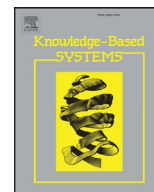




Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Hierarchical task network planning with resources and temporal constraints

Chao Qi^{a,b,c,*}, Dan Wang^d, Héctor Muñoz-Avila^e, Peng Zhao^{b,c}, Hongwei Wang^{a,b,c}

^aSchool of Management, Huazhong University of Science and Technology, Wuhan 430074, China

^bSchool of Automation, Huazhong University of Science and Technology, Wuhan 430074, China

^cKey Laboratory of Education Ministry of Image Processing and Intelligent Control, Huazhong University of Science and Technology, Wuhan 430074, China

^dThe David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, N2L3G1 ON, Canada

^eComputer Science and Engineering, Lehigh University, 19 Memorial Drive West, Bethlehem, PA 18015, USA

ARTICLE INFO

Article history:

Received 23 September 2015

Revised 22 June 2017

Accepted 26 June 2017

Available online xxx

Keywords:

Task planning

Hierarchical task networks

Multi-capacity discrete resource

Temporal constraint

ABSTRACT

Planning problems in many real-world areas are characterized by the involvement of various types of resources and complex temporal and functional relationships among numerous tasks. Hierarchical Task Network (HTN) planning is suitable for large-scale practical planning problems due to its hierarchical task decomposition principle and expressiveness for domain knowledge representation. In this paper, we propose an HTN planning algorithm named GSCCB-SHOP2 to handle multi-capacity discrete resources and complex temporal constraints simultaneously during planning. The algorithm integrates three carefully designed and interrelated sub-modules. First, the Resource model realizes resource reasoning with the designed state updating rules. Second, the Check Consistency and Backtrack (CCB) module is designed to determine temporal constraints and maintain the consistency of those constraints. Third, the Guide Search (GS) module is designed to improve the resource utilization and thus shorten the makespan performance of the generated action plan. Experimental studies are conducted to verify the efficiency of the proposed algorithm.

© 2017 Elsevier B.V. All rights reserved.

1. Background

Making plans to complete a set of prescribed tasks or goals is an essential component of routine decisions in various areas, especially for cases with a large variety of complex procedures, such as logistics, manufacturing, mission planning, and emergency response [1]. The generation of action plans in these areas is crucial and challenging due to the features of abstract tasks and goals, complex domain knowledge, limited resources, and temporal constraints. Therefore, system support is needed to speed up the planning process and develop efficient plans.

The field of artificial intelligence planning provides a number of methods to reason about actions and plans [2]. Among these methods, Hierarchical Task Network (HTN) planning [3,4] is particularly useful for solving real-world planning problems [1]. It allows for the employment of predefined domain knowledge at different abstraction levels, which are often inherent in many domains. In HTN planning, high-level composite tasks are recursively decomposed into simpler subtasks by the domain artifacts' so-called

method. The decomposition process continues until so-called primitive tasks, which represent executable actions, are obtained. Primitive tasks are realized by the domain artifacts' so-called operator. Hierarchical models, such as HTNs, are believed to be natural representations for complex cognitive models [5]. HTN planning procedures have been shown to be highly efficient in running time. One of the main reasons for this efficiency is their demonstrated ability to represent complex problem-solving knowledge [6] even though, in the worst case, it is harder than classical planning [7]. The abovementioned features are the motivations to adopt HTN planning as the foundation of this paper.

To enhance the HTN-based planning technique to resolve real-world planning problems, the ability to reason with resources and temporal constraints is necessitated. Take the emergency response task planning as an example: the efficiency of the obtained action plans is significantly influenced by the deployment of a variety of resources, such as vehicles, airplanes, medical devices, and excavating tools, which are commonly of varied type and limited quantity [8,9]. Furthermore, tasks are intrinsically inter-related due to the emergence of temporal constraints from dependency relationships, execution policies, and resource scarcity. This requires that the planner not only generate the actions with corresponding sequence to complete the prescribed tasks but also schedule

* Corresponding author at: School of Automation, School of Management, Huazhong University of Science and Technology, Wuhan 430074, China.

E-mail address: qichao@hust.edu.cn (C. Qi).

the time and resources for those actions. Therefore, HTN planning with temporal and resources constraints can be seen as an integration of planning and scheduling, which are often separated into two phases for ease of problem solving. Considering that planning and scheduling inherently interact with each other, their integration is expected to improve the overall system performance.

Typically, users might define temporal constraints at the start time or end time of tasks or actions. Resource conflicts will result in additional temporal constraints, especially when resources are scarce. Dealing with temporal constraints and resource conflicts simultaneously during HTN planning is challenging because of competing constraints. First, traditional planning mainly stresses the identification of the sequence of actions according to prescribed logic. However, resource availability and temporal constraints require the planner to generate the proper plan with consideration of the scheduling of resources and time allocation, which adds significant difficulties. Constraints resulting from resource scarcity complicate the planning procedure since actions scheduled in a specific period might be invalid when resource becomes unavailable [10]. Second, considering the hierarchy of a planning problem, there might be multiple ways for a composite task to be decomposed. New temporal constraints can be inherited from parent nodes during the decomposition process. Consequently, all temporal constraints are obtained only when the planning process is completed. Third, the availability of various resources and the complex relationships among tasks, such as precedence constraints, deadlines, and dependency constraints [11], augment the difficulty and must be taken into account during planning. HTN planning has been applied in a variety of fields where task planning is needed, such as mobile robots [12,13], manufacturing [14], feature modeling [15], and emergency response decision-making [16–18]. Although some efforts have been focused on temporal constraints in recent studies on HTN planning [19–21], simultaneously consideration of temporal constraints and resource deployment has not been deeply investigated [22].

A straightforward way to satisfy all of the constraints is to assign values to time variables until the decomposition process for HTN planning is completed, which is the least-commitment strategy for time variables [2]. Unfortunately, this strategy results in a challenging problem for resource management under HTN planning. Because some temporal constraints are associated with resource conflicts, least-commitment on time variables means that the quantity of resources at any point in time is uncertain. With an uncertain state, the HTN planner may have trouble choosing a suitable decomposition during planning. In other words, for HTN planning, resource reasoning requires early commitment on time variables, which conflicts with the requirement of temporal constraints' reasoning. Therefore, we have to find a way to maintain the consistency of all temporal constraints while simultaneously providing a deterministic resource state for the planner.

To address this problem, we propose an algorithm called GSCCB-SHOP2 for HTN planning with resources and temporal constraints. GSCCB-SHOP2 is capable of handling multi-capacity discrete resources and complex temporal constraints simultaneously. To address the challenges mentioned in the previous paragraph, three modules are proposed and integrated closely to make the algorithm address temporal and resource reasoning during planning. In particular, a timeline-based resource model and corresponding resource state updating rules are designed to enable resource reasoning within the HTN planning procedure. The algorithm follows the early-commitment strategy on variables to simplify the reasoning of resource availability. The well-known Simple Temporal Network (STN) [23] is utilized to represent temporal constraints. When the methods or operators are instantiated, resource constraints are converted into temporal constraints among associated tasks and are inserted into the STN. The second module is the Check Con-

flict and Backtrack (CCB) module, which is proposed to maintain the consistency of the STN. This module guarantees that all temporal constraints are satisfied even under the early-commitment strategy. To improve the resource allocation and the runtime performance, the third module, Guide Seek (GS), is proposed. This module can also guarantee that the preconditions of the actions already in the current plan will not be disrupted by the effects of the actions later added to the plan. The proposed HTN planning algorithm GSCCB-SHOP2 is developed based on Simple Hierarchical Ordered Planner 2 (SHOP2) [24–26], one of the latest well-known open-source HTN planners. SHOP2's functionality is inherited by GSCCB-SHOP2, allowing partial ordering between tasks in the problem description [24]. SHOP2 has been utilized in a variety of practical applications [27–29].

This paper is organized as follows: the related works are reviewed in the next section; Section 3 presents the proposed HTN planning algorithm; the experimental studies conducted are presented in Section 4 to validate the efficiency of the proposed planning algorithm; and finally, the study's conclusions and possible improvements are discussed in Section 5.

2. Related works

The temporal relationships among tasks addressed in the studies of HTN planning can be classified into three types: qualitative temporal relationships, durative and concurrency actions, and general temporal constraints. Qualitative temporal relationships, such as partially ordered actions, can be satisfied by enabling the planner to cope with ordered and unordered subtasks. Several existing HTN planners include this capability [20,24,30,31]. The other two types of temporal relationships are closely related to our work. Durative and concurrency actions are mainly presented and reasoned by applying the notion of timelines [19–25]. General temporal constraints underlying a plan are described and propagated with the help of STN in several HTN planners [8,17,20,32].

The timeline approach records the changes of state with a time advance mechanism. Timelines have associated starting and ending times. Hence, actions with durative and concurrency properties can be reasoned with by identifying the start times and end times of those actions. In particular, there are two forms of timeline approaches. The first form employs a single timeline associated with the state of the entire planning system [19,33,34]. The corresponding time advance mechanisms are designed to help the planner choose a decomposition branch while keeping track of the existing constraints. The second form employs a timeline for each single-valued dynamic property, such as the multi-timeline preprocessing (MTP) technique proposed by Nau et al. [25]. MTP is quite flexible to describe all types of dynamic properties. However, this method might deteriorate planning efficiency when the tasks are constrained by tight deadlines due to the increased number of backtracks necessary to generate a consistent plan.

With the timeline approach, the planner knows the current state exactly at each step of the planning procedure because it follows the early-commitment strategy on time variables. For actions already added to the plan, the associated time variables have deterministic assignments. Therefore, the current state is always deterministic during planning, which leads to the reduction in planning time. However, the timeline approach has limited ability to address general temporal constraints, for instance, the deadline constraints and the “before” relationship in Allen's algebra [19].

In the studies [17,20,32,35], STN is adopted to handle general temporal constraints. The relative temporal constraints among time variables are expressed as time windows, and consequently, an STN is constructed [32]. During the planning procedure, the temporal constraints are propagated, and the STN is checked by the path consistency (PC) algorithm to determine its consistency [23]. Af-

Download English Version:

<https://daneshyari.com/en/article/4946097>

Download Persian Version:

<https://daneshyari.com/article/4946097>

[Daneshyari.com](https://daneshyari.com)