The Soft Real-Time Agent Control Architecture

BRYAN HORLING
bhorling@cs.umass.edu
Department of Computer Science, University of Massachusetts, Amherst, MA 01003, USA

VICTOR LESSER
lesser@cs.umass.edu
Department of Computer Science, University of Massachusetts, Amherst, MA 01003, USA

RÉGIS VINCENT
vincent@ai.sri.com
SRI International, 333 Ravenswood Avenue, Menlo Park, CA 94025, USA

THOMAS WAGNER
Tom.Wagner@darpa.mil
DARPA/IPTO, 3701 North Fairfax Drive, Arlington, VA 22203, USA

Published online: 26 September 2005

Abstract. Real-time control has become increasingly important as technologies are moved from the lab into real world situations. The complexity associated with these systems increases as control and autonomy are distributed, due to such issues as temporal and ordering constraints, shared resources, and the lack of a complete and consistent world view. In this paper we describe a soft real-time architecture designed to address these requirements, motivated by challenges encountered in a real-time distributed sensor allocation environment. The system features the ability to generate schedules respecting temporal, structural and resource constraints, to merge new goals with existing ones, and to detect and handle unexpected results from activities. We will cover a suite of technologies being employed, including quantitative task representation, alternative plan selection, partial-order scheduling, schedule consolidation and execution and conflict resolution in an uncertain environment. Technologies which facilitate on-line real-time control, including meta-level accounting, schedule caching and variable time granularities are also discussed.

Keywords: agent control, scheduling, soft real-time, resource management.

1. Overview

In the field of multi-agent systems, much of the research and most of the discussion focuses on the dynamics and interactions between agents and agent groups. Just as important, however, is the design and behavior of the individual agents themselves. The efficiency of an agent’s internal mechanics contributes to the foundation of the system as a whole, and the degree of flexibility these mechanics offer affects the agent’s achievable level of real-time planning and scheduling of activities, particularly in its interactions with other agents [24, 30]. We believe that a general control architecture, responsible for both the planning for the achievement of temporally constrained goals of varying worth and the sequencing of actions local to the agent that have resource requirements, can provide a robust and reusable platform on which to build high level reasoning components. In this article, we will discuss the design and implementation of the Soft
Real Time Architecture (SRTA), a generic planning, scheduling and execution subsystem designed to address these needs [41]. The SRTA architecture provides several key features:

1. The ability to quickly generate plans and schedules for goals that are appropriate for the available resources and applicable constraints, such as deadlines and earliest start times.
2. The ability to merge new goals with existing ones, and multiplex their solution schedules.
3. The ability to use explicit representations of uncertainty and efficiently handle deviations in expected plan behavior that arise out of variations in resource usage patterns and unexpected action characteristics.

While the immediate motivation for this work is the domain described in Section 2, we are more generally interested in demonstrating that agents employing complex modeling and decision making techniques can address problems posed by real-world scenarios. The architecture presented in this paper uses such techniques, enabling it to operate effectively in open, unpredictable environments by using online planning and scheduling algorithms that explicitly reason about uncertainty and have the ability to explore alternative ways to satisfy goals, temporal constraints and resource requirements. At the same time, it is also efficient enough to work in soft real-time, manage interdependencies between tasks and resources, and satisfy commitments that may be formed between entities. In particular, this work is differentiated from others [1, 32, 33] in its explicit use of uncertainty, soft interrelationships, probabilistic action expectations, and a range of commitment types. We will show how this type of framework can provide many capabilities needed for sophisticated multi-agent applications.

Abstractly, SRTA operates as a single functional unit within an agent, which itself is running on a conventional (i.e. not real-time) operating system. The SRTA controller is designed to be used in a layered architecture, occupying a position below the high-level reasoning component in an agent [2, 51]. In this role, it will accept new goals, report the results of the activities used to satisfy those goals, and also serve as a knowledge source about the potential ability to schedule future activities by answering what-if style queries.

The system has evolved and been constructed over several research projects into a set of interacting components and representations, as shown in Figure 1. We first assume that goals can arrive at any time, in response to environmental change, local planning, or because of requests from other agents. A domain-independent, hierarchical task network description language called TÆMS is used to describe goals, which supports quantitative, probabilistic models of activities, including non-local effects of actions and resources and a variety of ways to define how tasks decompose into subtasks (for example, Figure 4) [6, 15]. In particular, the uncertainty associated with activities can be directly modeled through the use of quantitative distributions describing the different outcomes a given action may produce. Commitments and constraints can be used