Position Statement: Reactive Workshop Guide Tool System Structure

Sueli CUNHA
Laboratoire d'Informatique Appliquée (LIA)
Université de Pau et des Pays de l'Adour
Av. de l'Université - BP : 1155 - Pau - CEDEX
Sueli.Cunha@univ-pau.fr
http://www.univ-pau.fr/~cunha

Abstract

The planning problem is based on a production specification that is composed of real world tasks which can be described by functions and constraints. This work deal with a description of a reactive workshop guide tool system, which takes unknown factors into account. This system is based on task definition and the feasibility of the tasks can be verified by analyzing the coherence among their constraints by using constraints propagation.

Introduction

The AI planning problem is basically described as given an initial world description, a goal world description and a set of tasks/operations/actions, to find a sequence of actions, called *plan*, which leads from the initial world description to the goal world description. The approach presented here is the base of LIA's research subject which deals with *formal specification of operation tasks* and *reactive workshop guide tool.* Figure 1 illustrates the structure of a reactive workshop guide tool system. The dot arrow represents the workshop feedback in a possible failure, brought about by unknown factors.

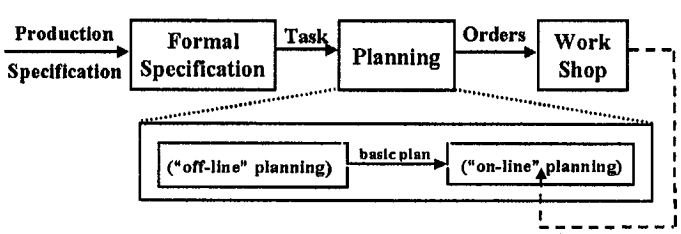

Figure 1: Structure of the reactive workshop guide tool system proposed

A formal task specification aims to describe *what* will be done but not *how* it will be done. Then, from a production specification, expressed in a specific technique vocabulary, the *formal specification* phase produces, based on task definition (Ouriachi 1991), a formal and general description of a task. Next, by way of an *"off-line" planning* (or *predictive planning*), a set of plans allowing to realize this task can be described (Ouriachi 1991; Cunha 1999). From this predictive plan, one basic plan is chosen, according to some criterea, to be carried out ; orders (actions) are giving to the *work shop* to be realized as they advanced (*on-line planning*). The others plans can be used, entirely or partially, as an alternative solution if some problems happen at the time realization of the task. In the next sections, each one of these phases is presented.

Formal Specification

A formal task specification aims to describe the *function* of a task T without considering the *manner of* it will be realized (Ouriachi 1991). It is noted by a triplet $\mathrm{SF(T)} = (\theta_I, \mathcal{H}, \theta_F)$ where : i) θ_I indicates the initial world descriptions and/or some preconditions needfull to carry out the task T ; ii) θ_F indicates the goal world descriptions and/or some post-conditions which must be verified after performing the task T; finally iii) $\mathcal{H}$ is a function whose application allows a transformation of the initial world description into the goal world description.

A task specification can be refined by describing the various abstraction levels of its functionality. In a formal task specification, these abstraction levels can be expressed by using *intermediated tasks* description. In other words, in an abstraction level i, the formal specification of a task T can be expressed through an intermediated tasks set $\{\mathrm{T}_{i_1}, \mathrm{T}_{i_2}, \ldots, \mathrm{T}_{i_n}\}$ with respective formal specification $\mathrm{SF}(\mathrm{T}_{i_k}) = (\theta_{I_k}, \mathcal{H}_k, \theta_{F_k})$ $1 \leq k \leq n$.

The $\mathrm{SF}(\mathrm{T}_{i_k})$ expressions are named subspecifications of T and the *intermediated specification* of T, at the level i, is done by the n-uple below :
$\mathrm{SF(T)} = <(\theta_{I_1}, \mathcal{H}_1, \theta_{F_1}), (\theta_{I_2}, \mathcal{H}_2, \theta_{F_2}), \ldots, (\theta_{I_n}, \mathcal{H}_n, \theta_{F_n})>_{op}$
where *op* represents the relationship among the tasks T_{i_k}, $1 \leq k \leq n$. In other words, *op* indicates, for example, if the tasks T_{i_k} (also called *subtasks* of T) must be realized in *sequence* ($op = s$) or if one of them can be *chosen* ($op = c$) to achieve the T functionality. A choice among subtasks of a task characterizes either a similarity between them or an option according to the case (preconditions and/or post-conditions).

The level the less abstract is represented by a named *elementary specification* which is expressed exclusively by predefined functions, which specify *elementary tasks.*

An elementary task is either an operation or a (little more) complex task which has already been specified. The second case indicates the possibility of a formal task specification reuse. Moreover, in the case of each elementary task is an operation, the elementary specification corresponds to a plan.

The relationships quoted above, among the subtasks of a task T, can be represented by a *concatenation* and a *branching* of *E-task-graphs* (Cunha 1999). An E-task-graph is a directed graph which has exactly one *entry vertex* and one *exit vertex* which represent, respectively, the initial state θ_I and the final state θ_F of T. The other vertices represent the intermediated states. The edges, each labeled by an intermediated task name, represent the relationships between intermediated tasks. Elementary E-task-graphs, those ones composed by only one edge, represent elementary tasks.

The pre-conditions, the postconditions, as well as the initial and the goal world description can be expressed by constraints (temporal constraints, state constraints, capacity constraints, among others). Thus, the feasibility of a task T can be verified by analyzing the coherence among their constraints by using constraints propagation or by modelling the task as *structural constraint satisfaction problem* (Nareyek 1999) over E-task-graphs. Moreover, if T is in a high abstraction level, the feasibility of T can be verified by analzyng not only the feasibility of each one of its subtasks, but also their relationship coherence.

Planning

Once a formal task specification is formulated for a task T, a predictive plan $\mathcal{P}$ can be stated by expressing a set of possible plans ($\mathcal{P} = \{P_i,\ 1 \leq i \leq m\}$) which allow to perform T. Then, for example, let $\mathcal{T}$ be a set of known tasks, $\mathcal{T} = \{T_i,\ 1 \leq i \leq 20\}$ where $\mathcal{O} \subset \mathcal{T}$, $\mathcal{O} = \{T_4, T_5, T_6, T_9, T_{11}, T_{12}, T_{13}, T_{15}, T_{16}, T_{17}, T_{18}, T_{19}, T_{20}\}$ is a set of operations ; let T be a new task such that $SF(T) = <(\theta_{I_1},\mathcal{H}_1,\theta_{F_1}),(\theta_{I_2},\mathcal{H}_2,\theta_{F_2}),(\theta_{I_3},\mathcal{H}_3,\theta_{F_3})>_s$ or in another way, $SF(T) = <SF(T_1), SF(T_2), SF(T_3)>_s$. Consider also $\mathbf{SF(T_1)} = <SF(T_4), SF(T_5), SF(T_6)>_s$, $\mathbf{SF(T_2)} = <SF(T_7), SF(T_8)>_c$, $\mathbf{SF(T_3)} = <SF(T_9), SF(T_{10})>_s$; $\mathbf{SF(T_7)} = <SF(T_{11}), SF(T_{12})>_s$, $\mathbf{SF(T_8)} = <SF(T_{13}),SF(T_{14}),SF(T_{15}),SF(T_{16})>_s$; $\mathbf{SF(T_{10})} = <SF(T_{17}), SF(T_{18})>_c$; $\mathbf{SF(T_{14})} = <SF(T_{19}), SF(T_{20})>_c$.

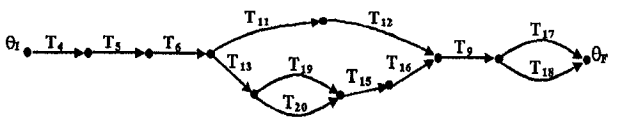

Figure 2: E-task-graph representing T

This example holds 6 plans because of the choices existing in tasks T_2, T_{10} and T_{14}. Thus, based on some criterea (e.g. time, costs) a basic plan P_i is chosen to be carried out. The operations set which compose P_i are carried out the further they advanced according to the work shop conditions like resource assignement, capacity limitation, machine's breakdown, among others.

Reactivity

Whenever a disruption of running system happens during an execution of one of the P_i's tasks, a workshop feedback requests an alternative solution from the on-line planning system. When no solution is found, the on-line planning system, in its turn, tries to find in the predictive plan $\mathcal{P}$ another basic plan P_j which can, entirely or partially, stands in for P_i.

The partial replacement means finding an alternative task that can replace only the failed subtask. Then, the execution of P_i goes on according the original basic plan. In the previous example, supposes $P_i = < SF(T_4), SF(T_5), SF(T_6), SF(T_{13}), < \mathbf{SF(T_{19})}, \mathbf{SF(T_{20})}>_c SF(T_{15}), SF(T_{16}), SF(T_9), SF(T_{17})>_s$. If the task $\mathbf{T_{19}}$ fails, the on-line planning system can propose its replacement by $\mathbf{T_{20}}$. After the execution of T_{20}, P_i goes on with T_{15}. By contrast, if a faillure happens at the time execution of T_{17}, only the predictive plan system can propose T_{18} to stand in for T_{17}.

Conclusion and Perspectives

This paper presented an approach, based on task definition, which aims to supply a reactive workshop guide tool system. The various abstraction level of a task fonctionality on the one hand allows the reuse of task specifications and on the other hand can offer choices between tasks with similar fonctionality. These choices can facilitate a reactivity not only in the on-line planning level (scheduling revision (E.Bensana & Sicard 1993)) but also in the off-line planning level (planning revision). My personal further research topics include the introduction of constraints techniques in tasks feasibility analyze in predictive planning system, by analyzing, inspired by (Nareyek 1999), the E-task-graph constraint satisfaction. The goal is do not include in the predictive plan set a plan relative to tasks not well described, e.g. tasks whose the pre and/or post-condtions are no coherent.

References

Cunha, S. F. 1999. *Similarité de Raisonnements Exprimées par un Modèle de Tâches.* Ph.D. Dissertation, Université Joseph Fourier (Grenoble I). http://www.univ-pau.fr/~cunha/these.ps

E.Bensana, and Sicard, M. 1993. A constraint based system for schedule construction, maintenance and revision. *13th IJCAI, Workshop on Knowledge Based Scheduling.* ftp://ftp.cert.fr/pub/bensana/ijcai93.ps.

Nareyek, A. 1999. Structural constraint satisfaction. In Press, A., ed., *AAAI Workshop on Configuration, Technical Report*, 76–82. http://www.ai-center.com/home/alex/publications.html.

Ouriachi, K. 1991. *Contribution à la Spécification Formelle des Tâches Opératoires.* Ph.D. Dissertation, Université de Valenciennes et du Hainaut-Cambrésis.