

\\ 173 \\

**A Linear Time Algorithm for Scheduling Outforests
with Communication Delays on Two or
Three Processors**

by

Mauro Dell'Amico

May 1997

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

A Linear Time Algorithm for Scheduling Outforests with Communication Delays on Two or Three Processors

Mauro Dell'Amico

*Dipartimento di Economia Politica, Università di Modena
via J. Berengario 51, 41100 Modena, Email: dellamic@elet.polimi.it*

We consider the problem of scheduling n unit-length tasks on identical m parallel processors, when outforest precedence relations and unit interprocessor communication delays exist. Two algorithms have been proposed in the literature for the exact solution of this problem: a linear time algorithm for the special case $m = 2$, and a dynamic programming algorithm which runs in $O(n^{2m-2})$. In this paper we give a new linear time algorithm for instances with $m = 2$ and $m = 3$.

1. INTRODUCTION

In this paper we consider scheduling problems arising in the management of parallel programs on a distributed memory multiprocessor system. A parallel program is usually represented by a digraph G where each node corresponds to a *task* and the existence of an arc (i, j) between nodes i and j , means that task j requires as an input the results produced by task i . A number of identical *processors* are given, each of which can execute at most one task at a time. We assume that the program is implemented on a synchronized multiprocessor system which allocates the operations to time slots of fixed length. With such a system we can restrict ourselves to considering only tasks having unitary execution time (UET). Our goal is to map all tasks on the processors in such a way that the overall computation time, or *makespan*, is minimized and the precedence constraints are satisfied. This is a classic scheduling problem which has been intensively studied (see e.g. [3] for an updated annotated bibliography).

In real parallel architectures there is a significant *communication delay* between the time a task terminates its computation on a processor and the time the output of the task is available on another processor. If we adopt an architecture-independent model of multiprocessing (see [7]), this delay is the same for any pair of processors. In this paper we consider unitary communication delays and, as usual (see e.g. [8] and [1]), we assume that the communications can be overlapped by computation and that no delay exists for tasks assigned to the same processor. More precisely let i and j be two tasks linked by a precedence (i, j) and t be the time slot in which task i is executed. Then task j can be executed from time slot $t + 1$, if i and j are assigned to the same processor, and from time slot $t + 2$, if i and j are assigned to different processors. Using the notation introduced in [12] this problem can be denoted as $P|prec, p_j = 1, c_{jk} = 1|C_{max}$.

In [8] it has been shown that the problem is *NP*-hard, whereas in [11] it has been proved that the complexity status does not change if we impose that the precedence digraph be a tree.

In the remainder of the paper we focus our attention on precedence digraphs which take the form of an outforest, i.e. there is at most an arc (i, j) directed into any given node j . Given an instance of this problem, with n tasks and m processors, a dynamic programming algorithm exists [9] which solves $Pm|tree, p_j = 1, c_{jk} = 1|C_{max}$, in $O(n^{2m-2})$ time. In [4] Lawler has presented a linear time algorithm which provides the optimal solution for two processors and is at most $m - 2$ time units from the optimum, for $m \geq 3$. The above results have been gathered in a single paper in [10]. The worst case performance of Lawler's algorithm has been reduced to $\lceil \frac{m-2}{2} \rceil$ in [2]. Another linear time algorithm for $P2|tree, p_j = 1, c_{jk} = 1|C_{max}$ has been independently developed in [5]. In this paper we examine thoroughly the characteristics of the schedules produced by Lawler's algorithm, in order to derive a new algorithm, called *LP2-3*, which exactly solves $P2|tree, p_j = 1, c_{jk} = 1|C_{max}$ and $P3|tree, p_j = 1, c_{jk} = 1|C_{max}$, in linear time.

In the following Section 2 we introduce the notation we used, and we recall the main results from the literature. In the next Section 3 we study the properties of Lawler's heuristic algorithm. In Section 4 we describe our procedure *LP2-3* and we show how to implement it to run in linear time. The final section summarizes our work.

2. NOTATION AND PREVIOUS WORKS

We are given an outforest $T = (V, A)$, that is a directed acyclic digraph with node set $V = \{1, \dots, n\}$ and arc set A such that the indegree of each node is at most one. Two nodes i, j of V are said to be *father* and *child*, respectively, if arc $(i, j) \in A$. We will denote with $f(j)$ the unique predecessor of node j in T ($f(j)$ being undefined if node j has zero indegree). Two nodes j and k are said to be *brothers* if $f(j) \in V$ and $f(j) = f(k)$. Finally we denote with $T(i)$ the subtree of T rooted at node i .

A *schedule* of T on m identical processors is an assignment of the tasks to the processors and to a number of unitary time slots in which the tasks are executed. The schedule is feasible if no more than m tasks are assigned to the each time slot and precedence and communication delay constraints are respected. More precisely, let us define $t(i)$ as the time slot at which task $i \in V$ is executed: a feasible schedule $\mathcal{S} = (S_1, \dots, S_q)$ is a partition of node set V in q subsets such that:

- (a) $|S_t| \leq m$, for $t = 1, \dots, q$;
- (b) for each pair of tasks $i \in V, j \in V$ such that $(i, j) \in A$, either i and j are assigned to the same processor and $t(j) \geq t(i) + 1$, or i and j are assigned to different processors and $t(j) \geq t(i) + 2$.

It is not difficult to see that due to the precedence and communication delay constraints, if a node i scheduled at time $t(i)$ has more than one child, then at most one of these children can be scheduled in time slot $t(i) + 1$. Lawler [4] says that "node i has a *favoured child* when exactly one of its children is assigned to an earlier time slot than the others, if i has a child at all". Moreover he defines a schedule \mathcal{S} as having the *favoured children property* if each node has a

favorable child, and he proves the following:

THEOREM 1. (See [4].) *For any unit-delay outforest T there is an optimal and feasible schedule \mathcal{S} with the favored child property.*

This apparently simple theorem points out the real difficulty of the problem suggesting a research direction for possible solution techniques. Indeed, if we know the set F of favored children of an outforest T we can transform it into a new outforest $T_F = (V, A_F)$ in which the arc set A_F is obtained from A considering each favorite child $i \in F$, one at a time, and replacing each arc $(f(i), k) \in A$ with arc (i, k) . Lawler calls this new outforest a *delay-free* outforest and shows that:

PROPOSITION 1. (See [4].) *Let T be a unit delay outforest, and let F be any choice of favored children. Then any schedule \mathcal{S} that satisfies condition (a) above and the precedence constraints with respect to T_F , also satisfies conditions (a) and (b) for T .*

The problem defined by a delay-free outforest T_F has no constraint due to the delays and can be denoted as $P|tree, p_j = 1|C_{max}$, i.e. the problem of scheduling unitary tasks on identical processors with tree-like precedence constraints. Efficient algorithms exist (see e.g. [6]) which optimally solve this problem in linear time. It follows that, in theory, the problem with communication delays could be solved by defining the set of favored children and determining the optimal solution to the corresponding delay-free problem.

Lawler uses this approach to derive an effective approximation algorithm. He starts by defining a *shortest* delay-free outforest T_F that is a particular delay-free outforest such that the height of each subtree $T_F(i)$ of T_F is as small as possible. The *height* $h(i)$ of each shortest delay-free subtree rooted to i can be computed in linear time with the recursion:

$$h(i) = \begin{cases} 1 & \text{if } i \text{ is a leaf,} \\ \max(1 + \max\{h(j) : (i, j) \in A\}, 2 + \text{smax}\{h(j) : (i, j) \in A\}) & \text{otherwise,} \end{cases} \quad (1)$$

where $\text{smax}\{X\}$ denotes the second largest value of set X . The shortest delay-free outforest T_F is determined by choosing as favored child, among a set of brothers, the node i having maximum $h(i)$ value.

The approximation algorithm terminates by determining the optimal solution of the problem defined by the shortest delay-free outforest and taking this schedule as the solution of the original problem. Lawler has proved the following.

THEOREM 2. (See [4].) *Given an instance T of $Pm|tree, p_j = 1, c_{jk} = 1|C_{max}$ and the corresponding shortest delay-free outforest T_F , then the length of the optimal schedule of T_F is at most $m - 2$ time unit greater than the length of the optimal schedule of T .*

This implies that the algorithm provides an optimal solution for $P2|tree, p_j = 1, c_{jk} = 1|C_{max}$, and applied to $P3|tree, p_j = 1, c_{jk} = 1|C_{max}$ gives a solution whose value exceeds the optimum of at most one unit.

Lawler's algorithm has the same performances when any linear time algorithm is used for determining the optimal solution of $P|tree, p_j = 1|C_{max}$. In particular Lawler shows that the *Critical Path Scheduling* algorithm (*CP*) can be used to solve the problem. The *CP* algorithm

assigns the tasks one slot at a time, starting from slot S_1 and increasing the time index one by one. When a new slot S_t is considered, the tasks available to be scheduled in S_t are those having all predecessors assigned to prior slots. If the number of available tasks is not larger than m then all available tasks are assigned to the slot, otherwise CP selects m tasks having maximum *priority* where the priority $\nu(i)$ of a node i is the length of the longest path from i to a leaf.

OBSERVATION 1. *Given a shortest delay-free outforest T_F derived from an outforest T , then $\nu(i) \geq h(i)$, for each task $i \in V$.*

In the next Section 4 we describe an algorithm for $P3|tree, p_j = 1, c_{jk} = 1|C_{max}$ which starts with the solution of Lawler's algorithm and iteratively updates the set of favored children until a criterion shows that the current shortest delay-free outforest is optimal. If we have an instance with only two processors we can obviously apply the algorithm by returning the initial solution without updating, so the same procedure can be used to solve problems with two or three processors.

3. CHARACTERIZATION OF THE LAWLER'S SCHEDULE

In this section we study the structure of the schedule produced by Lawler's heuristic algorithm showing that, in many cases, it is optimal.

When we construct the shortest delay-free outforest T_F associated with a set of favorite children F , we introduce in T_F some arcs which do not exist in the original outforest T . In the sequel we need to distinguish between the two kinds of arcs.

DEFINITION 1. *Given a shortest delay-free outforest T_F obtained from outforest T and an arc $(h, k) \in A_F$, we call this arc true if $(h, k) \in A$, otherwise we call it false. A path is called true if it contains only true arcs, otherwise it is called false.*

The *false* arcs are introduced in T_F when a task l is chosen as favored son among its brothers. For each brother k of l we put in T_F the *false* arc (l, k) instead of arc $(f(l), k) \in A$. One can easily see that no two consecutive *false* arcs can exist and a path in T_F is a sequence of *true* paths, separated by single *false* arcs.

We call $\Pi(i, j)$, with $i \neq j$, the path of T_F between nodes i and j . When node i is a root of T_F we use the simplified notation $\Pi(j)$ to identify the unique path from the root to j .

The algorithm we are going to describe considers the schedule \mathcal{S} of a shortest delay-free outforest obtained with the critical path algorithm CP . In the remainder of the section we study this schedule and we introduce propositions which provide a detailed characterization of the structure of \mathcal{S} .

Let $\lambda(\mathcal{S})$ denote the length of the schedule, i.e. the number of slots used. The *critical slot* S_c of schedule \mathcal{S} (with $1 \leq c < \lambda(\mathcal{S})$) is the last slot with less than m tasks, except the last one. The following lemma has been proved in [2].

LEMMA 1. *(See [2].) Given a delay-free outforest and its critical path schedule \mathcal{S} , then for each task $\ell \in S_c$ which has successors scheduled after c it is $|\Pi(\ell)| = c$ (i.e. the unique path*

from a root to ℓ has exactly one task scheduled in each slot S_1, S_2, \dots, S_c).

The following new propositions will be used to prove the correctness of our algorithm.

PROPOSITION 2. *Given a shortest delay-free outforest T_F and its corresponding critical path schedule \mathcal{S} , let k be a task available to be scheduled at time instant τ , if infinite processors can be used (i.e. τ is the length of the unique path of T_F from a root to k). Then, if $t(k) > \tau$, m tasks are scheduled in each slot $S_\tau, \dots, S_{t(k)+\nu(k)-2}$.*

Proof. Algorithm *CP* delays task k available to be scheduled at a time instant t , only when m tasks with priority not less than $\nu(k)$ have been assigned to slot S_t . This proves that m tasks are scheduled in each slot $S_\tau, \dots, S_{t(k)-1}$. Each task assigned to slot $S_{t(k)-1}$ has priority not less than $\nu(k)$, but the priority of a task is the length of the longest path of T_F , from the task to a leaf, therefore m tasks are available to be scheduled in each of the $\nu(k)$ slots $S_{t(k)-1}, \dots, S_{t(k)+\nu(k)-2}$, so these slots are completely filled. ■

(A simpler version of the above proposition has also been introduced by Lawler in the proof of Theorem 2.)

PROPOSITION 3. *Given a shortest delay-free outforest T_F and its corresponding critical path schedule \mathcal{S} , let ℓ be a task scheduled in S_c , such that $|\Pi(\ell)| = c$. Then for each arc $(r, s) \in A_F$ with $r \in \Pi(\ell), s \in \Pi(\ell)$, and for each $k \in \Pi(s, \ell)$ it is*

$$h(r) \geq t(k) - t(r) + h(k), \quad \text{if } (r, s) \text{ is true;} \quad (2)$$

$$h(r) \geq t(k) - t(r) + h(k) - 1, \quad \text{if } (r, s) \text{ is false.} \quad (3)$$

Moreover if ℓ has successor in T_F it is:

$$h(r) \geq c - t(r) + 1. \quad (4)$$

Proof. We begin the proof by showing that equations (2) and (3) hold. From equation (1) we know that if arc (r, s) is *true* then $h(r) \geq 1 + h(s)$, therefore if path $\Pi(r, k)$ is *true* the difference between $h(r)$ and $h(k)$ is at least equal to the number of arcs in $\Pi(r, k)$. Since the tasks of $\Pi(\ell)$ are scheduled in contiguous slots, the number of arcs in $\Pi(r, k)$ is $t(k) - t(r)$ and equation (2) holds when $\Pi(r, k)$ is *true*. Something different happens when a *false* arc is encountered. Let (α, β) be the *false* arc closest to ℓ such that $\alpha \in \Pi(\ell)$ and $\beta \in \Pi(\ell)$ (see Figure 1, where arrows are used to identify the precedences and dotted arrows indicate *false* arcs). Tasks α and β are brothers in T , and α is the favorite child. In a shortest delay-free outforest a task is chosen as favorite child if it has height not smaller than that of its brothers, therefore $h(\alpha) \geq h(\beta)$. Since path $\Pi(\beta, k)$ is *true*, for each $k \in \Pi(\beta, \ell)$, we already know that (2) holds, so $h(\beta) \geq t(k) - t(\beta) + h(k)$, but task β is scheduled exactly one time unit later than α , thus equation (3) holds for $r \equiv \alpha$.

We have thus proved that (2) and (3) hold for each node of $\Pi(\ell)$ in the subpath which goes from the last *false* arc to ℓ . Since any path in T_F is a sequence of *true* paths, separated by

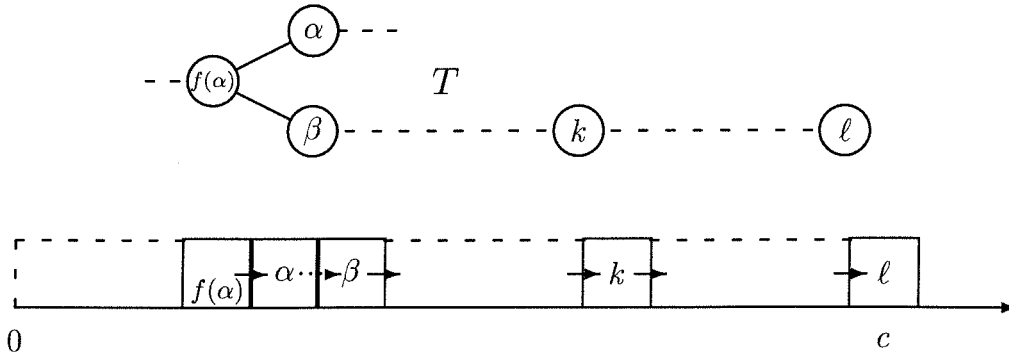


Figure 1: A *false arc* (α, β) in path $\Pi(\ell)$

single *false arcs*, to prove the thesis it is enough to show that (2) and (3) hold for the first arc preceding α and then apply the above reasoning to the remaining nodes.

Consider task $f(\alpha)$ and note that: (i) in the original outforest T task $f(\alpha)$ has at least two children: α and β ; (ii) task α is the favored child of $f(\alpha)$, so arc $(f(\alpha), \alpha)$ is *true*. We have already shown that given a task $k \in \Pi(\beta, \ell)$ the height of each of the two children α and β is not smaller than $\theta = t(k) - t(\alpha) + h(k) - 1 (= t(k) - t(\beta) + h(k))$, so $h(f(\alpha)) \geq \theta + 2$ (see equation (1)). Since the tasks of $\Pi(\ell)$ are scheduled in contiguous slots it is $t(f(\alpha)) = t(\alpha) - 1$ and one can obtain $\theta + 2 = t(k) - t(f(\alpha)) + h(k)$, so equation (2) holds for $r \equiv f(\alpha)$, which concludes the first part of the proof.

We now show that equation (4) holds. When $h(\ell) \geq 2$ equation (4) is immediately obtained by applying (2) and (3) with $k = \ell$, so we assume that $h(\ell) = 1$. In this case ℓ is a leaf of T , but it has successors in T_F , so it must be a favorite child, and $f(\ell)$ has at least two children each of them having height equal to one. Recalling again equation (1), it immediately follows $h(f(\ell)) = 3$. Given a *false arc* (r, s) in $\Pi(\ell)$, applying (3) with $k = f(\ell)$ we obtain $h(r) \geq t(f(\ell)) - t(r) + h(f(\ell)) - 1 = c - t(r) + 1$ which concludes the proof. ■

PROPOSITION 4. *Given a shortest delay-free outforest T_F and its corresponding critical path schedule \mathcal{S} , let $S \subseteq S_c$ be the set of tasks scheduled in the critical slot which have successor in T_F . Then the number of tasks in S_c is strictly greater than the number of false arcs in $\hat{\Pi} = \{\Pi(\ell) : \ell \in S\}$.*

Proof. Consider a task $\ell \in S$. From Proposition 3, it follows that for each *false arc* (α, β) such that $\alpha \in \Pi(\ell)$ and $\beta \in \Pi(\ell)$ it is $h(\alpha) \geq c - t(\alpha) + 1$. Task α is a favored child so it is the root of a subtree of T , having height $h(\alpha)$, which does not contain task β and its successors. It follows that there is a path π of T_F , with $\pi \neq \Pi(\alpha, \ell)$, which starts from α , and has at least $h(\alpha)$ nodes (see Figure 2). The priority ν of the k -th task of π is equal to $\nu(\alpha) - k + 1$, but $\nu(\alpha) \geq h(\alpha)$ (see Observation 1), so $\nu \geq c - t(\alpha) - k + 2$. Moreover one can see that this task can be scheduled from time instant $t(\alpha) + k - 1$, if enough processors are available, and it cannot be scheduled later than this instant, otherwise, according to Proposition 2, m tasks could be scheduled in S_c , so contradicting the hypothesis $|S_c| < m$. It follows that the tasks of

π are scheduled contiguously in slots $S_{t(\alpha)}, \dots, S_c, \dots, S_{t(\alpha)+|\pi|-1}$.

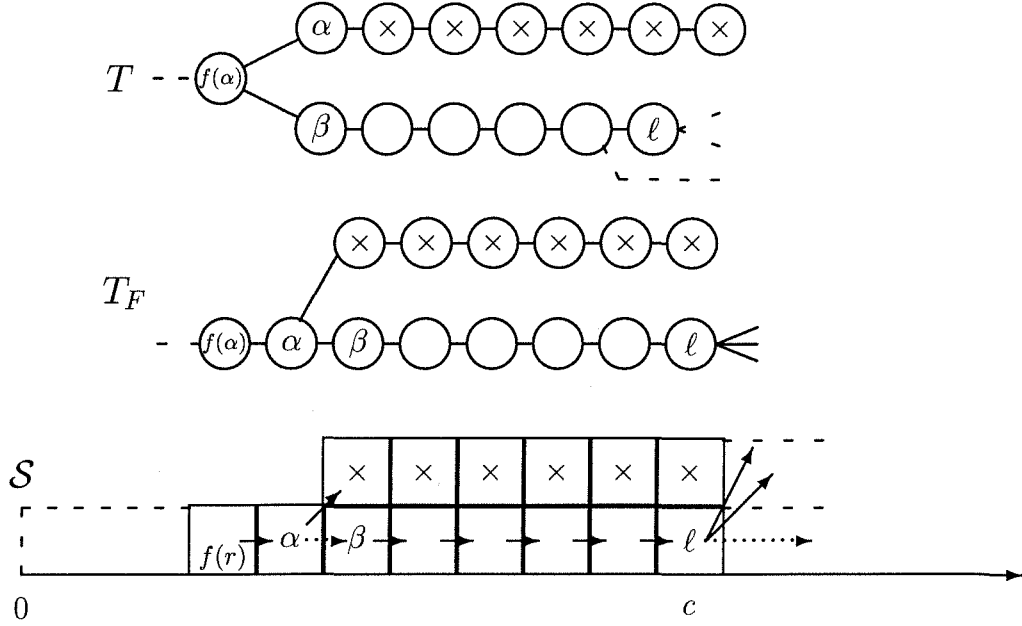


Figure 2: Outforests and schedule when task ℓ has successors in T_F

Thus for each *false* arc in $\hat{\Pi}$ there are two paths in T_F : one starting with the *false* arc, and the second one starting with a *true* arc. Both of them lead to a task scheduled in S_c . Since the structure of the precedences is an outforest the number of such tasks is at least the number of *false* arcs in $\hat{\Pi}$ plus one, and the thesis holds. ■

From the above proposition the following immediately descends.

COROLLARY 1. *Given a shortest delay-free outforest T_F and its corresponding critical path schedule, then the number of false arcs in $\hat{\Pi} = \{\Pi(\ell) : \ell \in S_c \text{ and } \ell \text{ has successors in } T_F\}$ is at most $|S_c| - 1$.*

The next two propositions give sufficient conditions for the optimality of a schedule \mathcal{S} .

PROPOSITION 5. *Given the critical path schedule \mathcal{S} of a shortest delay-free outforest T_F , let $S \subseteq S_c$ be the set of tasks scheduled in the critical slot which have successor in T_F . If path $\Pi(\ell)$ is true for each task $\ell \in S$, then the schedule is optimal.*

Proof. Given a task $\ell \in S$ observe that task $f(\ell)$ cannot be scheduled before time $c - 1$, indeed path $\Pi(\ell)$ is *true* by hypothesis, and exactly one task of this path is scheduled in each slot S_1, S_2, \dots, S_c (see Lemma 1).

For each $\ell \in S$ let $X(\ell) \subset V$ denote the set of tasks in the subtree of T_F rooted at ℓ ,

excluding task ℓ itself; define $X = \cup_{\ell \in S} X(\ell)$ and note that the definition of critical slot implies

$$\lambda(\mathcal{S}) = c + \left\lceil \frac{|X|}{m} \right\rceil. \quad (5)$$

In the original delay-free outforest T the tasks in $X(\ell)$ are brothers of ℓ , or successors of ℓ and his brothers (otherwise they could be scheduled at time c), and task ℓ is the favorite child of $f(\ell)$. We have already observed that $f(\ell)$ cannot be scheduled before time $c - 1$, so for any possible choice the favorite child must be scheduled at time c and the remaining $|X(\ell)|$ tasks from set $\{\ell\} \cup X(\ell)$ must be scheduled from time $c + 1$. Applying the same reasoning to all tasks in S it follows that for any choice of the favorite children exactly $|S|$ tasks from $S \cup X$ are scheduled at time c and $|X|$ tasks are scheduled from time $c + 1$, therefore the right-hand-side of (5) is a lower bound on the length of the optimal schedule of T and the current schedule \mathcal{S} is optimal. ■

PROPOSITION 6. *Given the critical path schedule \mathcal{S} of a shortest delay-free outforest T_F , if only one task, say i , is scheduled in the critical slot, then $\Pi(f(i))$ is a true path and the schedule is optimal.*

Proof. By definition of critical slot we know that S_c is not the last slot of the schedule, so there are tasks scheduled after time c which are successors of the unique task in S_c . From Corollary 1 it follows that $\Pi(i)$ is a true path, thus Proposition 5 shows that the schedule is optimal. ■

The propositions given hitherto apply to problems with any number of processors. In the remainder of the paper we restrict ourselves to the special case $m = 3$. Our knowledge on the schedules produced by Lawler's algorithm for $P3|tree, p_j = 1, c_{jk} = 1|C_{max}$ can be summarized as in Figure 3

At the root of the tree in the figure we have the schedule \mathcal{S} produced by Lawler's algorithm (i.e by algorithm CP when applied to a shortest delay-free outforest). Since we have exactly three processors, in the critical slot there could be either one or two tasks: cases (A) and (B). Case (A) is a leaf of our tree, indeed, according to Proposition 6, the associated schedule is optimal. Double lines are used to indicate the cases corresponding to optimal schedules.

When the critical slot S_c contains exactly two tasks, say i and j , several cases arise. By definition of S_c there must be tasks scheduled after time c and these tasks must be successors of i and/or j , in T_F (otherwise they could be scheduled in S_c). We separate the schedules in which both the tasks in the critical slot have successors in T_F (case (B.1)), from the schedules in which only one of these tasks have successors (case (B.2)). According to Corollary 1 we know that in case (B.1) at most one false arc may exist in paths $\Pi(i)$ and $\Pi(j)$, thus we further partition (B.1) in two subcases: (B.1.1) and (B.1.2). The first case corresponds to an optimal schedule, as shown by Proposition 5. In the second case, instead, the schedule may be improved, as we will show later.

When only one of the tasks in the critical slot, say i , has successors in T_F , if path $\Pi(i)$ is true, then the schedule is optimal (case (B.2.1), see Proposition 5), otherwise (path $\Pi(i)$ is

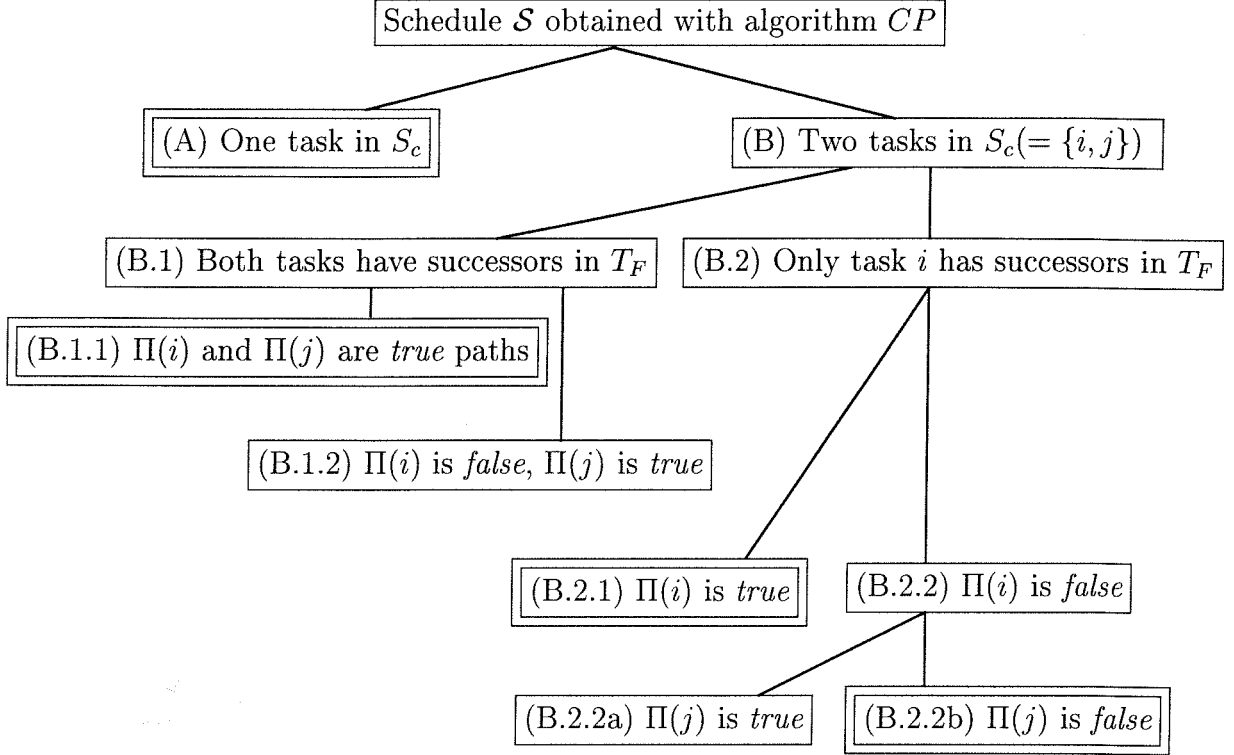


Figure 3: Case analysis of Lawler's schedule, when $m = 3$

false) we further split the case in two final subcases. The first one corresponds to schedules in which $\Pi(i)$ is *false* and $\Pi(j)$ is *true*, whereas the last one corresponds to schedules in which both paths $\Pi(i)$ and $\Pi(j)$ are *false*. If the schedule satisfies the conditions of case (B.2.2b) it is optimal, as proved by the next Proposition 7 which uses the following lemma.

LEMMA 2. *Given the critical path schedule \mathcal{S} of a shortest delay-free outforest T_F , if a task i of the critical slot S_c has successors in T_F , and $\Pi(i)$ is *false*, then the schedule could be improved only by choosing as favorite child the task β defined by the unique false arc (α, β) of $\Pi(i)$.*

Proof. We already know that schedule \mathcal{S} is optimal if $|S_c| = 1$ so we assume $S_c = \{i, j\}$. Let $(\alpha, \beta) \in A_F$ be the *false* arc closest to i , such that $\alpha \in \Pi(i)$ and $\beta \in \Pi(i)$ (see Figure 4). We know that (α, β) is the only *false* arc in $\Pi(i) \cup \Pi(j)$ (see Corollary 1), and according to the proof of Proposition 4 path $\Pi(j)$ is the union of the two subpaths $\Pi(\alpha)$ and $\Pi(\alpha, j)$. If task α is fixed as favored child of $f(\alpha)$, then path $\Pi(i)$ becomes *true* (indeed the precedence (α, β) is imposed in the solution and the corresponding arc is added to the instance T). If Task j has no successors in T_F , then Proposition 5 proves the optimality of \mathcal{S} . If, otherwise, j has successors in T_F , again using Corollary 1 we can see that $\Pi(j)$ is *true*, and Proposition 5 still proves the optimality of the schedule. If we fix a brother of α different from β as favorite child of $f(\alpha)$, the new solution cannot be improved either. Indeed the schedule of $\Pi(\beta, i)$ remains unchanged, the path $\Pi(i)$ becomes *true*, and Proposition 5 again applies. Thus the only possibility to shorten

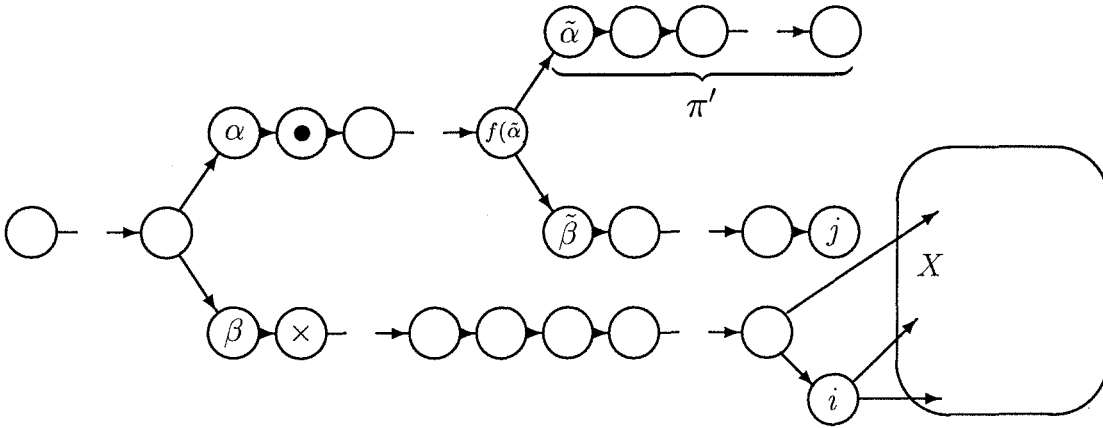


Figure 4.a: Original outforest T

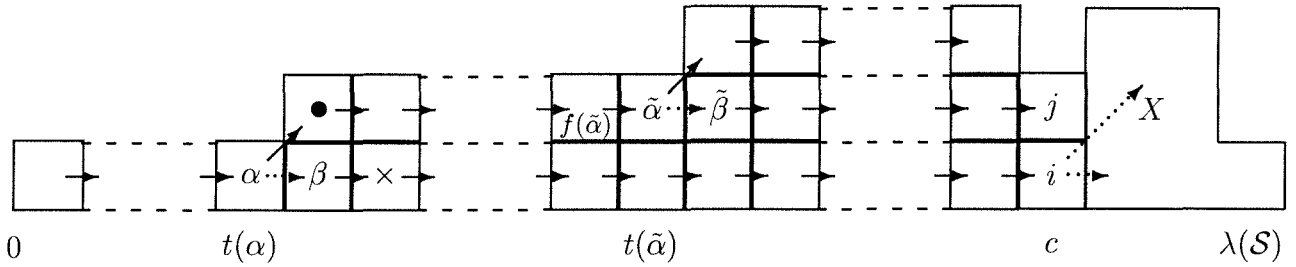


Figure 4.b: Schedule \mathcal{S}

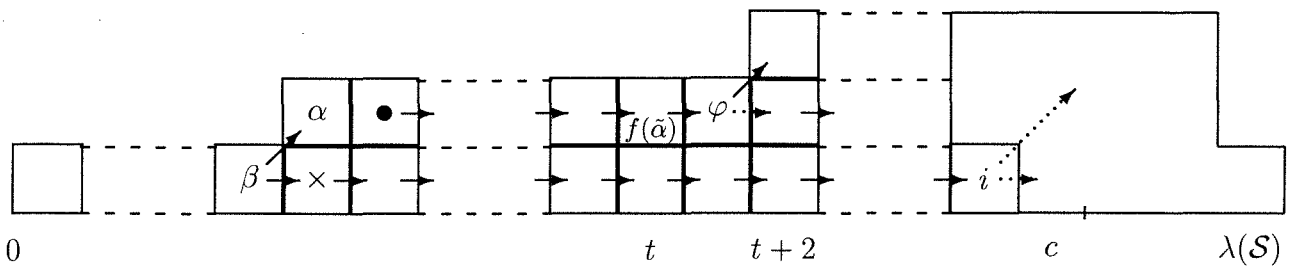


Figure 4.c: Schedule \mathcal{S}'

Figure 4: Original and improved schedule when $\Pi(i)$ is *false* and i has successors in T_F

the schedule is to choose β as favorite child. ■

PROPOSITION 7. *Given the critical path schedule \mathcal{S} of a shortest delay-free outforest T_F , if exactly two tasks, say i and j , are scheduled in the critical slot, only one has successors in T_F and both paths $\Pi(i)$ and $\Pi(j)$ are false, then the schedule is optimal.*

Proof. Let us start the proof with some more considerations on the structure of the outforest T and schedule \mathcal{S} . As in Lemma 2 let $(\alpha, \beta) \in A_F$ be the unique *false* arc such that $\alpha \in \Pi(i)$ and $\beta \in \Pi(i)$. We have already shown that j is a successor of α in T and that $h(\alpha) \geq c - t(\alpha) + 1$ (see the proof of Proposition 4), so using Proposition 2 we can see that the tasks of $\Pi(\alpha, j)$ are scheduled contiguously one in each slot $S_{t(\alpha)}, \dots, S_c$ (otherwise three tasks could be scheduled in S_c). Since $\Pi(j)$ is *false* and the only *false* arc of $\Pi(i)$ is (α, β) , then a *false* arc exists in $\Pi(\alpha, j)$. Let $(\tilde{\alpha}, \tilde{\beta})$ with $\tilde{\alpha} \in \Pi(\alpha, j)$ and $\tilde{\beta} \in \Pi(\alpha, j)$, be the *false* arc closest to α and observe that $\alpha \neq \tilde{\alpha}$ since $\Pi(\alpha, j)$ starts with a *true* arc (see Figure 4.b). Using equation (3) with $k = j$ and $r = \tilde{\alpha}$ we obtain $h(\tilde{\alpha}) \geq c - t(\tilde{\alpha})$. Similarly to what happens for task α , from task $\tilde{\alpha}$ two paths start: the first is path $\Pi(\tilde{\alpha}, j)$ which starts with the *false* arc $(\tilde{\alpha}, \tilde{\beta})$; the second, say π' , is a path of $h(\tilde{\alpha})$ tasks starting with a *true* arc. Since only tasks i and j are scheduled in S_c , then path π' cannot have more than $c - t(\tilde{\alpha})$ tasks and $h(\tilde{\alpha}) = c - t(\tilde{\alpha})$ holds. Therefore from time $t(\tilde{\alpha}) + 1$ to time $c - 1$ there are three paths scheduled: $\Pi(i)$, $\Pi(j)$ and π' . This also shows that arc $(\tilde{\alpha}, \tilde{\beta})$ is the only *false* arc in $\Pi(\alpha, j)$, otherwise by applying the same reasoning as above we would obtain that a fourth path be available to be scheduled between time $t(\tilde{\alpha}) + 1$ and time $c - 1$, but only three processors exist so more than two tasks could be scheduled in S_c .

We now have all the information on the structure of the outforest T and of the schedule \mathcal{S} , necessary to prove that \mathcal{S} is optimal. From Lemma 2 we know that the schedule could be improved only by choosing β as favored child, but we now prove that even with this choice we cannot have a solution better than the current one.

Let us define $t = t(\tilde{\alpha})$ and consider the schedule \mathcal{S}' obtained fixing β as a favorite child (see Figure 4.c). In \mathcal{S}' task α is scheduled one time unit later than in \mathcal{S} , so task $f(\tilde{\alpha})$, which has been scheduled at time $t - 1$ in \mathcal{S} , it is scheduled at time t in \mathcal{S}' (remind that $\Pi(\alpha, \tilde{\alpha})$ is *true*). We have shown above that task $f(\tilde{\alpha})$ has two brothers: $\tilde{\alpha}$ and $\tilde{\beta}$, each of which is the first of a path of $c - t$ tasks. Therefore for any choice of the favored child of $f(\tilde{\alpha})$, this child is scheduled in \mathcal{S}' at time $t + 1$ (task $\varphi \in \{\alpha, \beta\}$ in Figure 4.c) and $2(c - t) - 1$ tasks are scheduled from time $t + 2$. Path $\Pi(\beta, i)$ is scheduled in \mathcal{S}' one time unit before than in \mathcal{S} . If we call X the set of successors of i in T_F and we assume $|X| = 3p + q$ (where p and q are integers such that $p \geq 0$ and $0 \leq q \leq 2$), then we know that $c - t - 2$ tasks from $\Pi(i)$ plus $3p + q$ tasks are scheduled in \mathcal{S}' from time $t + 2$. It follows that a total of $3(c - t - 1 + p) + q$ tasks are scheduled from time $t + 2$ in \mathcal{S}' . Using all the three processors these tasks require at least $c - t + p + \lceil q \rceil - 1$ slots to be scheduled, so $\lambda(\mathcal{S}') \geq c + p + \lceil q \rceil$, but this value is equal to $\lambda(\mathcal{S})$ (see equation (5)), so schedule \mathcal{S}' cannot be shorter than \mathcal{S} and the proposition holds. ■

We now analyze in more detail the remaining cases (B.1.2) and (B.2.2a) in order to identify further characteristics which render the schedule optimal. We summarize here the more relevant

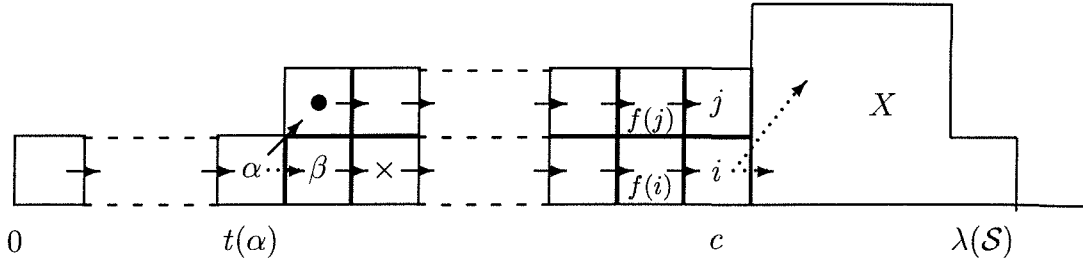


Figure 5.a: Schedule \mathcal{S}

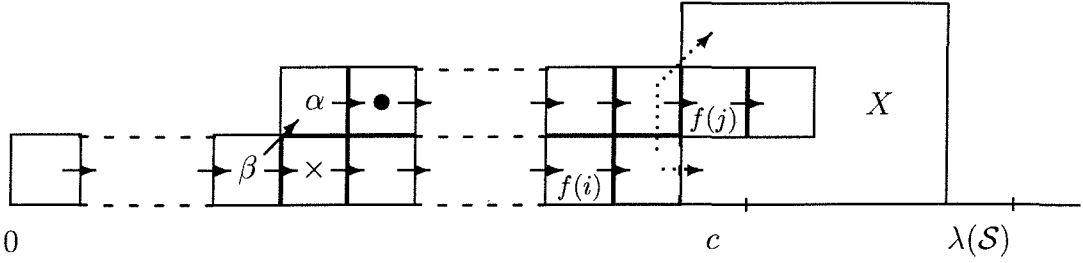


Figure 5.b: Schedule \mathcal{S}'

Figure 5: Schedules considered in the analysis of cases (B.1.2) and (B.2.2a)

properties of these cases (see also Figure 5).

- (i) The tasks of $\Pi(i)$ are scheduled one in each slot, from S_1 to S_c (see Lemma 1);
- (ii) the *false* arc (α, β) of $\Pi(i)$ is the only *false* arc in $\Pi(i)$ (see Corollary 1);
- (iii) path $\Pi(j)$ is *true* and it is the union of the two subpaths $\Pi(\alpha)$ and $\Pi(\alpha, j)$ (see the proof of Proposition 4). The tasks of $\Pi(j)$ are scheduled one in each slot from S_1 to S_c (otherwise three tasks could be scheduled in S_c , see Proposition 2);
- (iv) the schedule can be improved only by fixing task β as favorite child (see Lemma 2).

When we fix β as favorite child, then in the new schedule, say \mathcal{S}' , subpath $\Pi(\alpha, j)$ is assigned one unit later than in the current schedule \mathcal{S} , subpath $\Pi(\beta, i)$ is scheduled one unit earlier and path $\Pi(i)$ becomes *true*. So the schedule of the new paths $\Pi(i)$ and $\Pi(j)$ is fixed up to $f(i)$ and $f(j)$ (which are scheduled at times $c - 2$ and c , respectively). Tasks i and j , instead, are not necessarily scheduled immediately after $f(i)$ and $f(j)$, respectively. Indeed a *false* arc $(i, x) \in A_F$ or $(j, y) \in A_F$ could exist and tasks i and j could be scheduled later with a different choice of the favorite children. In any case, if we call X the set of task scheduled in \mathcal{S} after the critical slot and we assume $|X| = 3p + q$ (with $p \geq 0$, and $0 \leq q \leq 2$), it follows that $3p + q$ tasks from $X \cup \{i\}$ and the two tasks j and $f(j)$ will be scheduled in \mathcal{S}' , starting from time c . It follows that $\lambda(\mathcal{S}') \geq c - 1 + \lceil (3p + q + 2)/3 \rceil$, i.e. the length of schedule \mathcal{S}' is at least $c + p$ if $q \leq 1$, and at least $c + p + 1$ if $q = 2$. But from the definition of critical slot we know that $\lambda(\mathcal{S}) = c + \lceil (3p + q)/3 \rceil = c + p + \lceil q \rceil$, so we can improve the current schedule \mathcal{S} only if $q = 1$, i.e. if there is only one task scheduled in the last slot.

The above arguments also show that schedule \mathcal{S} may be improved only if it is possible to assign exactly three tasks from $X \cup \{i, j, f(j)\}$ in each slot, from time c ahead. More specifically, in the slot at time c there must be scheduled task $f(j)$ and other two tasks from $X \cup \{i\}$. If task i has at least two successors in T_F we can schedule i at time $c - 1$ and fill the slot at time c with the two successors. If otherwise task i has only one successor in T_F , we can try to fill the slot at time c choosing a brother of i as favorite child, but this new choice does not allow to schedule three tasks at time c . Indeed, let l be the single successor of i in T_F , and assume that l is a brother of i in T (otherwise it cannot be chosen as favorite child of $f(i)$). It follows that task i is a leaf of T , so $h(i) = 1$, but since i has been chosen as favorite child in the shortest delay-free outforest, it is $h(i) \geq h(l)$ (see Section 2), and $h(l) = 1$. Therefore also task l is a leaf of T and the slot at time c can never be filled with three tasks, so the current schedule cannot be shortened. We have thus proved the following.

PROPOSITION 8. *Let \mathcal{S} be the critical path schedule of a given shortest delay-free outforest T_F such that exactly two tasks, say i and j , are scheduled in the critical slot, path $\Pi(i)$ is false, and path $\Pi(j)$ is true. Then the current schedule \mathcal{S} is optimal if more than one task is scheduled in the last slot or task i has only one successor in T_F .*

We conclude the section by giving a theorem which summarizes the case analysis of Lawler's algorithm (for the special case $m = 3$), and Propositions 5–8.

THEOREM 3. *Given an instance of $P3|tree, p_j = 1, c_{jk} = 1|C_{max}$, and the critical path schedule \mathcal{S} of its shortest delay-free outforest T_F , let i be one of the tasks in the critical slot. Then \mathcal{S} is optimal if one of the following conditions is true: (i) only task i is scheduled in the critical slot; (ii) more than one task is scheduled in the last slot; (iii) task i has successors in T_F and $\Pi(i)$ is true; (iv) task i has exactly one successor in T_F and $\Pi(i)$ is false; (v) $S_c = \{i, j\}$ and both $\Pi(i)$ and $\Pi(j)$ are false.*

4. THE ALGORITHM

The results of the previous section suggest a simple algorithm for the optimal solution of $P3|tree, p_j = 1, c_{jk} = 1|C_{max}$. We start by applying Lawler's algorithm (see Section 2), i.e. we define a shortest delay-free forest T_F , breaking ties arbitrary, and we schedule T_F with the CP algorithm. We examine the resulting solution, looking for the occurrence of one of the five cases described by Theorem 3. If one of such cases occurs the current solution is optimal, otherwise the schedule satisfies the hypothesis of Lemma 2, thus we know that it could be improved only by choosing a particular task β as favorite child. Then we fix β as the favored child for the next iterations, we compute the new shortest delay-free outforest, we obtain the new current solution by applying again algorithm CP , and we iterate the procedure from the analysis of the solution. The pseudocode of the algorithm follows.

Algorithm $LP2-3(T, m, \mathcal{S}^*)$

input: T = an unit-outforest; m = the number of processors, with $m \in \{2, 3\}$;

output: \mathcal{S}^* = the optimal schedule for T ;

begin

define a shortest delay-free outforest T_F for T ; $opt := False$;

schedule T_F with algorithm CP , giving solution \mathcal{S}^* ; set $z^* := \lambda(\mathcal{S}^*)$;

if ($m = 2$ **or** an optimality criterion holds) **then** $opt := True$ **else** $\mathcal{S} := \mathcal{S}^*$;

while (not opt) **do**

identify the unique *false* arc $(\alpha, \beta) \in T_F$ such that task β must be a favored child in any outforest associated to a schedule shorter than $\lambda(\mathcal{S})$ (see Lemma 2);

$F := F \setminus \{\alpha\} \cup \{\beta\}$; mark β as a fixed favored child;

compute the new delay-free outforest T_F ; schedule the current shortest delay-free outforest T_F with algorithm CP , giving solution \mathcal{S} ;

if $\lambda(\mathcal{S}) < z^*$ **then** $z^* := \lambda(\mathcal{S})$; $\mathcal{S}^* := \mathcal{S}$;

if (an optimality criterion holds) **then** $opt := True$;

endwhile

end.

Theorem 3 is used by $LP2-3$ to check if a schedule is optimal. Moreover, when the first schedule \mathcal{S}^* has been obtained, we can prove the optimality of a different schedule \mathcal{S} , generated at an iteration of the **while** loop, comparing the lengths of these two schedules. If $\lambda(\mathcal{S}) = \lambda(\mathcal{S}^*) - 1$, then \mathcal{S} is optimal, according to Theorem 2.

The correctness of algorithm $LP2-3$ immediately descends from Theorem 3, Lemma 2 and the case analysis of the previous Section 3.

EXAMPLE. The outforest of Figure 6.a was given in [5] to show that the authors' algorithm for $P2|tree, p_j = 1, c_{jk} = 1|C_{max}$ cannot be extended to instances with $m = 3$. (As a matter of fact they give an *inforest*, but one can see that we can solve an instance described by an inforest by reverting the direction of each arc, optimally scheduling the resulting outforest, and finally reverting the schedule again (i.e. moving the tasks of slot S_k , for $k = 1, \dots, \lambda(\mathcal{S})$ to slot $S_{\lambda(\mathcal{S})-k+1}$.) The first solution (see Figure 6.b) may choose as favorite child of a task b , so obtaining a schedule with length 6. The critical slot is S_4 and case (B.2.2a) occurs. The *false* arc to be removed from the current delay-free outforest is (b, e) and task β of the pseudocode is task e . Fixing e as favorite child, we immediately obtain the optimal schedule (see Figure 6.c), having length 5.

A straightforward implementation of algorithm $LP2-3$ runs in $O(n^2)$ time. In fact, the initial shortest delay-free outforest and the first solution \mathcal{S}^* can be computed in $O(n)$, as shown by Lawler [4], and also each iteration of the **while** loop requires $O(n)$ time. Indeed: (a) the identification of the critical slot can be certainly done in $O(\lambda(\mathcal{S}))$ ($=O(n)$) time; (b) using pointers to store the arcs of the outforest $O(|A_F|)$ ($=O(n)$) time is necessary to determine if paths $\Pi(i)$ and $\Pi(j)$ are *true* or *false*; at the same time we can identify and store the possible *false* arc (α, β) which defines the task to be fixed as favored child; (c) if case (B.1.2) or (B.2.2a)

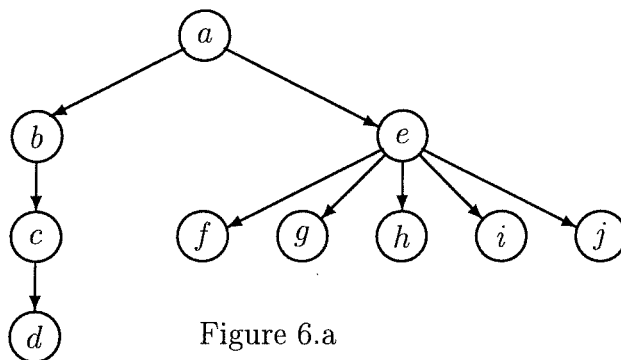


Figure 6.a

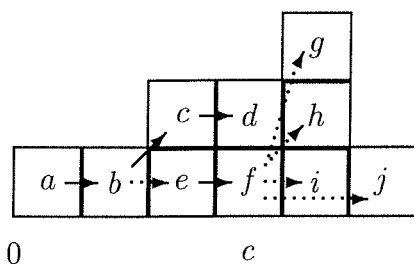


Figure 6.b

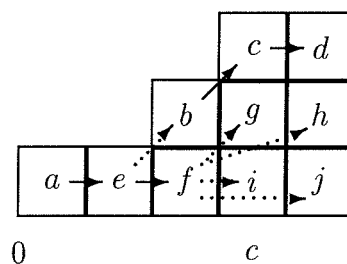


Figure 6.c

Figure 6: Instance and schedules considered in the example

occurs, then the updating of the delay-free outforest requires no more than $O(n)$ operations (the updating is $A_F = A_F \cup \{(\beta, x) : x \text{ is a brother of } \beta\} \setminus \{(f(\alpha), \alpha)\}$); and finally (d) again $O(n)$ time is required to reschedule the new outforest with algorithm *CP*.

At each iteration we fix a favored child (or we terminate), therefore the total number of iterations is bounded by the number of tasks, so the overall time bound $O(n^2)$ holds.

We now describe how to implement the algorithm so that it runs in $O(n)$ time. Our idea to reduce the computational complexity of *LP2-3* is based on the existence of an oracle which give us the index of the critical slot of the current schedule \mathcal{S} (without building the entire schedule). Moreover we use the following property.

Fact 1. When we fix task β as favored child, the new schedule \mathcal{S}' is identical to the current schedule \mathcal{S} up to time $t(\alpha) - 1$.

(One can easily see that Fact 1 holds by looking at the proof of Lemma 2.)

Using the oracle we can build a schedule \mathcal{S} only up to time instant c , identify the *false* arc (α, β) , update \mathcal{S} from time instant $t(\alpha)$ to time c , and continue to construct the new schedule \mathcal{S}' up to the new critical slot. More precisely, our implementation is as follows.

We completely build the first schedule \mathcal{S}^* and we define $z^* = \lambda(\mathcal{S}^*)$. For the second schedule \mathcal{S} we apply algorithm *CP* assigning the tasks to one slot S_t at a time, for increasing t values.

When no further task is available to be scheduled in the current slot S_t and $|S_t| < 3$ we define X as the set of unscheduled tasks and we compute the value $LB = t + \lceil \frac{|X|}{3} \rceil$, which is a lower bound on the solution value of the schedule we are building. We have three cases and, according to the analysis below, we continue to build the schedule with the same technique, or we terminate.

Case 1: $LB < z^$.* We continue to build the schedule, indeed: (i) the complete schedule has three tasks scheduled in each slot after time instant t , except the last one, so its final value is LB and the schedule is optimal (remind that the optimal solution value is equal to $z^* - 1$ or z^* , see Theorem 2); or (ii) we will encounter another slot with less than three tasks scheduled and we will again apply the same reasoning to that slot.

Case 2: $LB > z^$.* From Theorem 2 we know that the maximum reduction of length we could have for a schedule associated with the current outforest is one unit, so S^* is an optimal solution and the algorithm stops.

Case 3: $LB = z^$.* We assume the current slot to be the critical one and we adopt a decision according to the case analysis of the previous section.

Note that it is possible that if we had completed the current partial schedule we would have found that the correct critical slot is at a time instant later than t , however assuming S_t as the critical slot does not affect the correctness of our final decision. In fact, if the correct critical slot of the complete schedule \mathcal{S} is at time instant $t' > t$, either $\lambda(\mathcal{S}) > LB = z^*$ and S^* is optimal (see again Theorem 2), or $\lambda(\mathcal{S}) = LB = z^*$. In the latter case we know that at least $r = (|X| - 3(t' - t - 1) - 2)$ tasks have to be scheduled in \mathcal{S} , after time t' (i.e. the $|X|$ tasks to be scheduled after time t are assigned three at a time to each slot $S_{t+1}, \dots, S_{t'-1}$ and two to $S_{t'}$). Defining $|X| = 3p + q$, with p and q integers such that $p > 0$ and $0 \leq q \leq 2$, it follows $r = 3(p - t' + t) + 1 + q$, so

$$\lambda(\mathcal{S}) \geq t' + \left\lceil \frac{3(p - t' + t) + 1 + q}{3} \right\rceil = p + t + \left\lceil \frac{q + 1}{3} \right\rceil. \quad (6)$$

But we have $LB = t + \lceil (3p + q)/3 \rceil = p + t + \lceil q/3 \rceil$, so $\lambda(\mathcal{S}) = LB$ if and only if $q \in \{1, 2\}$ and the tasks in X are scheduled three for each slot $S_{t+1}, \dots, S_{\lambda(\mathcal{S})}$, with the exceptions of slots $S_{t'}$ and $S_{\lambda(\mathcal{S})}$. Slot $S_{t'}$ has two tasks assigned, whereas $q + 1$ tasks belong to $S_{\lambda(\mathcal{S})}$. It follows that the last slot of \mathcal{S} has at least two tasks and the schedule cannot be improved, as shown by Theorem 3, therefore schedule S^* is optimal.

The above arguments show that if $LB = z^*$ and our guess on the critical slot is not correct, then any schedule of the current outforest gives a solution not shorter than z^* , so schedule S^* remains the optimal one, independently of the choice of the critical slot.

The efficiency of the above implementation derives from the following claim.

CLAIM 1. *Implementing algorithm LP2-3 as above, the same slot is considered at most five times for assigning tasks.*

Proof of Claim 1. If only Case 1 and Case 2 occur we build the schedule considering each slot only one time. Instead, a slot must be reconsidered if Case 3 occurs and we try to improve the

current solution. In this case we identify a *false* arc (α, β) and we rebuild the schedule from slot $S_{t(\alpha)}$ to the current critical slot S_c . Let $T_{F'}$ be the new delay-free outforest and $\mathcal{S}' = S'_1, \dots, S'_c$, be the new partial schedule (with $S'_k = S_k$ for $k = 1, \dots, t(\alpha) - 1$). If it is not possible to fill the slot S'_c with three tasks we know that schedule \mathcal{S} cannot be improved and we stop; otherwise we continue to build the schedule using the above technique to identify the new critical slot, say $S'_{c'}$.

We now show that in the next iteration, if a rebuilding of the partial solution \mathcal{S}' occurs, then only slots S'_{c-1} and S'_c , among slots S'_1, \dots, S'_c , may be changed.

From the proof of Lemma 2 we know that the two paths $\Pi(i)$ and $\Pi(j)$, of $T_{F'}$, are *true*, and the tasks in S'_c are successors of task i and/or j . Therefore the first possible *false* arc to be removed from $T_{F'}$, in order to improve schedule \mathcal{S}' , is $(i, x) \in A_{F'}$, or $(j, y) \in A_{F'}$. Task i is scheduled at time $c - 1$ in \mathcal{S}' , and that task j is scheduled at time $c + 1$ (see Figure 5), so we do not need to reschedule slots S'_1, \dots, S'_{c-2} , but only $S'_{c-1}, S'_c, \dots, S'_{c'}$. We have thus proved the following.

PROPOSITION 9. *Let \mathcal{S}' be a partial schedule obtained improving a previous partial schedule \mathcal{S} having critical slot S_c . Then, when a third partial schedule, say \mathcal{S}'' is obtained improving schedule \mathcal{S}' , the earliest slot of \mathcal{S}' which may be changed is S'_{c-1} .*

Summarizing, the implementation of LP2-3 based on the construction of partial schedules defines the slots as follows. We assign tasks to slots $S_1, \dots, S_{\lambda(\mathcal{S}^*)}$, for the first time, building schedule \mathcal{S}^* . We compute for the second time slots S_1, \dots, S_c in the first iteration of the **while** loop, to obtain the first schedule \mathcal{S} . We compute again slots $S_{t(\alpha)}, \dots, S_c$ to obtain the new partial schedule \mathcal{S}' . Only the two slots S'_{c-1}, S'_c may be rescheduled to improve \mathcal{S}' (see above Proposition 9), giving the third partial schedule \mathcal{S}'' . From Proposition 9 it also follows that, if the critical slot of \mathcal{S}' is at time c' only slots $S''_{c'-1}, S''_{c'}$ may be changed to improve \mathcal{S}'' . But we know that when we improve a solution the current critical slot must be filled with three tasks, so it cannot be the critical slot of the next schedule, therefore $c' \geq c + 1$ which implies that the slot at time instant $c - 1$ is computed at most four times and the slot at time instant c at most five times as claimed.

Since no slot is computed more than five times during the entire execution of the algorithm, then the overall computing time of algorithm LP2-3 is $O(n)$.

THEOREM 4. *Algorithm LP2-3 runs in linear time.*

5. CONCLUSIONS

We have considered the problem of scheduling n unit length tasks, subject to precedence constraints and unit communication delays, on m identical processors. In particular, we have addressed the case in which the precedence graph is an outforest. We have studied the solutions obtained through an heuristic algorithm proposed by Lawler [4], giving several new properties. From these properties we have derived an algorithm for the special cases $m = 2$

and $m = 3$. A detailed case analysis has been used to prove the optimality of the algorithm and to propose an implementation which runs in $O(n)$ time, thus improving the previous best results consisting of a dynamic programming algorithm which runs in $O(n^4)$, when $m = 3$.

REFERENCES

1. P. Chrétienne and C. Picouleau, Scheduling with Communication Delays: A Survey, in “Scheduling Theory and Applications” (P. Chrétienne, E.G. Coffman Jr, J.K. Lenstra and Z. Liu, Eds), pp. 65–90, J. Wiley and Sons, Chichester, 1995.
2. F. Guignand, C. Rapine and D. Trystram, Worst Case Analysis of Lawler’s Algorithm for Scheduling Trees with Communication Delays, Technical Report APACHE 17, IMAG, Grenoble, 1996
3. J.A. Hoogeveen, J.K. Lenstra and S.L. Van de Velde, Sequencing and Scheduling, in “Annotated Bibliographies in Combinatorial Optimization” (M. Dell’Amico, F. Maffioli and S. Martello, Eds), J. Wiley and Sons, Chichester (to appear).
4. E.L. Lawler, Scheduling Trees on Multiprocessors with Unit Communication Delays, “Workshop on Models and Algorithms for Planning and Scheduling Problems”, Villa Vigoni, Como, 1993
5. J.K. Lenstra, M. Veldhorst and B. Veltman, The Complexity of Scheduling Trees with Communication Delays, *J. Algorithms* **20** (1996), 157–173.
6. C.L. Monma, Linear-Time Algorithms for Scheduling on Parallel Processors, *Oper. Res.* **30** (1992), 116–124.
7. C.H. Papadimitriou and M. Yannakakis, Toward a Architecture-Independent Analysis of Parallel Algorithms, *SIAM J. Comput.* **19** (1990), 322–328.
8. V.J. Rayward-Smith, UET Scheduling with Unit Interprocessor Communication Delays, *Discr. Appl. Math.* **18** (1987), 55–71.
9. T.A. Varvarigou, V.P. Roychowdhury and T. Kailath, Scheduling In and Out Forests in Presence of Communication Delays, in “Proc. of 7th Intern. Parallel Symposium”, pp. 222–229, 1993.
10. T.A. Varvarigou, V.P. Roychowdhury, T. Kailath and E.L. Lawler, Scheduling In and Out Forests in Presence of Communication Delays, *IEEE Trans. Parallel and Distrib. Syst.* **7** (1996), 1065–1074.
11. B. Veltman, Multiprocessor Scheduling with Communication Delays, Ph. D. thesis, CWI, Eindhoven University of Technology, 1993.
12. B. Veltman, B.J. Lageweg and J.K. Lenstra, Multiprocessor Scheduling with Communication Delays, *Parallel Comput.* **16** (1996), 173–182.

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 perspectives", 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: Pricate Vices Pubblis Virtues", pp. 20
84. Sebastiano Brusco e Sergio Paba [1991] "Connessioni, competenze e capacità concorrenziale 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 form 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
140. Mauro Dell'Amico, Martine Labbé, Francesco Maffioli [1996] "Exact solution of the SONET Ring Loading Problem", pp. 20
141. Mauro Dell'Amico, R.J.M. Vaessens [1996] "Flow and open shop scheduling on two machines with transportation times and machine-independent processing times in NP-hard, pp. 10
142. M. Dell'Amico, F. Maffioli, A. Sciomechen [1996] "A Lagrangean Heuristic for the Pirze Collecting Travelling Salesman Problem", pp. 14
143. Massimo Baldini [1996] "Inequality Decomposition by Income Source in Italy - 1987 - 1993", pp. 20
144. Graziella Bertocchi [1996] "Trade, Wages, and the Persistence of Underdevelopment" pp. 20
145. Graziella Bertocchi and Fabio Canova [1996] "Did Colonization matter for Growth? An Empirical Exploration into the Historical Causes of Africa's Underdevelopment" pp. 32
146. Paola Bertolini [1996] "La modernization de l'agriculture italienne et le cas de l'Emilie Romagne" pp. 20
147. Enrico Giovannetti [1996] "Organisation industrielle et développement local: le cas de l'agroindutrie in Emilie Romagne" pp. 18
148. Maria Elena Bontempi e Roberto Golinelli [1996] "Le determinanti del leverage delle imprese: una applicazione empirica ai settori industriali dell'economia italiana" pp. 31
149. Paola Bertolini [1996] "L'agriculture et la politique agricole italienne face aux recents scenarios", pp. 20
150. Enrico Giovannetti [1996] "Il grado di utilizzo della capacità produttiva come misura dei costi di transizione. Una rilettura di "Nature of the Firm" di R. Coase", pp. 65
151. Enrico Giovannetti [1996] "Il I° ciclo del Diploma Universitario Economia e Amministrazione delle Imprese", pp. 25
152. Paola Bertolini, Enrico Giovannetti, Giulia Santacaterina [1996] "Il Settore del Verde Pubblico. Analisi della domanda e valutazione economica dei benefici", pp. 35
153. Giovanni Solinas [1996] "Sistemi produttivi del Centro-Nord e del Mezzogiorno. L'industria delle calzature", pp. 55
154. Tindara Addabbo [1996] "Married Women's Labour Supply in Italy in a Regional Perspective", pp. 85
155. Paolo Silvestri, Giuseppe Catalano, Cristina Bevilacqua [1996] "Le tasse universitarie e gli interventi per il diritto allo studio: la prima fase di applicazione di una nuova normativa" pp. 159
156. Sebastiano Brusco, Paolo Bertossi, Margherita Russo [1996] "L'industria dei rifiuti urbani in Italia", pp. 25
157. Paolo Silvestri, Giuseppe Catalano [1996] "Le risorse del sistema universitario italiano: finanziamento e governo" pp. 400
158. Carlo Alberto Magni [1996] "Un semplice modello di opzione di differimento e di vendita in ambito discreto", pp. 10
159. Tito Pietra, Paolo Siconolfi [1996] "Fully Revealing Equilibria in Sequential Economies with Asset Markets" pp. 17
160. Tito Pietra, Paolo Siconolfi [1996] "Extrinsic Uncertainty and the Informational Role of Prices" pp. 42
161. Paolo Bertella Farnetti [1996] "Il negro e il rosso. Un precedente non esplorato dell'integrazione afroamericana negli Stati Uniti" pp. 26
162. David Lane [1996] "Is what is good for each best for all? Learning from others in the information contagion model" pp. 18

163. Antonio Ribba [1996] "A note on the equivalence of long-run and short-run identifying restrictions in cointegrated systems" pp. 10
164. Antonio Ribba [1996] "Scomposizioni permanenti-transitorie in sistemi cointegrati con una applicazione a dati italiani" pp. 23
165. Mario Forni, Sergio Paba [1996] "Economic Growth, Social Cohesion and Crime" pp. 20
166. Mario Forni, Lucrezia Reichlin [1996] "Let's get real: a factor analytical approach to disaggregated business cycle dynamics" pp. 25
167. Marcello D'Amato e Barbara Pistoresi [1996] "So many Italies: Statistical Evidence on Regional Cohesion" pp. 31
168. Elena Bonfiglioli, Paolo Bosi, Stefano Toso [1996] "L'equità del contributo straordinario per l'Europa" pp. 20
169. Graziella Bertocchi, Michael Spagat [1996] "Il ruolo dei licei e delle scuole tecnico-professionali tra progresso tecnologico, conflitto sociale e sviluppo economico" pp. 37
170. Gianna Boero, Costanza Torricelli [1997] "The Expectations Hypothesis of the Term Structure of Interest Rates: Evidence for Germany" pp. 15
171. Mario Forni, Lucrezia Reichlin [1997] "National Policies and Local Economies: Europe and the US" pp. 22
172. Carlo Alberto Magni [1997] "La trappola del Roe e la tridimensionalità del Van in un approccio sistemico", pp. 16