Future Generation Computer Systems 31 (2014) 200–212

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Using hybrid scheduling for the semi-autonomous formation of expert teams

Edmund H. Durfee *, James C. Boerkoel Jr., Jason Sleight

Computer Science and Engineering, University of Michigan, Ann Arbor, MI 48109, USA

HIGHLIGHTS

- Assignment and scheduling of expert teams is modeled as a hybrid scheduling problem.
- The hybrid scheduling formulation is often faster to solve and yields good solutions.
- The application uncovered opportunities for improving hybrid scheduling algorithms.
- Human insights can improve solutions by incrementally introducing constraints.

ARTICLE INFO

Article history:
Received 19 December 2011
Received in revised form
24 October 2012
Accepted 6 April 2013
Available online 19 April 2013

Keywords:
Teamwork
Scheduling
Constraint satisfaction/optimization
Collaborative systems

ABSTRACT

Selecting and scheduling human experts to cooperatively solve a problem can be a highly complex task, given various constraints (such as what expertise is needed and when) and preferences (such as which expertise an expert most prefers to exercise). Computational agents can thus greatly help users form and schedule expert teams. This paper introduces a new formulation of the team formation and scheduling problem as a Hybrid Scheduling Problem (HSP) and compares the performance of an agent using the HSP formulation to a prior agent-based approach. We empirically demonstrate the promise of the HSP formulation and highlight how the application of HSP techniques to this problem has led us to identify important modifications to mechanisms that improve HSP solving. Finally, we summarize how the HSP formulation can support human-agent collaboration during the process of forming and scheduling expert teams.

1. Introduction

Large-scale selection and scheduling problems can overwhelm the cognitive limitations of people. This paper, for example, focuses on dynamically forming teams of human experts whose combined expertise, applied in coordinated ways, can solve difficult problems, such as delivering emergency medical care and diffusing inflammatory confrontations with complex cultural, linguistic, and religious undertones. Given a worldwide network of possible experts to draw upon, each with his or her own specific aptitudes, interests, and availabilities, optimizing the formation of a team involves considering a vast space of possible combinations of individuals. Furthermore, because demands for teams often arise contemporaneously from different sources, assignment and scheduling decisions about experts' participation are intertwined.

As mentioned above, a motivating example of where this type of team-formation problem arises is in delivering medical care in an emergency situation, such as to support treatment of a wounded soldier in a combat situation. A medic coming to the aid of a wounded soldier will have limited expertise and context for making decisions that can have long-term consequences for the soldier's future quality of life. Thus real-time, on-demand consultation with a team of experts knowledgeable about triage, battlefield medical procedures, prognoses for various outcomes, rehabilitation possibilities, and even religious or cultural biases pertinent to the wounded person could be invaluable. A team of experts from around the world whose collective expertise covers these needs could be rapidly assembled and cooperatively interact over the communication network to converge on situation-specific recommendations for the medic. Among the many challenges in realizing such a vision, and the challenge that this paper focuses on, is sifting through the combinatorial number of possible expert teams to find ones with the right mix of expertise, with experts whose schedules mesh well and who are able to interact effectively (e.g., have shared language and ontology).

To meet this kind of challenge, we posit that human decision making can be augmented with computational agents to solve problems that exceed human capabilities. However, we also recognize that computational agents are unable to make nuanced distinctions between alternatives that could impact the quality, and even functionality, of an expert team. For example, a person

* Correspondence to: 2260 Hayward Ave., University of Michigan, Ann Arbor, MI 48109, USA. Tel.: +1 734 936 1563; fax: +1 734 763 1260.

E-mail addresses: durfee@umich.edu (E.H. Durfee), boerkoel@umich.edu (J.C. Boerkoel Jr.), jsleight@umich.edu (J. Sleight).

0167-739X/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.future.2013.04.008

in need of an expert team might have preferences that are difficult or too sensitive to express, such as doubts about some expert's competence or collegiality, and yet should be reflected in teaming decisions. Human-agent collaboration (HAC) should thus be central to the system's design and implementation.

We have participated in designing a high-level framework for HAC [1] that integrates human-centered system notions in a multiagent system (MAS) architecture. This paper, which elaborates and extends on our work presented in [2], investigates and more precisely characterizes the specific problem of expert team formation and scheduling. It introduces a new strategy for solving this type of complex problem and contrasts it with a prior approach, describing the representational and empirical techniques employed, and explains the manner in which human collaboration is achieved with (and through) the agent system. The techniques we introduce are well-suited to a broader space of problem domains where a full specification of all constraints and preferences is infeasible (e.g., because of privacy concerns or an evolving understanding of the situation), of which the formation of teams of compatible medical experts is the representative example we explore in this paper.

We begin with a brief overview of the HAC context, summarize the components of the HAC framework, and review related work from the literature (Section 2). We then drill down to formally describe the core expert team formation and scheduling problem and summarize a previously-reported strategy for solving it (Section 3). In Section 4, we describe a new strategy for solving this problem by formulating it as a Hybrid Scheduling Problem (HSP) and applying state-of-the-art HSP solution techniques. Next, we conduct a comparison, both analytical and empirical, between the previous and new strategies, evaluating their strengths and limitations along various dimensions of interest for the specific problem and for HAC more generally (Section 5). To even further improve the new HSP-based strategy's performance, in Section 6 we investigate the application of Hybrid Constraint Tightening (HCT) and show how the specific expert teaming application identifies more widely-applicable improvements to HCT. Finally, in Section 7 we look at the opportunities for human-agent collaboration with the new HSP formulation, and in Section 8 we summarize the lessons learned in this work and suggest possible future directions.

2. Background

An initial design of and a conceptual solution to the problem of employing human knowledge within human-agent collaboration processes in a simulated combat medical scenario was presented in [3]. The combat medical scenario required assigning and scheduling human experts in an on-demand fashion, and that paper outlined the design of a human-agent collaboration (HAC) framework for solving this problem.

A subsequent paper [1] elaborated on the explicit design and implementation of the HAC framework. The software architecture embodying the HAC framework employs a multiagent-system-based design and implementation strategy. The HAC software system includes various constituent functional components, implemented as agents (e.g., a case manager, a core HAC agent, a matchmaking agent, and a scheduling agent), the information flow and control flow among these agents, and the underlying algorithms and mechanisms realizing the HAC capabilities.

2.1. HAC architecture

The schematic of the HAC architecture from that prior paper [1] is shown in Fig. 1. The boxes represent the reasoning agents. The arrows represent the information and control flow in response to a user's request for an expert team to solve a particular problem—referred to as an HAC case. The numbers with the arrows indicate

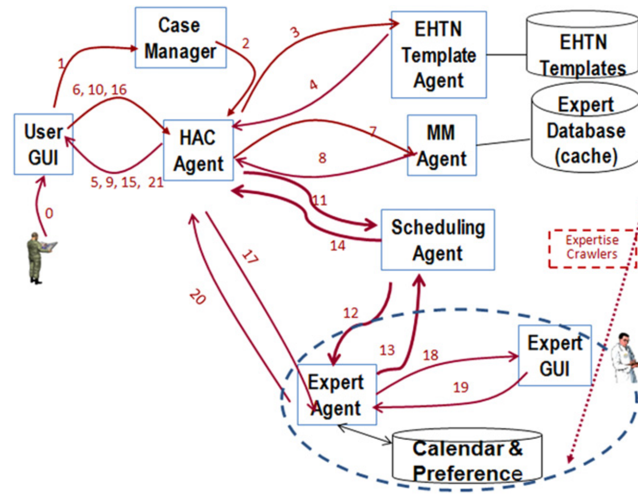

Fig. 1. HAC schematic architecture.

the order of steps during a typical HAC process without any exceptions arising.

In brief, in step 0 a human user expresses a medical case, through her GUI interface agent, that represents the need for a team of (human) experts to resolve that case. The GUI interface agent contacts the Case Manager agent (step 1) which instantiates an HAC agent for that case (step 2). Given the case description, the HAC agent requests (step 3) and receives (step 4) from the EHTN Template agent candidate plan templates for resolving the case, where as the agent's name suggests the plan templates are represented as extended hierarchical task networks (EHTNs) [4]. EHTNs capture the hierarchical decomposition of larger tasks into smaller ones, along with the relationships between the tasks and that must hold between agents performing them. The EHTN Template agent has access to a library of domain-specific EHTNs that capture standard operating procedures in the domain.

The candidate EHTNs are presented to the user via the GUI (step 5) and the user's preferences over which of them to pursue is reflected back to the HAC agent (step 6). Given the most preferred template, the HAC agent interacts (steps 7 and 8) with the matchmaking (MM) agent to find candidate experts (from potentially thousands available on the network) to play the various roles in the plan template, and then the HAC agent again works with the GUI agent (steps 9 and 10) to give the human user a chance to express preferences over the choices of experts. The HAC agent then contacts the Scheduling agent (step 11) to converge on the timings of the experts' interactions, where the Scheduling agent in turn probes the agents representing each of the experts (steps 12 and 13) to determine available time intervals. The Scheduling agent communicates candidate schedules for consulting the experts (step 14) to the HAC agent, which again shares these with the GUI agent to acquire any preferences of the user over the options (steps 15 and 16). Given a chosen option, the HAC agent then communicates with the Experts' agents (step 17) to reserve the expert at the desired times, where the Expert agent will interact with its corresponding human expert via the GUI (steps 18 and 19) to confirm acceptance of the commitment, which then is finally reflected back to the HAC agent (step 20). Of course, if at any point in this process the system hits a dead end (e.g., no experts available for a role, chosen experts have incompatible availabilities, etc.), the process can backtrack to the next-most-preferred option, where the user's preferences over plans, expert assignments, and timings are actively sought throughout this process as described.

There are many steps to the process as just described. In this paper, we focus on a particular portion of the overall problem-solving process, which is the subproblem of how experts are