Available online at www.sciencedirect.com
ScienceDirect
Robotics and Computer-Integrated Manufacturing 23 (2007) 624–629
Robotics and Computer-Integrated Manufacturing
www.elsevier.com/locate/rcim

Multi-agent-based task assignment system for virtual enterprises

Kyung-Hyun Choi[a,*], Dong-Soo Kim[b], Yang-Hoi Doh[b]

[a]School of Mechanical Engineering, Cheju National University, 66 Jejudaehakno, Jeju-si, Jeju-do 690-756, Republic of Korea
[b]School of Electronic Engineering, Cheju National University, 66 Jejudaehakno, Jeju-si, Jeju-do 690-756, Republic of Korea

Abstract

Manufacturing enterprises continuously have to cope with changing markets that are unpredictable and diverse, with increased global competition and with ever-changing customer demands. These requirements have led to the emergence of the virtual enterprise (VE). The creation of these enterprises, called as VEs, is becoming a growing trend as enterprises concentrate on their core competencies and economic benefit. An enterprise participating as a member of the VE should take initiative in involvement according to the internal production condition, which is changing dynamically with independent management sovereignty. However, most of researches have not considered these issues, that is the constituting enterprises have been regarded as one of the enterprises having their work stations distributed geographically. This paper proposes a multi-agent-based task assignment system for VEs, which attempts to address the selection of individually managed partners and the process of assigning tasks to them. A case example of assigning the tasks to partners is presented to illustrate and prove the proposed system's applicability.

Keywords: Virtual enterprise; Multi-agent system; Resource allocation; Intelligent manufacturing system

1. Introduction

Manufacturing enterprises continuously have to cope with changing markets that are unpredictable and diverse, with increased global competition and with ever-changing customer demands. These requirements have led to the emergence of the virtual enterprise (VE). Therefore, the non-core business portion of the enterprise is conducted by cooperative vendors. VEs are defined as a way of organizing manufacturing activities, where different and independent partners exploit business opportunities by establishing an enterprise cooperative [1].

A VE can be created through an enterprise consortium. In these companies, planning and control activities can be very complex, and have to take place both within the enterprise and across the whole supply network in order to achieve high level of performance. Planning methodologies have been introduced to distribute tasks into proper enterprises in the consortium. Pronet II was introduced to develop a reference architecture for VEs. Henry et al. [2], and Baker et al. [3] selected an enterprise base on the minimum inventory cost by using the neural network model. Ko et al. [4] designated an enterprise considering minimizing production and transportation costs by using Tabu Search Heuristics. With integer programming, Wo et al. [5] suggested an enterprise selection method based on minimum transportation cost according to geographic position and transportation method. Banaszak et al. [6] employed the constraints theory framework to study constraint-based production flow coordination rules for a VE [2–6].

An enterprise participating as a member of the VE should take initiative in involvement according to the internal production condition, which is changing dynamically with independent management sovereignty. However, most of researches have not considered these issues, that is the constituting enterprises have been regarded as one of the enterprises having their work stations distributed geographically. By the nature of VE decentralization, multi-agent approaches have been introduced to deal with problems associated with distribution process planning and control [7,8].

*Corresponding author. Tel.: +080 064 754 3713; fax: +080 064 756 3886.
E-mail address: khchoi@cheju.ac.kr (K.-H. Choi).

0736-5845/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.rcim.2007.02.003

The purpose of this paper is to propose the multi-agent-based task assignment system considering the participating enterprise's production status. The assignment system executes the selection of individually managed partners and the optimal task assignment, which consists of agents such as ordered product data. Those agents cooperate to achieve the optimal task assignment based on the operation sequence graph. A case example of assigning the tasks to partners is presented to illustrate and prove the proposed system applicability.

2. Task assignment system structure

In this paper, the enterprise that carries out the plan and control of production orders is called the main enterprise among the enterprises comprising a VE. The enterprises being supposed to manufacture products in cooperation with each other are called partner enterprises. The multi-agent-based task assignment system carries out unique functions based on the four elements as shown in Fig. 1.

Ordered product data management (OPDM) system stores and manages ordered product information such as customer requests for tasks due, quality level and volumetric and the geometric information on product and parts. Product agent is the only communication channel connecting the VE to the external world and assigning tasks through the coordinating partner enterprise. Physically, the product agent exists in the main enterprise and is extinguished within the VE when the product task is completed. Task agent dynamically assigns tasks registered in the OPDM system to the several product source groups distributed geographically. It monitors and coordinates production facilities within the group and is extinguished automatically when the production is completed. Resource agent manages the status of the task's progression that is accomplished in the facilities using facility parameters and status. It is affiliated with all the production facilities: machining and assembling facilities, AS/RS, conveyer, AGV, and the materials and parts of each enterprise involved in production.

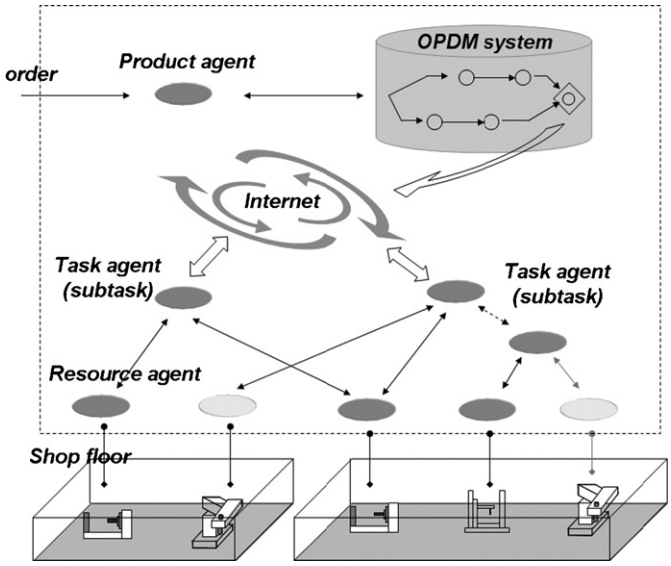

Fig. 1. Architecture of multi-agent-based task assignment system.

3. Task assignment method

3.1. Operation sequence graph

The product agent relays the ordered product information to the OPDM system. This information contains operational sequencing information as shown in Fig. 2. The node represented as circle (O) means "operation," and an O depicted inside symbol ◇ is the abbreviation of "or" and indicates that only one of the several operational groups should be executed. "A" notated inside symbol ◇ is the abbreviation for "and" and means that all the operational tasks should be carried out regardless of the priority order. Also, for operations j and k, if operation j precedes operation k, it is noted as $j<k$.

The task agent selects resource agents that accomplish production in order from the last operation in the subtask. That is, after the agent to perform operation k is selected, the agent for operation $j<k$, is selected.

Here, the set of operational set able to decide assignments after the completion of operation k's assignment is defined as H_k^-. The set of operations that should be decided prior to the decision to assign a resource agent to operation k is designated as H_k^+. H_k^{++} is defined as the operational set performed following operation k.

3.2. Task assignment process

Task assignment employs a method that divides tasks into a sequence of subtasks and assigns them to partner enterprises. As shown in Fig. 2, each task is gradually divided into details on the basis of nodes, until it reaches the operational node with no branch node subtasks. It is then assigned to the resource agent. The task agent is generated in each subtask and it initially assigns tasks or subtasks to a resource agent or to a subtask agent (Fig. 3).

In assigning unit operation, the resource agent first chooses the most suitable operation among the feasible ones. Then the task agent completes the assignment of one

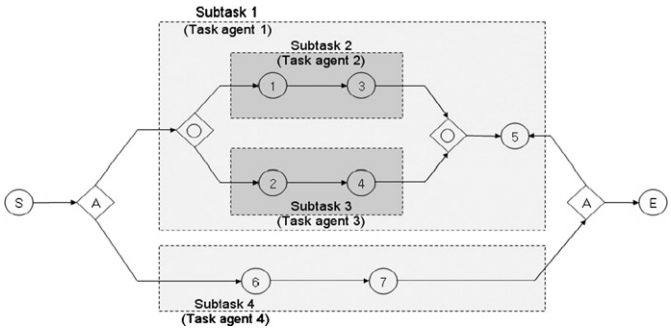

Fig. 2. Operational sequence graph of a task.