Available online at www.sciencedirect.com
ScienceDirect
Decision Support Systems 43 (2007) 1014–1030
Decision Support Systems
www.elsevier.com/locate/dss

The theory and experiments of designing cooperative intelligent systems

Parag C. Pendharkar *

Information Systems, School of Business Administration, Penn State University at Harrisburg, 777 West Harrisburg Pike, Middletown, PA 17057, United States

Available online 13 June 2005

Abstract

In this paper, we identify the business problems that lend themselves to the design of cooperative intelligent systems and empirically demonstrate the design and application of a multi-agent intelligent system for production scheduling. Our experiments suggest that a multi-agent system where agents coordinate their actions generally performs better than a multi-agent system where agents do not coordinate their actions.

Keywords: Genetic algorithms; Cooperative intelligent systems; Distributed artificial intelligence; Learning; Multi-agent systems

1. Introduction

Cooperative intelligent systems is one of the dynamic research areas in information systems [3,29]. Several researchers have shown that cooperative intelligent systems can be used for low maintenance cost decision support applications that allow decision-makers to take complex decisions [29]. However, not all business problems are suitable for the design of cooperative intelligent systems and the design of cooperative intelligent systems requires consideration of several factors.

Cooperative intelligent systems are also called distributed artificial intelligence (DAI) systems [25]. DAI consists of two sub-fields distributed problem-solving (DPS) and multi-agent systems (MAS) [5]. A DPS system consists of a set of independent geographically dispersed computer systems (problem-solvers) that share "solutions" to solve a problem that none of the independent computer systems can solve independently. Both data and knowledge, in a DPS system, are geographically dispersed. Unlike the distributed database systems and the inter-organizational systems (IOS), a DPS system shares solutions and not data. In a DPS system, problem-solvers that will work together on a given problem and the solutions that will be shared by problem-solvers are usually known. A multi-agent system (MAS) is a set of geographically dispersed computer systems that "dynamically" work together, through communication, to solve problems that none of the independent computer systems can solve indepen-

* Tel.: +1 717 948 6028; fax: +1 717 948 6456.
E-mail address: pxp19@psu.edu.

0167-9236/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.dss.2005.05.028

dently. Unlike a DPS system, computer systems that will work together to solve a given problem are not known in a MAS and are decided dynamically. Further, communication among computer systems in a MAS can be data, hypotheses and knowledge.

Multi-agent systems are of considerable complexity with respect to their functionality and structure [43]. For most application tasks, even in simple environments, it is difficult to determine the behavioral repertoire of an agent in a multi-agent system [43]. For example, determining behavioral repertoire of an agent in a multi-agent system requires a decision-maker to have a priori knowledge of future environmental requirements, knowledge of the availability of each agent at each environmental state in the future and knowledge about how agents will interact in response to the future environmental requirements [43]. The lack of availability of a priori knowledge necessitates the design of an adaptive system that can react to uncertain dynamic situations. A MAS offers features such as parallelism, robustness and scalability, which cannot be handled by centralized systems [43]. In particular, a MAS is used in domains that require integration of knowledge from multiple sources, resolution of different interests and goal conflicts [43]. Learning coordination in a MAS requires that agents adapt, adjust and learn to work with others agents to solve problems. The key issues of learning effective coordination in a MAS are the information exchange scheme and the coordination strategies between loosely coupled agents to achieve effective overall system performance.

Given the complexity of designing a MAS, the current research aims to address the following issues. (1) What types of business situations are suitable for deploying a multi-agent system? And, (2) what are different design considerations related to a multi-agent system design? We answer both of these questions through literature review and an empirical demonstration of a multi-agent intelligent system for a production scheduling environment. The rest of the paper is organized as follows: in Section 2, we review available literature in cooperative intelligent systems (DPS and MAS) area. In Section 3, we describe a production scheduling problem, a framework for genetic algorithm (GA)-based learning and different types of coordination groups, and propose a general multi-agent coordination strategy. In Section 4, we detail the results of our simulation experiments and statistical analyses. In Section 5, we conclude the research by describing significance of our research and highlighting possible future extensions.

2. Overview of distributed problem-solving and multi-agent systems

Traditional work in artificial intelligence and decision support systems has been limited to problem-solving in the context of a single knowledge base. Research in the DPS, on the other hand, has focused on problem-solving related to a group of decentralized and loosely coupled knowledge bases. A DPS system has several advantages over a single, monolithic, centralized problem-solving system [5,15,34,35]. These advantages are—(1) faster problem-solving by exploiting parallelism, (2) reducing communication by transmitting only high-level partial solutions rather than raw data to a central site, (3) increased flexibility by creating problem-solvers, with different abilities, to solve the current problem and (4) increased reliability by allowing other problem-solvers to replace failed ones [13,43]. The distributed control makes a DPS system suitable to problems that transcend highly specialized functional boundaries [11].

There are several applications of the DPS that were reported in the literature. Among the popular applications are electronic meeting systems (EMS) and distributed meeting scheduling system (DMSS) [12]. A study by Nour and Yen [26] provides a DPS conceptual base for developing an EMS. The study presents a new conceptual framework where an EMS was at the center of three other large variables called fundamental elements. The EMS proposes *self-reliance* and *interdependence* as fundamental issues for effective decision-making. Both self-reliance and interdependence provide a useful perspective towards decision-making by achieving “global coherence with local control” [17]. Hewitt [17] noted that having both self-reliance and interdependence often led to conflict because of the following reasons.

(1) Asynchrony enables the decision-makers more impervious to communication failures and allows them to be more self-reliant.
(2) Autonomous decision-making enables the participants in an EMS to react immediately to changing circumstances.