply this bound to all the nodes of the tree.

All the above improvements of the basic branch-and-bound have been tested with several computational experiments. The impact of each technique on the computational

times is the following.

- Each kind of *presorting* leads to smaller computational times for some instances and largest for other instances, so, on average, no improvement can be obtained with this technique.
- The *improvement of the lower bound* halves the computational times.
- The *second lower bound* has value larger than that of the continuous relaxation only for a few instances. In these cases, it allows to reduce drastically the size of the branch-decision-tree, but on average the computational time required to compute the bound is not compensated by the reduction of the tree.

From the above preliminar computational experiments we have seen that significative improvements are obtained only with a strenghtening of the lower bound coming from continuous relaxation. In order to obtain the maximum information from this bound we have developed the following *fixing procedure* which tries to fix the values of the fractional variables either to zero or one. The procedure performs a series of iterations. At each iteration we consider, in turn, each demand, say j , with fractional value in the optimal solution to C(P). We route the entire traffic of demand j clockwise updating the values t_ℓ accordingly and we compute again the lower bound L . If the lower bound is larger than or equal to the current best solution value then the demand must be routed counter-clockwise to have a chance to obtain a better solution value. If the value of the fractional variable has not been fixed by this tentative, we route the demand j counter-clockwise and we repeat the same steps to try to fix it. When all the fractional demands have been considered we compute the new solution of the continuous relaxation of the problem obtained from P by fixing the routing of some demands, and a new iteration is started. The algorithm stops when, at one iteration, no demand have been fixed. The pseudocode follows.

Procedure FIXING

input: upper bound value U ;

repeat

 compute lower bound L ; find the optimal solution x^* of C(P);

 NOFIX := TRUE;

for each demand j with $x_j^* \neq \lfloor x_j^* \rfloor$ **do**

 route demand j clockwise; compute lower bound L ;

if $L \geq U$ **then**

 fix $x_j^* := 0$; let NOFIX := FALSE;

else

 route demand j counter-clockwise; compute lower bound L ;

if $L \geq U$ **then** fix $x_j^* := 1$; let NOFIX:=FALSE;

endif

endfor

until (NOFIX = TRUE)

The time complexity of procedure FIXING is $O(I(mn^2 + Fmn))$ where I is the number of iteration performed and F the number of fractional variables. In the worst case it is $O(m^3n^2)$, but in practice, since the number of fractional demands is small, it is quite fast.

The use of this procedure at each node of the branch-decision-tree reduced the computational times of one order of magnitude. Moreover we noted that with this procedure the improved lower bound has no more effect. So our final code is the basic branch-and-bound with procedure FIXING applied at each node.

5 Computational results

We have coded in C language the heuristic procedure *KHR* of Section 2, the lower bound procedure *CSONET* of Section 3 and the branch-and-bound of the previous Section. Moreover we have implemented two greedy heuristics as those described in [1]. The first heuristic, that we call 1PG, is a One Phase Greedy which consider a demand at a time and tentatively routes it clockwise and counter-clockwise. The demand is routed on the path which determines the minimum increasing in the current solution value. The second heuristic, say 2PG, is a Two Phase Greedy which tentatively routes the current demand clockwise and counter-clockwise, then complete the partial solution with 1PG and routes the demand along the path associated with the minimum solution value. This algorithm is slightly different from that in [1], since we use 1PG to determine the routing of each demand, instead Cosares and Saniee apply 1PG only if there is a tie in the partial load determined by the routing of the current demenad clockwise or counter-clockwise.

We generated random instances as in [1]: the weights of the demands are randomly selected according to the uniform distribution in $[5,100]$ and origins and destinations are different integers also uniformly randomly selected. The results for this class of problems, with values of $n = (20,40,60,80,100)$ and values of m ranging from 50 to 1000 are reported in Table I.

Let z^H be the solution value of the generic heuristic H for a specific instance, and let z^* be its otimum solution value (computed using our branch-and-bound algorithm). For each instance we compute the percentage error respect to the optimal solution, i.e. $err = 100(z^H - z^*)/z^*$. The columns labelled Δ_{avg} and Δ_{max} , in the table, report, respectively, the average and the maximum percentage error. The columns labelled *time* give the average computing time in seconds on a PC Pentium with a clock at 100 Mhertz. For each entry in the tables 20 instances have been generated and solved. The labels F_{avg} and F_{max} indicate, respectively, the average and maximum number of fractional variables obtained in the solution of the continuous relaxation.

For these experiments we fixed the parameter K of heuristic *KHR* to five.

Table I. Uniformly random instances with $w_j \in [5, 100]$: heuristic solutions

n	m	KHR					1PG			2PG		
		Δ_{avg}	Δ_{max}	F_{avg}	F_{max}	$time$	Δ_{avg}	Δ_{max}	$time$	Δ_{avg}	Δ_{max}	$time$
20	50	2.3	6.3	1.55	3	0.01	16.8	40.0	0.01	9.7	20.1	0.04
	100	1.2	2.7	1.60	3	0.04	20.2	50.1	0.01	12.0	21.6	0.12
	200	0.5	1.0	1.20	3	0.04	21.2	32.2	0.01	12.0	18.0	0.69
40	100	1.1	2.9	1.75	3	0.08	19.2	29.3	0.01	12.3	22.7	0.33
	200	0.6	1.7	1.40	2	0.20	24.0	31.6	0.01	14.3	19.6	1.44
	400	0.3	0.6	1.30	3	0.46	25.8	31.2	0.04	17.9	24.1	5.80
60	150	0.7	1.8	1.50	5	0.36	18.5	31.6	0.01	9.6	17.5	1.19
	300	0.5	1.0	1.45	3	0.78	23.6	32.6	0.04	14.6	19.6	4.86
	600	0.2	0.5	1.00	1	1.60	25.0	32.6	0.12	18.0	25.0	19.44
80	200	0.5	1.4	1.65	3	0.90	21.1	33.9	0.01	13.4	21.3	2.18
	400	0.3	0.7	0.95	2	1.81	24.1	37.6	0.08	16.0	21.1	8.73
	800	0.1	0.3	1.15	3	3.58	26.7	31.0	0.28	20.4	23.7	34.85
100	250	0.6	1.2	1.65	3	1.72	23.2	28.8	0.04	15.6	21.1	4.23
	500	0.2	0.6	1.11	2	2.29	24.0	27.2	0.09	15.9	20.2	16.71
	1000	0.1	0.2	1.12	2	4.81	27.3	35.9	0.22	20.8	23.0	67.01

Pentium/100 seconds; averages over 20 instances.

Procedure *KHR* dominates the other heuristics and it is fast. Its average errors are very small: they exceeds one percent only for small instances ($n = 20$, $m \leq 100$, and $n = 40$, $m = 100$). The maximum error never exceeds a few percent. Heuristics 1PG and 2PG have errors one order of magnitude larger than that of *KHR*. 1PG has negligible computing time, but produces the worst solutions. The average error of 2PG is one third to one half less that of 1PG, but its computing time is two order of magnitude larger. *KHR* has computing times larger than that of 1PG, but very small: less than five seconds for the largest instances.

It is worth noting that currently the real instances have $n \leq 20$ and $m \leq 200$ and the largest weight of a demand is bounded by 100, so the first three rows of our table give a sample of real life problems.

The same instances of Table I have been solved with the branch-and-bound algorithm of the previous Section. The results are given in Table II. For each pair n, m we give the average and maximum computing time (over 20 instances), the average and maximum number of explored nodes, the average and maximum depth of the branch-decision-tree and the ratio between the value of the lower bound at the root node ($L1$) and the optimum solution value (z^*).

Table II. Uniformly random instances with $w_j \in [5, 100]$: exact solution

n	m	Time		Exp. Nodes		Depth		$L1/z^*$
		<i>avg</i>	<i>max</i>	<i>avg</i>	<i>max</i>	<i>avg</i>	<i>max</i>	
20	50	0.26	0.82	35.8	215	6.3	12	0.994
	100	1.02	2.55	44.6	127	10.2	22	0.997
	200	2.68	7.86	23.0	77	9.3	33	0.993
40	100	2.49	7.70	46.0	209	10.1	25	0.998
	200	7.71	17.71	38.0	133	12.2	28	0.996
	400	36.40	81.89	87.0	323	21.4	57	0.996
60	150	9.83	22.16	42.3	103	14.1	33	0.999
	300	24.17	71.92	43.1	263	13.5	43	0.999
	600	233.94	2541.64	674.4	12761	19.1	72	0.999
80	200	25.30	109.95	56.8	507	13.7	50	0.999
	400	135.31	682.79	185.3	1637	19.2	48	0.999
	800	251.03	949.85	44.8	219	18.5	80	0.999
100	250	41.97	129.26	52.0	208	15.4	32	0.992
	500	162.16	752.41	253.8	2191	22.7	63	0.999
	1000	304.84	1117.81	72.0	338	24.0	114	0.999

Pentium/100 seconds; averages over 20 instances.

All the instances with size as those of the largest real life problems ($n \leq 20$ and $m \leq 200$) are solved by the exact approach with small computing times. Larger instances are also solved in reasonable computing times, but the maximum times grow up to one hour. The average number of explored nodes is, in general, small. This is mainly due to our fixing procedure (see Section 4). For large instances however the lower bound at the root node, in many cases, is equal to the optimum solution value, but the algorithm determines a feasible solution with this value only after visiting a large number of nodes.

To better understand the behaviour of the heuristic *KHR* and of the branch-and-bound procedure we generated instances from three further classes: (i) $w_j \in [5, 1000]$; (ii) $w_j \in [50, 100]$ and (iii) $w_j \in [500, 1000]$.

The performances of the three heuristics are very similar to those obtained with weights $w_j \in [5, 100]$, but the new instances are more difficult for the branch-and-bound. The results are reported in Tables III-V: the numbers in brackets give the number of instances which have not been solved within one hour of CPU time (in these cases the averages are computed on the number of solved instances).

Table III. Uniformly random instances with $w_j \in [5, 1000]$: exact solution

n	m	Time		Exp. Nodes		Depth		$L1/z^*$
		<i>avg</i>	<i>max</i>	<i>avg</i>	<i>max</i>	<i>avg</i>	<i>max</i>	
20	50	0.16	1.57	399.9	1569	199.7	784	0.998
	100	0.47	1.57	376.3	1195	187.9	597	0.999
	200	3.93	15.71	636.7	1632	318.1	816	1.000
40	100	2.99	6.29	516.3	1123	257.9	561	0.997
	200	20.27	103.71	1396.8	9094	698.0	4547	1.000
	400	39.60	80.14	557.8	1989	278.6	994	1.000
60	150	15.09	33.00	903.6	2490	451.7	1245	1.000
	300	24.04	73.86	310.6	1414	155.0	707	1.000
	600	55.00	128.86	160.3	666	79.8	333	1.000
80	200	20.43	48.71	333.6	894	166.5	447	1.000
	400	73.86	198.00	308.1	943	153.8	471	1.000
	800	260.32	(1)831.29	4813.8	33127	2406.6	16563	0.998
100	250	50.29	136.71	541.6	1608	270.6	804	1.000
	500	161.07	781.00	782.0	6512	390.9	3256	1.000
	1000	374.79	1145.57	197.1	666	98.3	333	1.000

Pentium/100 seconds; averages over 20 instances.

Table IV. Uniformly random instances with $w_j \in [50, 100]$: exact solution

n	m	Time		Exp. Nodes		Depth		$L1/z^*$
		<i>avg</i>	<i>max</i>	<i>avg</i>	<i>max</i>	<i>avg</i>	<i>max</i>	
20	50	0.31	3.14	330.3	2382	164.9	1191	0.994
	100	2.20	7.86	916.2	3244	458.0	1622	1.000
	200	1.26	4.71	91.8	441	45.7	220	0.999
40	100	4.09	17.29	498.3	1735	249.0	867	1.000
	200	16.81	106.86	1068.5	7902	534.1	3951	0.999
	400	33.52	(1)138.29	746.5	4394	373.3	2197	0.999
60	150	8.96	26.71	479.6	2020	239.6	1010	0.998
	300	42.74	262.43	735.7	6272	367.6	3136	0.999
	600	106.86	660.00	195.7	1516	97.8	758	1.000
80	200	60.34	180.71	1181.6	3952	590.7	1976	0.999
	400	46.04	161.86	366.8	2605	183.2	1302	0.999
	800	143.47	424.29	145.9	609	72.8	304	1.000
100	250	348.07	2669.86	42582.5	394236	21291.1	197118	0.999
	500	79.99	220.00	85.2	229	42.3	114	1.000
	1000	229.74	534.29	135.7	717	67.8	358	1.000

Pentium/100 seconds; averages over 20 instances.

Table V. Uniformly random instances with $w_j \in [500, 1000]$: exact solution

n	m	Time		Exp. Nodes		Depth		$L1/z^*$
		<i>avg</i>	<i>max</i>	<i>avg</i>	<i>max</i>	<i>avg</i>	<i>max</i>	
20	50	4.40	15.71	3442.7	11066	1721.1	5533	0.995
	100	4.09	14.14	1838.3	6947	919.0	3474	0.997
	200	29.86	103.71	6109.6	20117	3054.7	10058	0.999
40	100	15.09	48.71	2843.7	10667	1421.6	5333	0.998
	200	47.14	201.14	4274.4	13821	2137.0	6910	0.997
	400	348.07	1796.14	11493.3	57066	5746.5	28533	1.000
60	150	107.49	271.86	7317.2	22844	3658.4	11422	0.999
	300	247.19	595.57	24586.2	193262	12293.0	96631	1.000
	600	226.60	1293.29	9066.6	32528	4533.1	16264	1.000
80	200	234.77	895.71	6001.7	12460	3000.7	6230	0.998
	400	454.61	3034.43	24827.5	77485	12413.5	38742	1.000
	800	1331.85	(2)2742.14	8428.1	22283	4214.3	11141	0.998
100	250	480.80	1535.28	7704.1	36110	3851.8	18055	0.998
	500	256.14	3177.45	11912.3	52994	5956.0	26497	0.999
	1000	1002.67	(2)2819.56	32456.2	113245	11087.4	112341	0.999

Pentium/100 seconds; averages over 20 instances.

6 Some final remarks

During the final writing of this work it came to our attention that part of it, namely the arguments upon which the algorithm for the continuous relaxation of Section 3 is based has been independently developed in an undated report by Schrijver, Seymour and Winkler [4]. In this report the authors develop a very efficient heuristic for SONET Ring Loading Problem, and also prove that it is $3/2$ -Approximable by a simple Greedy rounding algorithm. Moreover their algorithm finds the optimum solution whenever all demands are equal to one.

Whenever the computing time available is sufficient it is obviously much more satisfactory to use the exact method presented here. As computational results of Section 5 tend to show, this exact approach is quite efficient and probably suitable in most practical situations.

Acknowledgements

This work has been partially sponsored by Human Capital and Mobility Contract n. ERBCHRCXT930087. The third author thanks the NSFR of Belgium for financial support

References

- [1] S. Cosares and I. Saniee, “An optimization problem related to balancing loads on SONET rings”, *Telecommunication Systems* **3**, 165–181 (1994).
- [2] A. Frank, T. Nishiseki, N. Saito, H. Suzuki and E. Tardos, “Algorithms for routing around a rectangle”, *Discrete Applied Mathematics* **40**, 363–378 (1992).
- [3] S. Martello and P. Toth, *Knapsack Problems, Algorithms and Computer Implementations*. J. Wiley & sons, Chichester (1992).
- [4] A. Schrijver, P. Seymour and P. Winkler, “The Ring Loading Problem”, *unpublished manuscript*.

1. Maria Cristina Marcuzzo [1985] "Yoan Violet Robinson (1903-1983)", pp. 134
2. Sergio Lugaresi [1986] "Le imposte nelle teorie del sovrappiù", pp. 26
3. Massimo D'Angelillo e Leonardo Paggi [1986] "PCI e socialdemocrazie europee. Quale riformismo?", pp. 158
4. Gian Paolo Caselli e Gabriele Pastrello [1986] "Un suggerimento hobsoniano su terziario ed occupazione: il caso degli Stati Uniti 1960/1983", pp. 52
5. Paolo Bosi e Paolo Silvestri [1986] "La distribuzione per aree disciplinari dei fondi destinati ai Dipartimenti, Istituti e Centri dell'Università di Modena: una proposta di riforma", pp. 25
6. Marco Lippi [1986] "Aggregations and Dynamic in One-Equation Econometric Models", pp. 64
7. Paolo Silvestri [1986] "Le tasse scolastiche e universitarie nella Legge Finanziaria 1986", pp. 41
8. Mario Forni [1986] "Storie familiari e storie di proprietà. Itinerari sociali nell'agricoltura italiana del dopoguerra", pp. 165
9. Sergio Paba [1986] "Gruppi strategici e concentrazione nell'industria europea degli elettrodomestici bianchi", pp. 56
10. Nerio Naldi [1986] "L'efficienza marginale del capitale nel breve periodo", pp. 54
11. Fernando Vianello [1986] "Labour Theory of Value", pp. 31
12. Piero Ganugi [1986] "Risparmio forzato e politica monetaria negli economisti italiani tra le due guerre", pp. 40
13. Maria Cristina Marcuzzo e Annalisa Rosselli [1986] "The Theory of the Gold Standard and Ricardo's Standard Comodity", pp. 30
14. Giovanni Solinas [1986] "Mercati del lavoro locali e carriere di lavoro giovanili", pp. 66
15. Giovanni Bonifati [1986] "Saggio dell'interesse e domanda effettiva. Osservazioni sul cap. 17 della General Theory", pp. 42
16. Marina Murat [1986] "Betwin old and new classical macroeconomics: notes on Lejonhufvud's notion of full information equilibrium", pp. 20
17. Sebastiano Brusco e Giovanni Solinas [1986] "Mobilità occupazionale e disoccupazione in Emilia Romagna", pp. 48
18. Mario Forni [1986] "Aggregazione ed esogeneità", pp. 13
19. Sergio Lugaresi [1987] "Redistribuzione del reddito, consumi e occupazione", pp. 17
20. Fiorenzo Sperotto [1987] "L'immagine neopopulista di mercato debole nel primo dibattito sovietico sulla pianificazione", pp. 34
21. M. Cecilia Guerra [1987] "Benefici tributari nel regime misto per i dividendi proposto dalla commissione Sarcinelli: una nota critica", pp. 9
22. Leonardo Paggi [1987] "Contemporary Europe and Modern America: Theories of Modernity in Comparative Perspective", pp. 38
23. Fernando Vianello [1987] "A Critique of Professor Goodwin's 'Critique of Sraffa'", pp. 12
24. Fernando Vianello [1987] "Effective Demand and the Rate of Profits. Some Thoughts on Marx, Kalecki and Sraffa", pp. 41
25. Anna Maria Sala [1987] "Banche e territorio. Approccio ad un tema geografico-economico", pp. 40
26. Enzo Mingione e Giovanni Mottura [1987] "Fattori di trasformazione e nuovi profili sociali nell'agricoltura italiana: qualche elemento di discussione", pp. 36
27. Giovanna Procacci [1988] "The State and Social Control in Italy During the First World War", pp. 18
28. Massimo Matteuzzi e Annamaria Simonazzi [1988] "Il debito pubblico", pp. 62
29. Maria Cristina Marcuzzo (a cura di) [1988] "Richard F. Kahn. A discipline of Keynes", pp. 118
30. Paolo Bosi [1988] "MICROMOD. Un modello dell'economia italiana per la didattica della politica fiscale", pp. 34
31. Paolo Bosi [1988] "Indicatori della politica fiscale. Una rassegna e un confronto con l'aiuto di MICROMOD", pp. 25
32. Giovanna Procacci [1988] "Protesta popolare e agitazioni operaie in Italia 1915-1918", pp. 45
33. Margherita Russo [1988] "Distretto Industriale e servizi. Uno studio dei trasporti nella produzione e nella vendita delle piastrelle", pp. 157
34. Margherita Russo [1988] "The effect of technical change on skill requirements: an empirical analysis", pp. 28
35. Carlo Grillenzoni [1988] "Identification, estimations of multivariate transfer functions", pp. 33
36. Nerio Naldi [1988] "'Keynes' concept of capital", pp. 40
37. Andrea Ginzburg [1988] "locomotiva Italia?", pp. 30
38. Giovanni Mottura [1988] "La 'persistenza' secolare. Appunti su agricoltura contadina ed agricoltura familiare nelle società industriali", pp. 40
39. Giovanni Mottura [1988] "L'anticamera dell'esodo. I contadini italiani della 'restaurazione contrattuale' fascista alla riforma fondiaria", pp. 40
40. Leonardo Paggi [1988] "Americanismo e riformismo. La socialdemocrazia europea nell'economia mondiale aperta", pp. 120
41. Annamaria Simonazzi [1988] "Fenomeni di isteresi nella spiegazione degli alti tassi di interesse reale", pp. 44
42. Antonietta Bassetti [1989] "Analisi dell'andamento e della casualità della borsa valori", pp. 12
43. Giovanna Procacci [1989] "State coercion and worker solidarity in Italy (1915-1918): the moral and political content of social unrest", pp. 41
44. Carlo Alberto Magni [1989] "Reputazione e credibilità di una minaccia in un gioco bargaining", pp. 56
45. Giovanni Mottura [1989] "Agricoltura familiare e sistema agroalimentare in Italia", pp. 84
46. Mario Forni [1989] "Trend, Cycle and 'Fortuitous cancellation': a Note on a Paper by Nelson and Plosser", pp. 4
47. Paolo Bosi, Roberto Golinelli, Anna Stagni [1989] "Le origini del debito pubblico e il costo della stabilizzazione", pp. 26
48. Roberto Golinelli [1989] "Note sulla struttura e sull'impiego dei modelli macroeconomici", pp. 21
49. Marco Lippi [1989] "A Short Note on Cointegration and Aggregation", pp. 11
50. Gian Paolo Caselli e Gabriele Pastrello [1989] "The Linkage between Tertiary and Industrial Sector in the Italian Economy: 1951-1988. From an External Dependence to an International One", pp. 40
51. Gabriele Pastrello [1989] "Francois quesnay: dal Tableau Zig-zag al Tableau Formule: una ricostruzione", pp. 48
52. Paolo Silvestri [1989] "Il bilancio dello stato", pp. 34
53. Tim Mason [1990] "Tre seminari di storia sociale contemporanea", pp. 26
54. Michele Lalla [1990] "The Aggregate Escape Rate Analysed through the Queueing Model", pp. 23
55. Paolo Silvestri [1990] "Sull'autonomia finanziaria dell'università", pp. 11
56. Paola Bertolini, Enrico Giovannetti [1990] "Uno studio di 'filiera' nell'agroindustria. Il caso del Parmigiano Reggiano", pp. 164
57. Paolo Bosi, Roberto Golinelli, Anna Stagni [1990] "Effetti macroeconomici, settoriali e distributivi dell'armonizzazione dell'IVA", pp. 24
58. Michele Lalla [1990] "Modelling Employment Spells from Emilia Labour Force Data", pp. 18

59. Andrea Ginzburg [1990] "Politica Nazionale e commercio internazionale", pp. 22
60. Andrea Giommi [1990] "La probabilità individuale di risposta nel trattamento dei dati mancanti", pp. 13
61. Gian Paolo Caselli e Gabriele Pastrello [1990] "The service sector in planned economies. Past experiences and future prospectives", pp. 32
62. Giovanni Solinas [1990] "Competenze, grandi industrie e distretti industriali. Il caso Magneti Marelli", pp. 23
63. Andrea Ginzburg [1990] "Debito pubblico, teorie monetarie e tradizione civica nell'Inghilterra del Settecento", pp. 30
64. Mario Forni [1990] "Incertezza, informazione e mercati assicurativi: una rassegna", pp. 37
65. Mario Forni [1990] "Misspecification in Dynamic Models", pp. 19
66. Gian Paolo Caselli e Gabriele Pastrello [1990] "Service Sector Growth in CPE's: An Unsolved Dilemma", pp. 28
67. Paola Bertolini [1990] "La situazione agro-alimentare nei paesi ad economia avanzata", pp. 20
68. Paola Bertolini [1990] "Sistema agro-alimentare in Emilia Romagna ed occupazione", pp. 65
69. Enrico Giovannetti [1990] "Efficienza ed innovazione: il modello "fondi e flussi" applicato ad una filiera agro-industriale", pp. 38
70. Margherita Russo [1990] "Cambiamento tecnico e distretto industriale: una verifica empirica", pp. 115
71. Margherita Russo [1990] "Distretti industriali in teoria e in pratica: una raccolta di saggi", pp. 119
72. Paolo Silvestri [1990] "La Legge Finanziaria. Voce dell'enciclopedia Europea Garzanti", pp. 8
73. Rita Paltrinieri [1990] "La popolazione italiana: problemi di oggi e di domani", pp. 57
74. Enrico Giovannetti [1990] "Illusioni ottiche negli andamenti delle Grandezze distributive: la scala mobile e l'appiattimento delle retribuzioni in una ricerca", pp. 120
75. Enrico Giovannetti [1990] "Crisi e mercato del lavoro in un distretto industriale: il bacino delle ceramiche. Sez. I", pp. 150
76. Enrico Giovannetti [1990] "Crisi e mercato del lavoro in un distretto industriale: il bacino delle ceramiche. Sez. II", pp. 145
78. Antonietta Bassetti e Costanza Torricelli [1990] "Una riqualificazione dell'approccio bargaining alla selezioni di portafoglio", pp. 4
77. Antonietta Bassetti e Costanza Torricelli [1990] "Il portafoglio ottimo come soluzione di un gioco bargaining", pp. 15
79. Mario Forni [1990] "Una nota sull'errore di aggregazione", pp. 6
80. Francesca Bergamini [1991] "Alcune considerazioni sulle soluzioni di un gioco bargaining", pp. 21
81. Michele Grillo e Michele Polo [1991] "Political Exchange and the allocation of surplus: a Model of Two-party competition", pp. 34
82. Gian Paolo Caselli e Gabriele Pastrello [1991] "The 1990 Polish Recession: a Case of Truncated Multiplier Process", pp. 26
83. Gian Paolo Caselli e Gabriele Pastrello [1991] "Polish firms: Private Vices Public Virtues", pp. 20
84. Sebastiano Brusco e Sergio Paba [1991] "Connessioni, competenze e capacità concorrenziali nell'industria della Sardegna", pp. 25
85. Claudio Grimaldi, Rony Hamoui, Nicola Rossi [1991] "Non Marketable assets and households' Portfolio Choice: a Case of Study of Italy", pp. 38
86. Giulio Righi, Massimo Baldini, Alessandra Brambilla [1991] "Le misure degli effetti redistributivi delle imposte indirette: confronto tra modelli alternativi", pp. 47
87. Roberto Fanfani, Luca Lanini [1991] "Innovazione e servizi nello sviluppo della meccanizzazione agricola in Italia", pp. 35
88. Antonella Caiumi e Roberto Golinelli [1992] "Stima e applicazioni di un sistema di domanda Almost Ideal per l'economia italiana", pp. 34
89. Maria Cristina Marcuzzo [1992] "La relazione salari-occupazione tra rigidità reali e rigidità nominali", pp. 30
90. Mario Biagioli [1992] "Employee financial participation in enterprise results in Italy", pp. 50
91. Mario Biagioli [1992] "Wage structure, relative prices and international competitiveness", pp. 50
92. Paolo Silvestri e Giovanni Solinas [1993] "Abbandoni, esiti e carriera scolastica. Uno studio sugli studenti iscritti alla Facoltà di Economia e Commercio dell'Università di Modena nell'anno accademico 1990/1991", pp. 30
93. Gian Paolo Caselli e Luca Martinelli [1993] "Italian GPN growth 1890-1992: a unit root or segmented trend representatin?", pp. 30
94. Angela Politi [1993] "La rivoluzione fraintesa. I partigiani emiliani tra liberazione e guerra fredda, 1945-1955", pp. 55
95. Alberto Rinaldi [1993] "Lo sviluppo dell'industria metalmeccanica in provincia di Modena: 1945-1990", pp. 70
96. Paolo Emilio Mistrulli [1993] "Debito pubblico, intermediari finanziari e tassi d'interesse: il caso italiano", pp. 30
97. Barbara Pistoresi [1993] "Modelling disaggregate and aggregate labour demand equations. Cointegration analysis of a labour demand function for the Main Sectors of the Italian Economy: 1950-1990", pp. 45
98. Giovanni Bonifati [1993] "Progresso tecnico e accumulazione di conoscenza nella teoria neoclassica della crescita endogena. Una analisi critica del modello di Romer", pp. 50
99. Marcello D'Amato e Barbara Pistoresi [1994] "The relationship(s) among Wages, Prices, Unemployment and Productivity in Italy", pp. 30
100. Mario Forni [1994] "Consumption Volatility and Income Persistence in the Permanent Income Model", pp. 30
101. Barbara Pistoresi [1994] "Using a VECM to characterise the relative importance of permanent and transitory components", pp. 28
102. Gian Paolo Caselli and Gabriele Pastrello [1994] "Polish recovery from the slump to an old dilemma", pp. 20
103. Sergio Paba [1994] "Imprese visibili, accesso al mercato e organizzazione della produzione", pp. 20
104. Giovanni Bonifati [1994] "Progresso tecnico, investimenti e capacità produttiva", pp. 30
105. Giuseppe Marotta [1994] "Credit view and trade credit: evidence from Italy", pp. 20
106. Margherita Russo [1994] "Unit of investigation for local economic development policies", pp. 25
107. Luigi Brighi [1995] "Monotonicity and the demand theory of the weak axioms", pp. 20
108. Mario Forni e Lucrezia Reichlin [1995] "Modelling the impact of technological change across sectors and over time in manufacturing", pp. 25
109. Marcello D'Amato and Barbara Pistoresi [1995] "Modelling wage growth dynamics in Italy: 1960-1990", pp. 38
110. Massimo Baldini [1995] "INDIMOD. Un modello di microsimulazione per lo studio delle imposte indirette", pp. 37
111. Paolo Bosi [1995] "Regionalismo fiscale e autonomia tributaria: l'emersione di un modello di consenso", pp. 38
112. Massimo Baldini [1995] "Aggregation Factors and Aggregation Bias in Consumer Demand", pp. 33
113. Costanza Torricelli [1995] "The information in the term structure of interest rates. Can stochastic models help in resolving the puzzle?" pp. 25
114. Margherita Russo [1995] "Industrial complex, pôle de développement, distretto industriale. Alcune questioni sulle unità di indagine nell'analisi dello sviluppo." pp. 45

115. Angelika Moryson [1995] "50 Jahre Deutschland. 1945 - 1995" pp. 21
116. Paolo Bosi [1995] "Un punto di vista macroeconomico sulle caratteristiche di lungo periodo del nuovo sistema pensionistico italiano." pp. 32
117. Gian Paolo Caselli e Salvatore Curatolo [1995] "Esistono relazioni stimabili fra dimensione ed efficienza delle istituzioni e crescita produttiva? Un esercizio nello spirito di D.C. North." pp. 11
118. Mario Forni e Marco Lippi [1995] "Permanent income, heterogeneity and the error correction mechanism." pp. 21
119. Barbara Pistoresi [1995] "Co-movements and convergence in international output. A Dynamic Principal Components Analysis" pp. 14
120. Mario Forni e Lucrezia Reichlin [1995] "Dynamic common factors in large cross-section" pp. 17
121. Giuseppe Marotta [1995] "Il credito commerciale in Italia: una nota su alcuni aspetti strutturali e sulle implicazioni di politica monetaria" pp. 20
122. Giovanni Bonifati [1995] "Progresso tecnico, concorrenza e decisioni di investimento: una analisi delle determinanti di lungo periodo degli investimenti" pp. 25
123. Giovanni Bonifati [1995] "Cambiamento tecnico e crescita endogena: una valutazione critica delle ipotesi del modello di Romer" pp. 21
124. Barbara Pistoresi e Marcello D'Amato [1995] "La riservatezza del banchiere centrale è un bene o un male? Effetti dell'informazione incompleta sul benessere in un modello di politica monetaria." pp. 32
125. Barbara Pistoresi [1995] "Radici unitarie e persistenza: l'analisi univariata delle fluttuazioni economiche." pp. 33
126. Barbara Pistoresi e Marcello D'Amato [1995] "Co-movements in European real outputs" pp. 20
127. Antonio Ribba [1996] "Ciclo economico, modello lineare-stocastico, forma dello spettro delle variabili macroeconomiche" pp. 31
128. Carlo Alberto Magni [1996] "Repeatable and una tantum real options a dynamic programming approach" pp. 23
129. Carlo Alberto Magni [1996] "Opzioni reali d'investimento e interazione competitiva: programmazione dinamica stocastica in optimal stopping" pp. 26
130. Carlo Alberto Magni [1996] "Vaghezza e logica fuzzy nella valutazione di un'opzione reale" pp. 20
131. Giuseppe Marotta [1996] "Does trade credit redistribution thwart monetary policy? Evidence from Italy" pp. 20
132. Mauro Dell'Amico e Marco Trubian [1996] "Almost-optimal solution of large weighted equicut problems" pp. 30
133. Carlo Alberto Magni [1996] "Un esempio di investimento industriale con interazione competitiva e avversione al rischio" pp. 20
134. Margherita Russo, Peter Börkey, Emilio Cubel, François Lévêque, Francisco Mas [1996] "Local sustainability and competitiveness: the case of the ceramic tile industry" pp. 66
135. Margherita Russo [1996] "Camionetto tecnico e relazioni tra imprese" pp. 190
136. David Avra Lane, Irene Poli, Michele Lalla, Alberto Roverato [1996] "Lezioni di probabilità e inferenza statistica" pp. 288
137. David Avra Lane, Irene Poli, Michele Lalla, Alberto Roverato [1996] "Lezioni di probabilità e inferenza statistica - Esercizi svolti -" pp. 302
138. Barbara Pistoresi [1996] "Is an Aggregate Error Correction Model Representative of Disaggregate Behaviours? An example" pp. 24
139. Luisa Malaguti e Costanza Torricelli [1996] "Monetary policy and the term structure of interest rates", pp. 30

