Filtering Algorithms for Discrete Cumulative Problems with Overloads of Resource

Alexis De Clercq, Thierry Petit, Nicolas Beldiceanu, and Narendra Jussien

École des Mines de Nantes, LINA UMR CNRS 6241, 4, rue Alfred Kastler,
FR-44307 Nantes, France
{ADeClerc,TPetit,NBeldice,NJussien}@mines-nantes.fr

Abstract. Many cumulative problems are such that the horizon is fixed and cannot be delayed. In this situation, it often occurs that all the activities cannot be scheduled without exceeding the capacity at some points in time. Moreover, this capacity is not necessarily always the same during the scheduling period. This article introduces a new constraint for solving this class of problems. We adapt two filtering algorithms to our context: Sweep and P. Vilím's Edge-Finding algorithm. We emphasize that in some problems violations are imposed. We design a new filtering procedure specific to this kind of events. We introduce a search heuristic specific to our constraint. We successfully experiment our constraint.

1 Introduction

Scheduling problems consist of ordering activities. In *cumulative scheduling*, each activity has a duration and requires for its execution the availability of a certain amount of a renewable resource, its *consumption* (or *capacity demand*). Usually the objective is to minimize the *horizon* (maximum due date of an activity in the schedule), while at any point in time the cumulated consumption of activities should not exceed a limit on the available resource, the *capacity*.

However, many industrial problems require of their activities to be scheduled within a given time window, that is, the horizon is fixed and cannot be delayed. In this situation, it may occur that all the activities cannot be scheduled without exceeding the capacity at some points in time. To obtain a solution, exceeds can be tolerated under a certain limit provided operational rules that guarantee the practical feasibility are satisfied. Furthermore, in some problems the time window is partitioned: the capacity is not the same for each interval of the partition. Figure 1 depicts such a situation with overloads.

We introduce a new constraint, SoftCumulative, which extends the work presented in [9] to consider overloaded cumulative problems where different local capacities are defined for disjoint intervals of time. Several violation measures can be used for quantifying exceeds of local capacities as well as for computing the global objective value: one may wish to minimize the highest overload, or to minimize the sum of overloaded areas.

J. Lee (Ed.): CP 2011, LNCS 6876, pp. 240–255, 2011.

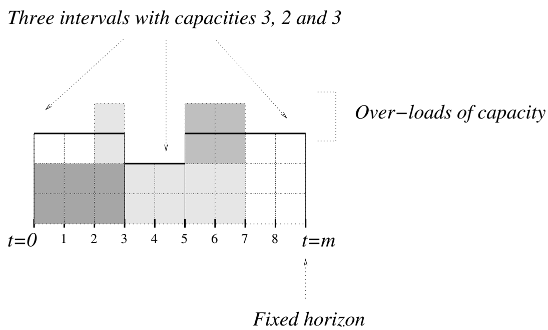

Fig. 1. A cumulative problem with a fixed horizon ($m = 9$) and 3 intervals with capacities respectively equal to 3, 2 and 3. Each activity requires 2 units of resource. The first one starts at $t = 0$ and ends at $t = 3$, the second one starts at $t = 2$ and ends at $t = 7$, the third one starts at $t = 5$ and ends at $t = 7$. There are two overloads of capacity: one in the first interval at time 2, one in the third interval at times 5 and 6.

We discuss concrete problems that can be encoded using SoftCumulative together with additional constraints. Some of these constraints may lead to situations where overloads are imposed at some intervals in time. The need of this propagation for imposed violations emphasizes that, for solving over-constrained applications, it is not always sufficient to consider only maximum violation values.

Our main contribution is a new filtering algorithm associated with SoftCumulative, which also considers imposed overloads, decomposed into three phases. The first phase is an adaptation of the $O(n \cdot log(n))$ sweep algorithm for Cumulative [4], where n is the number of activities. In our case, the time complexity also depends on the number p of user-defined local capacities: $O((n+p) \cdot log(n+p))$. The advantages of sweep are preserved: the profile is computed and the starting time variables of activities are pruned in one sweep, and the complexity does not depend on the number of points in time. To perform an energetic reasoning[1] complementary to the profile-based reasoning of sweep, in a second phase we adapt the $O(k \cdot n \cdot log(n))$ Edge-Finding algorithm of P. Vilím [13], where k is the number of distinct capacity demands. In our case, time complexity is $O(p \cdot k \cdot n \cdot log(n))$. In the two phases lower bounds of the objective are included into pruning conditions without increasing the complexity. The third phase is a new specific propagation for events on minimum values of variables which express violations of local capacities, without increasing the time complexity.

Section 2 presents the background for understanding our contributions and defines SoftCumulative. Section 3 presents some motivating examples. Section 4 describes the filtering algorithm of SoftCumulative. Section 5 presents a new dedicated search strategy and the experiments we performed using choco [1].

[1] Deductions based on the comparison between the consumed and available resource.