A Time and Resource Problem for Planning Architectures

Amedeo Cesta and Cristiano Stella

IP-CNR
National Research Council of Italy
Viale Marx 15, I-00137 Rome, Italy
amedeo@pscs2.irmkant.rm.cnr.it

Abstract. This paper concerns the problem of resource reasoning in planning. It defines formally a constraint satisfaction problem, the Time and Resource Problem ($\mathcal{TRP}$), in which resource reasoning is seen as integrated with temporal reasoning. Two propagation techniques are introduced that reason about resource constraints in the $\mathcal{TRP}$ framework. The Profile Propagation technique, similar to time-tabling techniques, considers resource utilization in single instants of time (the time values) to synthesize necessary quantitative temporal constraints. The Order Propagation technique is more original. It observes single time points (the time variables) and their orderings to synthesize necessary qualitative temporal constraints.

1 Introduction

The problem of temporal reasoning in planning has been addressed in a number of works and various techniques for temporal constraint management are now commonly used. On the contrary, the problem of resource reasoning only recently has received attention [8, 6, 9]. Such a problem is increasingly relevant in planning architectures that address realistic problems. In such problems the distinction between planning and scheduling activities becomes vague and taking into account resource constraints during the problem solving phase starts to be important also from the point of view of search control.

This paper considers the possibility of representing resource constraints in a specialized module that can be called into play by a planner to check consistency of the current resource utilization. In this work we define a constraint system for time and resources and give properties useful to propagate effects of resource constraints. In the same spirit of works on temporal constraints like [5] we consider such a module independent from a particular planner and endowed with a set of interface primitives. Similarly to [5] we define a constraint satisfaction problem that we call Time and Resource Problem ($\mathcal{TRP}$) that characterizes the whole module.

After defining the $\mathcal{TRP}$ and its computational complexity (Section 2), the problem of propagating resource constraints is addressed in (Section 3) where two different propagation techniques are defined. Section 4 compares the paper results with other approaches. Section 5 presents an experimental evaluation that shows the interest of the propagation techniques.

2 The Time and Resource Problem

This section defines a mixed constraint system with time and resource constraint representation. It also introduces a set of primitives that allows to interface the constraint systems to the external world.

2.1 Time Representation

The chosen time representation is the Simple Temporal Problem (STP) as defined in [5]. For the purpose of this paper we assume to be endowed with the ability to define incrementally: (a) a set T of temporal variables t_i named time-points by using the primitive $DefineTimeVariable(t_i)$; (b) a set of binary constraints C between time-point by using the function $AddTimeConstraint(t_i, t_j, d)$. The function $AddTimeConstraint$ takes $t_i, t_j \in T$ and $d \in \mathbf{N}$ and define the following constraint:

$$t_j - t_i \leq d \tag{1}$$

The temporal variables t_i assume values on the set $\mathbf{N}$ and such values[1] should satisfy any constraint that considers t_i. Due to the restrictions of the STP, given a set of constraints, the possible values for the time variable t_i are included in an interval $[lb_i, ub_i]$. To compute the sets $[lb_i, ub_i]$ for any t_i a directed constraint graph $G_d(V_d, E_d)$ named *distance graph* is associated to the STP, where the set of nodes V_d represents the set of variables $\{t_1, \ldots, t_n\}$ and the set of edges E_d represents the set of constraints. The presentation of the temporal propagation algorithms is out of the scope of this paper.

2.2 Resource Representation

Given a set of resources R we define for each resource $r_j \in R$ a level of availability (or resource profile) $Q_j(t)$ that represents the available resource r_j over time.

A resource constraints is defined stating that the function $Q_j(t)$ is bounded to fall in an interval $[min_j, max_j]$ [2]. The two primitives $Produce(r, q, t_i)$ and $Allocate(r, q, t_i, t_j)$ are defined to modify resource availability. The primitive $Produce(r, q, t_i)$ associates a resource production of the quantity q to a temporal variable t_i [3]. The primitive $Allocate(r, q, t_i, t_j)$ represents the use of a quantity[4] q of the resource r between the time points t_i and t_j where $t_i \leq t_j$.

[1] In the following we will use the greek letter τ to indicate time instants: possible values for temporal variables.

[2] It is to be noted that $min_j = -\infty$ means no bound to consumptions, while $max_j = \infty$ means no bound to productions.

[3] A production of $-q$ indicates a resource consumption.

[4] Also in this case q can be either positive or negative.

The primitive $Allocate(r, q, t_i, t_j)$ may be represented in terms of the $Produce$ by using the two productions $Produce(r, -q, t_i)$ and $Produce(r, q, t_j)$ and posting the constraint $t_i \leq t_j$. Such a primitive is explicitly defined because is very often used and the dependence between the two time points implied in the allocation is exploited to make propagation techniques more effective as shown later. By using such primitives it is possible to define for each resource r_j its usage $ru_{i,j}$ in the time-point t_i. A positive $ru_{i,j}$ represents a resource production, a negative $ru_{i,j}$ a resource consumption. If $ru_{i,j} = 0$ it means that the level of r_j is not modified in t_i.

When each time point t_i is assigned a value τ_i an availability level for each resource over time (named *resource profile*) may be defined as follows:

Definition 1. –Resource Profile– Let T be the set of time-points, for each resource r_j we call *resource profile* the temporal function:

$$Q_j(t) = \sum_{t_i \in T, \tau_i \leq t} ru_{i,j}$$

The function represents the available amount of the resource r_j over time. □

Definition 1 allows us to express a capacity constraint on a resource as a n-ary constraint on the time points that affect that resource. Considering that any $Q_j(t)$ is a piecewise constant function and its changes of value happen in the time instants where $ru_{i,j} \neq 0$ the constraint $min_j \leq Q_j(t) \leq max_j$ can be stated as follows:

Definition 2. –Resource Consistency– Let T_j be the set of time-points where $ru_{i,j} \neq 0$. A solution $\{t_1 = \tau_1, t_2 = \tau_2, \dots t_n = \tau_n\}$ of an STP is consistent with respect to resource constraints if and only if for each resource r_j the following property holds:

$$\forall t_i \in T_j : min_j \leq \sum_{t_h \in T_j, \tau_h \leq \tau_i} ru_{h,j} \leq max_j \tag{2}$$

□

The primitives allow to represent a class of resources that involves large part of the taxonomy defined in the KRSL language proposal [11]. In particular, we can represent resources that are consumable/producible, reusable/non-sharable, and reusable/independently sharable. This can be done with different combinations of $Produce$, $Allocate$, min, and max.

2.3 Integrating Time and Resource Constraints

The whole constraint system may be expressed defining a new constraint problem the $\mathcal{TRP}$ (Time and Resource Problem).

Definition 3. *–$\mathcal{TRP}$–* A *Time and Resource Problem* is a 7-ple

$$\mathcal{TRP} = < T, C, D, R, RU, MAX, MIN >$$

where: T is a set of time-points that may assume integer values; C is a set of temporal constraints $c_{i,j}$ of the kind $t_j - t_i \leq d_{i,j}$ with $d_{i,j} \in D$; R is a set of resources r_j which have an associated availability interval $[min_j, max_j]$ with $min_j \in MIN$ and $max_j \in MAX$; the function $RU : T \times R \to \mathbf{Z}$ represents the quantity of each resource used in any time-point. For each resource $r_j \in R$ a resource constraint is defined. A solution to a $\mathcal{TRP}$ is an assignment $\{t_1 = \tau_1, t_2 = \tau_2, \ldots\ t_n = \tau_n\}$ which is consistent with both time and resource constraints of the kind defined in equations 1 and 2. $\Box$

A $\mathcal{TRP}$ can be represented with an hypergraph that we call *Time/Resource Net*. Unfortunately adding resource constraints to an STP causes some inconvenience due to the presence of n-ary resource constraints. The following theorem holds:

Theorem 4. *$\mathcal{TRP}$ is NP-complete.*

Proof sketch. a) for each assignment $\{t_1 = \tau_1, t_2 = \tau_2, \ldots\ t_n = \tau_n\}$ it is possible to check the constraint satisfiability in polynomial time ($\mathcal{TRP} \in NP$); b) the Job-Shop Scheduling problem, which is known to be NP-complete, is polynomially reducible to $\mathcal{TRP}$. $\Box$

Known algorithms for NP-Complete problems are exponential. This involves that complete propagation algorithms in the Time/Resource Net may be impractical even on low size problems. In the rest of the paper we introduce incomplete techniques to propagate resource constraints that filter inconsistent solutions out from the search space.

3 Propagation Techniques

To propagate the resource constraints we have followed the idea of synthesizing new temporal constraints by reasoning on resource representation. This is what is usually done in scheduling to level the resource profiles (e.g., [4]). Here the attempt is to identify *necessary constraints*, that we call *implicit*, due to the verification of certain properties instead of synthesizing them heuristically as done in other approaches. This Section proposes two filtering techniques that partially propagate resource constraints: the first one reasons on single instant of time (the time values) defining a resource profile to synthesize necessary quantitative temporal constraints, the second one observes single time points (the time variables) and their orderings and synthesize necessary qualitative temporal constraints.

3.1 Profile-Propagation

The first technique requires the introduction of some preliminary concepts. The evaluation of a resource profile $Q_j(t)$ would require an assignment of a time value

τ_i to any time variable t_i that uses r_j. During the search for a solution such an assignment is not always available, as a consequence an approximate method to make deductions is needed. Our starting point is an observation also done in [6]: during problem solving at least lower and upper bounds may be given for each $Q_j(t)$. Such bounds can be called *optimistic resource profile* $orp_j(t)$ and *pessimistic resource profile* $prp_j(t)$ respectively. In particular $orp_j(t)$ $(prp_j(t))$ is a resource profile greater (lower) then any $Q_j(t)$ that can be obtained by completion of the current partial solution (that is $prp_j(t) \leq Q_j(t) \leq orp_j(t)$).

Before considering how to compute $orp_j(t)$ and $prp_j(t)$ let us give the basic properties they satisfy.

Proposition 5. *If for a resource r_j in an instant τ happens that $orp_j(\tau) < min_j$ or $prp_j(\tau) > max_j$ then the constraint system is inconsistent.*

Proposition 5 is used in [6] to check resource constraint inconsistency.

Proposition 6. *If for a given resource r_j in any instant τ happens that both $orp_j(\tau) \leq max_j$ and $prp_j(\tau) \geq min_j$ then any solution consistent with the current constraints is characterized by an admissible value for $Q_j(t)$.*

From Proposition 6 the following corollary follows:

Corollary 7. *If for any resource r_j in any instant τ happens that $orp_j(\tau) \leq max_j$ and $prp_j(\tau) \geq min_j$ then the constraint system is consistent.*

To compute the bound profiles we consider a resource r_j and the set $T_j (T_j \subseteq T)$ of the time-points in which the resource is modified. We remind that for any time point t_i the interval $[lb_i, ub_i]$ is computed and dynamically updated after any modification of the temporal network. Given any instant τ we can define two sets of time points: the set of time points $P_j(\tau)$ that are ordered before τ and the set of time point $I_j(\tau)$ about which the ordering with respect to τ is not deducible:

$$P_j(\tau) = \{t_i \in T_j \mid ub_i \leq \tau\}$$
$$I_j(\tau) = \{t_i \in T_j \mid lb_i \leq \tau \wedge ub_i > \tau\}$$

Given an assignment $\{t_1 = \tau_1, t_2 = \tau_2, \ldots\ t_n = \tau_n\}$ it should be clear that the sets $I_j(\tau)$ are empty because decisions have been taken. It is also clear that during the search process at any decision a subset of $I_j(\tau)$ is cut and inserted in $P_j(\tau)$. If we call $s_j(\tau)$ $(s_j(\tau) \subseteq I_j(\tau))$ the subset of $I_j(\tau)$ that falls in $P_j(\tau)$ in the final solution (by adding incremental choices to the current solution), we have:

$$Q_j(t) = \sum_{t_i \in P_j(t)} ru_{i,j} + \sum_{t_i \in s_j(t)} ru_{i,j}$$

that shows how the value of the resource profile is given by two terms (the first one of them known). As a consequence, to estimate the optimistic and pessimistic resource profiles we have to focus on two approximations for $s_j(t)$: $lb_j(t) \subseteq I_j(\tau)$ and $ub_j(t) \subseteq I_j(\tau)$ such that:

$$\sum_{t_i \in lb_j(t)} ru_{i,j} \leq \sum_{t_i \in s_j(t)} ru_{i,j} \leq \sum_{t_i \in ub_j(t)} ru_{i,j} \qquad (3)$$

The following subsets verify the equation 3:

$$lb^{'}_{j}(t) = \{t_i \in I_j(t) \mid ru_{i,j} < 0\}$$
$$ub^{'}_{j}(t) = \{t_i \in I_j(t) \mid ru_{i,j} > 0\}$$

With the two estimates we can compute the two bound profiles:

$$orp^{'}_{j}(t) = \sum_{t_i \in P_j(t)} ru_{i,j} + \sum_{t_i \in ub^{'}_{j}(t)} ru_{i,j}$$
$$prp^{'}_{j}(t) = \sum_{t_i \in P_j(t)} ru_{i,j} + \sum_{t_i \in lb^{'}_{j}(t)} ru_{i,j}$$

The computation of $orp^{'}_{j}(t)$ and $prp^{'}_{j}(t)$ and the verification of Proposition 5 and Corollary 7 allow to check for consistency of the current $\mathcal{TRP}$ network. This computation is equivalent to what is proposed in [6]. In fact it is possible to observe that the optimistic estimate consists in considering the consumption of resources as late as possible and the productions as early as possible. The pessimistic estimate can be obtained by reasoning the opposite way.

A more accurate estimate with respect to [6] may be obtained considering the duration constraints between the time points involved in an *Allocate* primitive. Without loss of generality, we assume that a single time-point can not be involved in more than one *Allocate*. With this assumption, we define a function $Link(t_i)$ such that $Link(t_i) = t_j$ if and only if either $Allocate(r, q, t_i, t_j)$ or $Allocate(r, q, t_j, t_i)$ have been performed ($Link(t_i) = \perp$ if $t_i$is not involved in any allocation). From the definition of *Allocate* it follows that $Link(t_i) \leq t_i$ for allocation of the kind $Allocate(r, q, t_j, t_i)$ and $Link(t_i) \geq t_i$ for $Allocate(r, q, t_i, t_j)$. It may be shown that when an optimistic and pessimistic estimate involve a time-point affected by an allocation a more accurate version of the estimate is the following (the subsets verify the equation 3):

$$lb_j(t) = \{t_i \in lb^{'}_{j}(t) \mid \neg(Link(t_i) \in I_j(t) \wedge Link(t_i) \leq t_i)\}$$
$$ub_j(t) = \{t_i \in ub^{'}_{j}(t) \mid \neg(Link(t_i) \in I_j(t) \wedge Link(t_i) \leq t_i)\}$$

With this new estimates of $s_j(t)$ it is possible to compute new bounds $orp_j(t)$ and $prp_j(t)$ more accurate of the previous ones.

Instead of stopping ourselves to this control for resource consistency, we use the bounds to synthesize new temporal constraints to be added in the STP problem. Those constraints are implied by the current situation (indicated with the operator $\Rightarrow$ in what follows). We can justify this technique with the following Theorem:

Theorem 8. *Let us consider a* Time/Resource Net *in which for any resource* r_j *the optimistic* $orp_j(t)$ *and pessimistic* $prp_j(t)$ *profiles have been defined. For any resource* r_j*, any instant* τ*, and any time point* $t_i \in I_j(\tau)$ *the following properties hold:*

$$\begin{array}{l} (t_i \in ub_j(\tau)) \wedge (orp_j(\tau) - ru_{i,j} < min_j) \Rightarrow t_i \leq \tau \\ (t_i \in negset_j(\tau)) \wedge (orp_j(\tau) + ru_{i,j} < min_j) \Rightarrow t_i > \tau \\ (t_i \in posset_j(\tau)) \wedge (prp_j(\tau) + ru_{i,j} > max_j) \Rightarrow t_i > \tau \\ (t_i \in lb_j(\tau)) \wedge (prp_j(\tau) - ru_{i,j} > max_j) \Rightarrow t_i \leq \tau \end{array} \quad (4)$$

where:

$$negset_j(t) = \{t_i \in lb_j^{'}(t) \mid \neg(Link(t_i) \in I_j(t) \wedge Link(t_i) \geq t_i)\}$$
$$posset_j(t) = \{t_i \in ub_j^{'}(t) \mid \neg(Link(t_i) \in I_j(t) \wedge Link(t_i) \geq t_i)\}$$

We call the technique derived from Theorem 8 *Profile Propagation* (PP). It consists of detecting situations in which some of the conditions 4 of the Theorem hold in order to deduce the new quantitative temporal constraints to cut out inconsistent search space. An incremental algorithm to apply these filtering techniques has been implemented and has a complexity $O(n^2 m)$ where n is the number of time-points and m is the number of resources. It is worth noting that Profile Propagation needs only the temporal information given by the admissible intervals $[lb_i, ub_i]$ for any time point t_i. As a consequence it can be used with a temporal algorithm that incrementally computes arc-B-consistency on the temporal network.

3.2 Order-Propagation

The Profile Propagation is the adaptation to the $\mathcal{TRP}$ of a class of techniques also known as "time-tabling" [10]. The presented formalization allowed us to define a second propagation technique that turns out to be more effective of the previous one but also more demanding in term of computational resources. The rationale for the new technique, named *Order Propagation* (OP) stands in observing the core of the Profile Propagation. In PP we observe the generic instant τ and the relative position with respect to τ of the time points that affect a certain resource. If we compute the path-consistency of the temporal network we know the relative position of any time-point with respect to the others because we compute the minimal distances on the graph G_d between any couple of time point. As a consequence, we can adapt the same observations done for the time instants to the time variables t_i and may repeat the same reasoning with respect to the t_i instead of the τ. We call d_{ij} the minimal distance on G_d (the shortest path as shown in [5]) from the time point t_i to the time point t_j (d_{ij} is the minimal integer l such that $t_j - t_i \leq l$).

For each resource r_j we identify the set $T_j (T_j \subseteq T)$ of the time-points in which the resource is used. Similarly to the previous discussion we define the set of time points $P_j(t_i)$ that are ordered before t_i and the set of time point $I_j(t_i)$ whose ordering with respect to t_i is not deducible. Observing that if $d_{ij} \leq 0$ it means that $t_j \leq t_i$ and that if $d_{ji} < 0$ it means that $t_j > t_i$, we may define the two sets as follows:

$$P_j(t_i) = \{t_h \in T_j \mid d_{ih} \leq 0\}$$
$$I_j(t_i) = \{t_h \in T_j \mid d_{ih} > 0 \wedge d_{hi} \geq 0\}$$

Indicating with $s_j(t_i)$ the subset of $I_j(t_i)$ that falls in $P_j(t_i)$ in the final solution, in which all the time points in T_j are ordered among them, we can again identify

$$Q_j(t_i) = \sum_{t_h \in P_j(t_i)} ru_{h,j} + \sum_{t_h \in s_j(t_i)} ru_{h,j}$$

In a way analogous to what done in the previous paragraph we can estimate an optimistic and pessimistic resource profile using the formulas:

$$
\begin{array}{l}
LB^{'}_j(t_i) = \{t_h \in I_j(t_i) \mid ru_{h,j} < 0\} \\
UB^{'}_j(t_i) = \{t_h \in I_j(t_i) \mid ru_{h,j} > 0\} \\
LB_j(t_i) = \{t_h \in LB^{'}_j(t_i) \mid \neg(Link(t_h) \in I_j(t_i) \wedge Link(t_h) \leq t_h)\} \\
UB_j(t_i) = \{t_h \in UB^{'}_j(t_i) \mid \neg(Link(t_h) \in I_j(t_i) \wedge Link(t_h) \leq t_h)\} \\
ORP_j(t_i) = \sum_{t_h \in P_j(t_i)} ru_{h,j} + \sum_{t_h \in UB_j(t_i)} ru_{h,j} \\
PRP_j(t_i) = \sum_{t_h \in P_j(t_i)} ru_{h,j} + \sum_{t_h \in LB_j(t_i)} ru_{h,j}
\end{array}
$$

The $ORP_j(t_i)$ and $PRP_j(t_i)$ satisfy also a Proposition and Corollary completely analogous to the one given above in which the generic instant τ is substituted by the generic t_i.

Also in this case we have used the properties for a propagation technique that is justified by the following theorem:

Theorem 9. *Let us consider a* Time/Resource Net *in which for any resource r_j the optimistic $ORP_j(t_i)$ and pessimistic $PRP_j(t_i)$ profiles have been defined. For any resource r_j, any $t_i \in T_j$, and any time point $t_h \in I_j(t_i)$ the following properties hold:*

$$
\begin{array}{l}
(t_h \in UB_j(t_i)) \wedge (ORP_j(t_i) - ru_{h,j} < min_j) \Rightarrow t_h \leq t_i \\
(t_h \in NEGSET_j(t_i)) \wedge (ORP_j(t_i) + ru_{h,j} < min_j) \Rightarrow t_h > t_i \\
(t_h \in POSSET_j(t_i) \wedge (PRP_j(t_i) + ru_{h,j} > max_j) \Rightarrow t_h > t_i \\
(t_h \in lb_j(t_i)) \wedge (PRP_j(t_i) - ru_{h,j} > max_j) \Rightarrow t_h \leq t_i
\end{array} \tag{5}
$$

where:

$$
\begin{array}{l}
NEGSET_j(t_i) = \{t_h \in LB^{'}_j(t_i) \mid \neg(Link(t_h) \in I_j(t_i) \wedge Link(t_h) \geq t_h)\} \\
POSSET_j(t_i) = \{t_h \in UB^{'}_j(t_i) \mid \neg(Link(t_h) \in I_j(t_i) \wedge Link(t_h) \geq t_h)\}
\end{array}
$$

We call the technique derived from Theorem 9 *Order-Propagation* (OP). It consists of detecting situations in which some of the four conditions of the Theorem hold in order to deduce the new qualitative temporal constraints to cut out inconsistent search space. An incremental algorithm to apply this filtering techniques has been implemented and has a complexity $O(n^2m)$ where n is the number of time-points and m is the number of resources. It is worth noting that Order-Propagation is inevitably more demanding from the computational side with respect to Profile-Propagation because it requires a more detailed temporal network. It needs the computation of the minimal distances between any couple of time points. As well known, arc-B-consistency on the temporal network is not sufficient to compute such minimal distances but path-consistency is needed. To this purpose we have adapted dynamic shortest path algorithms described in [2] to the temporal case.

4 Related Works

Resource reasoning has also been addressed in previous research on planning. The SIPE planner [15] is able to deal with consumable and producible resources but it does not allow an explicit representation of constraints to avoid conflicts or control the search space.

The HSTS architecture [8] addresses the problem of managing sharable resources, however it forces an over-constraining of the partial plan, since the sharable resources are managed specifying a total ordering of the activity within the plan.

O-Plan [6] addresses the managing of sharable resources using a criteria based on the optimistic and pessimistic resource profile in order to identify bounds violations. We started from the same basic observation concerning the bounds but investigated the synthesis of necessary new constraints that avoid bounds violations.

A different approach has been recently used in the IxTeT planner and described in [9]. In such system the resource management is entirely performed at search level. In particular, this technique consists of recognizing a potential conflict on the resource utilization and selecting in a heuristic way a further constraint able to avoid the conflict. It is worth noting that our techniques are somehow complementary to the one described in [9] and it would be interesting to use both the approaches in the same planner.

In [12] the problem MCJSSP is defined (Multiple-Capacitated Job-Shop Scheduling Problem), that is a generalization of the JSSP (Job-Shop Scheduling Problem) for resources with capacity greater than one. The paper presents techniques which currently represent the most efficient approach to address complex resource allocation problems through constraint-based methods. In such work constraint propagation is performed by two distinct techniques: the former is very similar to the Profile-Propagation, while the latter, which is called Sequencing-Checking, and can be considered an extension of the edge-finding technique [1] defined for the JSSP problem, is based on the analysis of the conflicts that arise between sets of operations. A theoretical comparison of sequencing-checking with our approach has not been done yet but an experimental comparison has been performed and it is described in Section 5.

4.1 *parc*Plan

The work reported in this paper is being developed from quite a while and previously described only in our native Italian language [14, 3]. In the meantime, an independent research has been published [7] which also uses orderings between time-points. Because of the similarities between the latter approach and ours we draw here a comparison.

In [7] a propagation technique for resource constraints is presented and applied to the *parc*Plan planner. The authors do not use time-tabling techniques like our Profile-Propagation. They start from a formalization, different from

$\mathcal{TRP}$, that considers only the *Allocate* primitive and propose a technique that can be compared with our Order-Propagation.

It is possible to prove that the deductions the *parc*Plan method allows to obtain are a proper subset of the deductions obtained by the Order-Propagation. Adapting the *parc*Plan propagation mechanism to the $\mathcal{TRP}$ it should assume the following form:

$$\begin{array}{l} (t_h \in UB_j(t_i)) \wedge (ORP_j(t_i) = min_j) \Rightarrow t_h \leq t_i \\ (t_h \in NEGSET_j(t_i)) \wedge (ORP_j(t_i) = min_j) \Rightarrow t_h > t_i \end{array} \tag{6}$$

It is possible to verify how both the relations use a stronger condition with respect to the corresponding relations (first two relations) in equation 5 of theorem 9. In fact, from the first relation in equation 5 we can notice that because $t_h \in UB_j$, from the definition of UB_j it follows that $ru_{h,j} > 0$. As a consequence, it is easy to deduce that the first relation in equation 6 is more restrictive than its correspondent in equation 5. A similar proof holds for the corresponding second relations. Furthermore, *parc*Plan does not use the deductions performed by the third and fourth relation of equation 5.

5 Experimental Evaluation

The performance of the approach implemented in the Time/Resource Net needs a deep evaluation on realistic problems. Here we describe some test performed in an experimental setting that although artificial gives some first reinforce in the results.

5.1 Setting

We have built an experimental apparatus creating a set of MCJSSP problems manipulating the benchmarks for JSSP proposed in [13]. The chosen problem setting requires less expressive power with respect to the one allowed by $\mathcal{TRP}$. In fact MCJSSP represents only resource allocations, contains operations of fixed duration, and does not specify metric separation constraints between operations.

To simulate different styles of problem solving we have used two different search algorithms:

– *Instant-Search*: This is an algorithm with backtracking which selects, at each step, one of the un-scheduled operations and assigns to it the earlier possible start time. Such an algorithm is similar to the one used in [12] and is specialized for scheduling problems. Instant-Search is not a least-commitment algorithm, therefore it is not suitable to simulate the behavior of the most recent planning architectures. For this reason we have designed the second algorithm.
– *Order-Search*: This is a least-commitment search algorithm that instead of assigning time values to the temporal variables, posts ordering relationship between the time points which use the same resource. More specifically, at

each step an ordering is selected, among those not yet posted, that implies the least commitment for the solution.

Each experimental test consists in resolving a set of 250 MCJSSP problems using the two search algorithms defined with four different kinds of constraint propagation techniques:

– *Profile-Check*: it is essentially the technique used by the O-Plan planner [6]. It is not an actual propagation technique but a check for optimistic and pessimistic bound violation.
– *Profile-Propagation*: it is the technique defined in Subsection 3.1.
– *Sequencing-Checking plus Profile-Propagation*: it coincides with the constraint propagation performed by [12].
– *Profile-Propagation plus Order-Propagation*: it consists of the mixed execution of the two propagations techniques proposed here.

The first three propagation techniques coupled with the Instant-Search algorithm do not need the array of the minimal distances between each pair of time points, so they are associated with a temporal propagation that just achieves arc-consistency on the temporal network. Instead, in all other cases an algorithm that achieve path-consistency on the temporal network is used. In fact the Order-Search solver needs the minimal distance between each pair of time points.

The problems to solve are split into 5 classes of growing size. Each class is identified by 5 resources with capacity 2 (as in [12]) and a number of jobs respectively equal to 12,16,20,24 and 30. In this way we have obtained a total number of operations (of capacity 1) per problem respectively equal to 60,80,100,120 and 150. For each class 50 problems have been randomly generated using a technique similar to the one used in [12]. The module code has been implemented in Allegro Common Lisp on a Sun SPARC10 workstation.

5.2 Results

Table 5.2 shows, for each class (number of operations indicated with O60, O80, O100, O120 and O150) and for each kind of propagation, and for each search algorithm the number of solved problems (out of the 50 generated for each class) stopping the search after a fixed amount of time growing according to the class size and equal to, respectively, 100,200,250,300 and 400 seconds of CPU time.

Analyzing the table some observation can be done.

The Profile-Checking approach seems to have a limited effectiveness with problems of increasing size. It has a very low cost so in general it allows to visit more deeply the search space in the same time interval at its disposal.

The Sequencing-Checking is confirmed to perform quite well when couple with the Instant-Search algorithm as done in [12]. But its effectiveness is surprisingly limited when coupled with a least-commitment strategy like Order-Search.

The Profile-Propagation and Order-Propagation techniques, instead seem to be the most effective with this second kind of solver, therefore its reasonable to

Table 1. Experimental results: number of solved problems for each propagation technique

	Instant Search					Order Search				
	O60	O80	O100	O120	O150	OA60	O80	O100	O120	O150
Profile-Checking	48	25	0	0	0	20	23	15	13	4
Profile-Propagation	39	15	15	12	4	24	26	18	13	5
Sequencing-Checking + Profile-Propagation	50	45	42	40	35	20	26	9	4	0
Order-Propagation + Profile-Propagation	49	37	25	15	10	43	46	42	48	41

suppose that they should constitute an effective support for a planning architecture that adopts a least-commitment strategy.

Furthermore, the cross analysis between the two halves of the table shows that, on the problem set, the approach consisting of the Time/Resource module plus the least-commitment solver is as much efficient as the approach proposed in [12].

Of course more experimentation is needed to confirm the trends shown here. In particular experiments with a real planning architecture would be very helpful to confirm the behavior shown with least-commitment strategies.

6 Conclusions

This paper has defined a mixed time and resource problem $\mathcal{TRP}$ that formalizes in CSP terms the representation of resource constraints for planning systems. Furthermore a Time/Resource Net has been introduced that can be a powerful module to be inserted in a planning architecture to support search control in expressive domains. Two propagation techniques have been defined, Profile-Propagation and Order-Propagation, that allow to synthesize new implicit but necessary temporal constraints after reasoning on resource profiles. The effectiveness of the propagation techniques has been compared with other state-of-the-art techniques and shown to be quite competitive.

Acknowledgments

Authors' work was supported by Italian Space Agency, CNR Committee 12 on Information Technology (Project SCI*SIA), and CNR Committee 04 on Biology and Medicine.

References

1. D. Applegate and W. Cook. A Computational Study of the Job-Shop Scheduling Problem. *ORSA Journal on Computing*, 3:149–156, 1991.
2. G. Ausiello, G.F. Italiano, A. Marchetti Spaccamela, and U. Nanni. Incremental Algorithms for Minimal Length Paths. *Journal of Algorithms*, 12:615–638, 1991.
3. A. Cesta and C. Stella. Rappresentazione e gestione di vincoli su risorse in una architettura per la pianificazione (in Italian). In *Proceedings of the Fifth AI*IA Symposium. Naples, Italy*, 1996.
4. C. Cheng and S.F. Smith. Generating Feasible Schedules under Complex Metric Constraints. In *Proceedings 12th National Conference on AI (AAAI-94)*, 1994.
5. R. Dechter, I. Meiri, and J. Pearl. Temporal Constraint Networks. *Artificial Intelligence*, 49:61–95, 1991.
6. B. Drabble and A. Tate. The Use of Optimistic and Pessimistic Resource Profiles to Inform Search in an Activity Based Planner. In *Proceedings of the 2nd Int. Conf. on Artificial Intelligence Planning Systems (AIPS-94)*, 1994.
7. A. El-Kholy and B. Richards. Temporal and Resource Reasoning in Planning: the *parc*PLAN Approach. In *Proceedings of the 12th European Conference on Artificial Intelligence (ECAI-96)*, 1996.
8. R.E. Frederking and N. Muscettola. Temporal Planning for Transportation Planning and Scheduling. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1992.
9. P. Laborie and M. Ghallab. Planning with Sharable Resource Constraints. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-95)*, 1995.
10. C. Le Pape. Three Mechanisms for Managing Resources Constraints in a Library for Constraint-Based Scheduling. In *Proceedings of the INRIA/IEEE Conference on Emerging Technologies and Factory Automation*, 1995. `http://www.ilog.fr`.
11. N. Lehrer. KRSL Reference Manual 2.0.2. ARPA/Rome Laboratory Planning and Scheduling Initiative. Technical report, ISX Corporation, 1993.
12. W.P.M. Nuijten and E.H.L. Aarts. Constraint Satisfaction for Multiple Capacitated Job-Shop Scheduling. In *Proceedings of the 11th European Conference on Artificial Intelligence(ECAI-94)*, 1994.
13. N. Sadeh. *Look-ahead Techniques for Micro-opportunistic Job-shop Scheduling.* PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 1991.
14. C. Stella. Trattazione di vincoli su tempo e risorse in una architettura per la pianificazione automatica (in Italian). Master's thesis, Department of Computer and System Science, University of Rome "La Sapienza", February 1996.
15. D.E. Wilkins. *Practical Planning: Extending the Classical AI Planning Paradigm.* Morgan Kaufmann Pub.Inc., 1988.