

## 一类基于多 Agent 和分布式规则的敏捷生产调度

王艳红<sup>1,2</sup>, 尹朝万<sup>1</sup>

(1. 中国科学院 沈阳自动化研究所, 辽宁 沈阳 110016; 2. 沈阳工业大学 信息学院, 辽宁 沈阳 110023)

**摘要:** Agent 范例为解决制造系统的敏捷生产调度问题提供了一条新途径, 如何构建敏捷生产调度多 Agent 系统结构和 Agent 间的协调与生产调度机制, 成为一个亟待解决的课题. 本文阐述了一类基于多 Agent 和分布式规则构建敏捷生产调度的方法. 首先通过基于功能分解的方法, 给出了管理、资源和工件等三类 Agent 基本组件组成的分布式多 Agent 调度系统结构、Agent 组件基本结构及定义. 其次, 利用基于分布式规则的方法, 建立了 Agent 间的协调策略和调度机制, 实现了敏捷生产调度. 最后给出了应用此方法的调度仿真实验结果.

**关键词:** 敏捷调度; 多 Agent 系统; 分布式规则

**中图分类号:** TH166 **文献标识码:** A

## Multi-agent and distributed-rules based agile production scheduling approach

WANG Yan-hong<sup>1,2</sup>, YIN Chao-wan<sup>1</sup>

(1. Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang Liaoning 110015, China;

2. Department of Information Sciences and Engineering, Shenyang University of Technology, Shenyang Liaoning 110023, China)

**Abstract:** The agent paradigm is one of the most prominent approaches to build an agile production scheduling system, but there are still some key problems to be solved, such as how to construct a proper multi-agent system architecture for the agile production scheduling, and how to create an appropriate mechanism for agents to coordinate or cooperate with each other and to create desired production schedulers. A multi-agent and distributed rule based approach is proposed for production scheduling in agile manufacturing systems in this paper. First, a function-based decomposition method was suggested to make a hierarchical multi-agent system, which was populated by three classes of agents, namely Resource Agents, Part Agents and Management Agents, and the basic structure and definitions of these agents were presented. Then, based on distributed rule methods, the coordination or cooperation strategies and the scheduling mechanism for agents were presented for the purposes of agile production scheduling. Finally, simulation results were given to illustrate this approach.

**Key words:** agile scheduling; multi-agent system; distributed rule

### 1 引言 (Introduction)

制造系统的生产调度问题一直是研究的热点, 早期的研究主要着眼于优化调度, 后来逐步转向柔性调度, 随着科技的发展和社会需求的变化, 目前更倾向于敏捷调度<sup>[1]</sup>. 敏捷调度面向以多品种、小批量、客户化产品为主的敏捷制造<sup>[2]</sup>, 要求在产品开发和制造周期内, 具有较高的适应性和效率响应客户的需求. 因此, 敏捷调度具有更大的难度和复杂性.

Agent 具有的智能性、自治性和合作能力, 为解决敏捷生产调度问题提供了一条新途径, 使之成为

智能制造等领域研究的新热点. 在多 Agent 系统 (MAS) 中, Agent 之间的交互方式大致有两类, 即通过中介的交互或直接交互. 前者通过构造某些中介 Agent 来协调其它 Agent 之间的关系, 例如 Maturana 和 Norrie 等提出的 MetaMorph I<sup>[3]</sup> 和 MetaMorph II<sup>[4]</sup> 系统, 由一类特殊的 Agent ——“Mediator”来协调系统内代表其它物理设备或逻辑功能的 Agent 之间的关系. 后者则不需要中介, 一切靠 Agent 自主完成. 鉴于 Agent 具有的智能性和自治性以及支撑技术水平, 本研究更倾向于后者. 通常, 多数 MAS 采用合同

网协议(contract net protocol)<sup>[5]</sup>方法或其它类似的各种拍卖(Auction)、谈判(Negotiation)方法来处理 Agent 之间的协调与合作问题.例如,Rabelo 等在文献[1]中提出了一种基于多 Agent 的敏捷调度模式,通过 Agent 之间的“谈判协商”完成生产任务分配.合同网方法着眼于环境的不确定性和 Agent 决策的敏捷性,但由于过多地强调任务的分配,而忽视了系统的优化性能.强调 Agent 间自由竞争可能导致某些任务不能分配,造成系统死锁,影响系统的稳定工作.为此,作者提出了一类基于多 Agent 和分布式规则的敏捷调度方法,其特点是:

- 随着 Agent 和规则的分布,复杂的敏捷调度问题得以分解,转变为一系列简单的单一对象、单目标的调度问题,因而能够容易、快速、有效地解决<sup>[6]</sup>;
- 有效控制整个系统的行为,使系统在追求全局性能的同时,实现 Agent 个体局部性能的最大化,进而获得满意的调度结果;
- 可在需要的时候将一些优秀的调度方法和策略集成到相应的规则中,以得到所需的满意性能;
- Agent 组件及其规则具有“即插即用”的特性,降低了系统构造和维护的复杂性,提高调度的敏捷性、适应性和重构性.

## 2 基于 Agent 生产调度系统的结构 (Agent-based production scheduling system architecture)

现有 MAS 建模方法主要有基于任务的分解或基于区域的分解<sup>[7]</sup>.前者着眼于使每个 Agent 能完成一项特殊任务,如任务规划、通讯、实施某加工任务等;后者按地域、分布等进行划分,使 Agent 具有多智能,代理一个或多个物理实体或逻辑实体.作者通过基于功能的分解方法,即按照系统中的物理或逻辑单元的功能对系统进行建模.针对制造系统的最小功能实现,将整个调度系统分解成管理、资源、工件三个功能子系统,分别由不同的 Agent 类来描述.在此基础上,每个子系统再进一步分解成若干功能体,分别由该类 Agent 中的一些个体来管理和控制.于是,调度系统的基本构件包括:管理 Agent (MA)、资源 Agent (RA) 和工件 Agent (PA) 3 类.这些 Agent 并行工作,共同组成一个分布式系统.系统中,Agent 个体可依生产的规模和任务需要增加或取消,也可重构或重组.系统模型结构如图 1 所示.

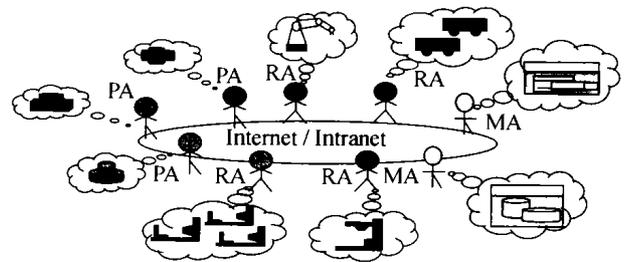


图 1 基于多 Agent 调度系统的结构

Fig. 1 Agent-based production scheduling system architecture

## 3 Agent 的模型和定义 (Structure and definition for agents)

### 3.1 Agent 的基本模型 (Model of agent)

鉴于敏捷调度问题的复杂性,所构造的 Agent 模型为复合式结构,如图 2 所示.每个 Agent 由若干功能构件组成,以便于重构和重组;它们基于黑板结构共享数据,在知识库的支持下,通过各构件间的交互和配合,实现预期功能.其基本构件有:

1) 黑板.作为 Agent 数据信息的载体,包括 4 种数据类型:

静态数据:Agent 的功能和能力参数.如某加工 Agent 的精度级别、加工类型等;

动态数据:当前状态信息.如某 PA 的当前加工状态参数;

结果数据:某一特定时期内的中期或最终结果数据.如 RA 完成任务的奖惩情况等;

任务列表:某一生产周期内的任务议程表,包括正在执行和待处理的任务等;

2) 知识库.具有 Agent 执行其功能所必须的知识,包括:

事实库:与调度有关的叙述性知识,如自身模型、相识模型和奖励模型等;

规则库:运行及合作时所需的产生式规则、专家控制表、调度策略;

算法库:调度所用的算法、程序和过程;

3) 行为控制构件.是 Agent 的“神经中枢”,负责 Agent 的行为决策、行为模式安排、必要准备和启动以及行为规则的制定等工作.

4) 自学习构件.通过对运行环境和运行结果的学习,对规则进行修正、补充,以适应环境需要.

5) 通讯协议构件.是遵循某些通讯协议的通用用户接口,实现与其它 Agent、人或设备之间通讯.

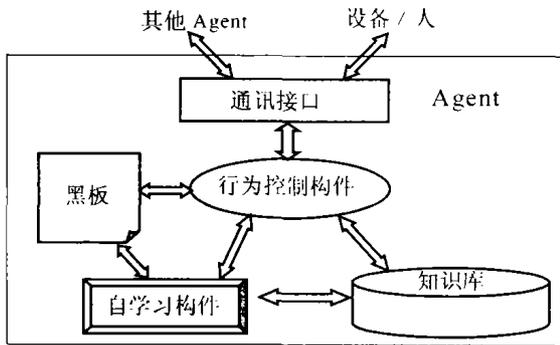


图2 Agent的模型结构

Fig. 2 Model structure for an agent

### 3.2 Agent类的定义(Definitions of agent classes)

**定义1** 系统管理 Agent(MA)——虚拟的生产调度特定功能管理者,具有对系统的资源、调度任务及其他 Agent 的管理与控制能力.按照其功能,定义—多元组表示一个 MA,即

$$\text{Manager} = \langle M\text{-id}, M\text{-att1}, M\text{-att2}, \dots, M\text{-att } n \rangle.$$

其中, M-id 是 MA 的标识, M-att1, M-att2, 分别代表 MA 具有的属性.系统所需 MA 个体的数量和各自任务范围,依据实际制造系统的规模大小来决定.例如系统内可设置任务管理 Agent(SMA),运行管理 Agent(EMA),监控 Agent(CMA)等.由于不同的 MA 各具有特定的功能和任务,因此其属性的内涵有所不同.例如, SMA 的属性可特化为

$$\text{Supervisor} = \langle S\text{-id}, S\text{-scale}, S\text{-order}, S\text{-relate} \rangle.$$

这里, S-scale 代表整个调度系统的规模信息. S-order 指示当前任务订单的信息, S-relate 表示该 MA 的关系客户或合作伙伴的信息.

**定义2** 资源 Agent(RA)——生产资源的代理,是生产订单的执行者,负责选择最优策略、监控资源的实时运行状况. RA 定义为一个四元组

$$\text{Resource} = \langle R\text{-id}, R\text{-bonus}, R\text{-task}, R\text{-state} \rangle.$$

其中, R-id 是 RA 的标识, R-bonus 是它的既往工作业绩的考核记录,作为其竞争任务的参考. R-task 是该资源的任务列表,表明在本调度周期内的任务状况. R-state 表示其当前生产状态.除 R-id 只在 RA 重构、重组时才可能变化外,其它属性值均随时间动态变化.

**定义3** 工件 Agent(PA)——工件任务的代理,其功能是在工件的生命周期内,根据任务目标和生产合同,生成完整的与 RA 相关的工件生产流程,为工件找到需要的生产资源. PA 定义为一个四元组

$$\text{Part} = \langle P\text{-id}, P\text{-data}, P\text{-process}, P\text{-state} \rangle.$$

其中, P-id, P-state 分别表示工件的名称(批次和号

码)和当前状态. P-data 引导该工件的数据文件,文件中包含工件的加工或装配类型、原料要求、几何尺寸、精度以及交工期等任务计划和工艺信息. P-process 表示该工件的工艺过程信息,可直接为系统资源所用. PA 的生命周期随工件任务进入系统而开始,随工件离开系统而终止.

## 4 分布式规则调度机制(Distributed-rules based scheduling mechanism)

由于调度系统的复杂性和动态性,试图通过全局集中规则来控制整个系统的行为,显然是困难的甚至是不可能的.但是,若分别为每个 Agent 建立规则,使之能够按照各自的局部信息和目标,同时兼顾系统的全局性能进行决策,将是一种有效的解决办法.为此,本文提出基于分布式规则的调度机制,实现 Agent 个体目标与系统全局目标的协调,以及分布式的整个系统行为控制.

### 4.1 分布式规则调度的概念(Concept for distributed-rules based scheduling)

分布式规则调度即在 MAS 体系结构支撑下,各个 Agent 并行工作,分别独立地建立和执行规则.就功能而言, Agent 的规则包括两部分,即 Agent 自行为规则以及 Agent 之间的互操作规则.自行为规则指导 Agent 基于一定的知识,进行独立决策;互操作规则使得 Agent 能够与其它 Agent 有效协作.于是,复杂的生产调度问题被分解成一系列单一对象、单目标的调度子问题,即 Agent 的自主行为调度或 Agent 之间的协调与合作行为调度.调度规则成为一个个模块化的规则组件,独立生成和执行,从而简化了规则构造和实施的复杂性,使调度问题得以容易、快速、有效地解决.

在这种调度机制的支持下,凭借 Agent 的智能性和自治性,调度规则对复杂问题的处理能力和对动态变化的适应性大大增强,并可以随 Agent 的重组而得以重构和重用.另外,当某些 Agent 的某些规则需要修改或补充时,只需要进行局部的变化,不会对系统内其它 Agent 和它们的规则发生影响.因此,分布式规则调度机制为 MAS 提供了一种简单、灵活、满足敏捷性和适应性需要的调度模式.

### 4.2 分布式规则的建立与执行方式(Ways of constructing and running for distributed rules)

同一类 Agent 的功能相似,只是在具体执行环境中,个体间的行为目标、策略有所不同,因此,系统首先按照代理类的特征知识,离线建立具有不同执行功能的类规则库.各个 Agent 启动后再对其类规

则进行继承和特化,建立自己的规则包。

Agent 不仅具有自治性、智能性,还具有交互性和合作能力,因而它们能够依据需要选择合适的行为规则,也可对既有规则进行修订、更新和补充;同时,还可按照一定的协议与其他 Agent 交互,实现对彼此规则的学习与继承。

### 4.3 Agent 之间的协调策略 (Coordination strategies between agents)

在同一个复杂环境中工作,各个 Agent 既要完成自己的行为目标又要兼顾全局性能,因此,Agent 之间的协调与合作显得尤为重要。本文设计了两类协调策略,即 Agent 之间的“自由协调”和在管理 Agent 指导下的“宿主协调”。协调过程在调度规则的支持与控制下进行。

通常情况下,Agent 之间采取自由协调方式,各 Agent 根据个体兴趣和目标的需要,寻求与其他 Agent 的协调与合作。当系统有特殊需要或出现 Agent 之间自由协商不能解决的问题时,则启动宿主协调策略。宿主协调策略的设计,使系统在逻辑结构上具有一定的递阶性,克服常规分布式结构过于松散、难以保证系统稳定性和全局性能的缺点。为此,系统赋予管理 Agent 一些特殊的能力,包括大局观、预测未来的能力以及管理与协调能力等,使其从系统的全局利益出发,为系统利益做出长远打算,协调其它 Agent 的行为,使系统在实现预期目标的前提下,充分发挥 Agent 个体的能力,协调、有序地稳定运行。

## 5 仿真实验 (Simulation studies)

### 5.1 实验环境 (System configuration)

仿真实验以某机床厂的机械加工车间为背景,侧重于多品种小批量工件的 jobshop 的敏捷调度。仿真实验原型系统中有车床、磨床、钻床三类机床,分别定义为 a, b, c 类,每类加工有若干不同型号的机床可供选择,设备情况如表 1 所示。

表 1 实验所用加工设备

Table 1 Machines used in the experiments

a 类			b 类			c 类		
a1	a2	a3	b1	b2	b3	c1	c2	c3
2	1	1	1	2	0	1	1	1

### 5.2 实验数据 (Experimental parameters)

设定此时系统内有 4 个工件加工订单,同时输入,其任务类型、工序、加工时间等信息描述如表 2 所示。

表 2 实验样本数据

Table 2 Sample data for the experiments

零件	批量	工序	加工类型	设备代码及对应生产时间		
Part 01	5	OP1	a	a1/06	a2/08	a3/10
		OP2	b	b1/11	b2/10	b3/13
		OP3	c	c1/07	c2/09	c3/11
		OP4	b	b1/04	b2/05	b3/07
		OP5	a	a1/10	a2/12	a3/14
Part 02	5	OP1	b	b1/05	b2/07	b3/9
		OP2	c	c1/12	c2/16	c3/19
		OP3	a	a1/16	a2/19	a3/21
		OP4	c	c1/03	c2/04	c3/06
		OP5	b	b1/5	b2/08	b3/10
Part 03	5	OP1	c	c1/12	c2/14	c3/16
		OP2	b	b1/04	b2/06	b3/08
		OP3	a	a1/08	a2/12	a3/16
		OP4	b	b1/10	b2/14	b3/18
		OP5	a	a1/13	a2/16	a3/19
		OP6	c	c1/15	c2/18	c3/22
Part 04	5	OP1	c	c1/12	c2/16	c3/20
		OP2	a	a1/16	a2/20	a3/24
		OP3	c	c1/04	c2/08	c3/12
		OP4	b	b1/12	b2/14	b3/16

工件按照工艺流程在机床上加工,加工后的工件由 AGV 小车运往下一工序。为简化仿真工作量,这里不对 AGV 进行调度,假定 AGV 有足够的力量按照需要完成所有工件的运送任务,并假设一个设备在某一时刻只能加工一个工件(任务)。在工艺规划中,各个工件任务在各个设备上的加工时间是确定的,所设定的生产时间中包括工件的安装时间、拆卸时间等,工件运输时间不记。任何一个操作不能抢先执行,即零件的加工工序是固定的。

### 5.3 实验结果 (Experimental results)

实验中,加工任务的开始调度时间为 8:00AM,交货期都设置为当天的 10:30AM。仿真的时间比率是 1:10。表 3 和图 3 是采用先进先出 (FIFO) 调度策略进行工件任务的分配和加工的实验结果。结果表明,尽管只使用了 FIFO 策略,但系统基本上能够完成预期的调度任务。

表 3 FIFO 策略调度实验的结果

Table 3 Experiment results obtained from the experiment which using FIFO strategy

	平均流动时间	加工完成时间	任务延迟率/%
Part 01	1 小时 18 分 07 秒	9 时 59 分 30 秒	0(0 个)
Part 02	1 小时 17 分 11 秒	10 时 2 分 30 秒	0(0 个)
Part 03	1 小时 20 分 46 秒	10 时 43 分 0 秒	40(2 个)
Part 04	59 分 53 秒	10 时 30 分 0 秒	0(0 个)
全部任务	1 小时 13 分 59 秒	10 时 43 分 0 秒	10(2 个)

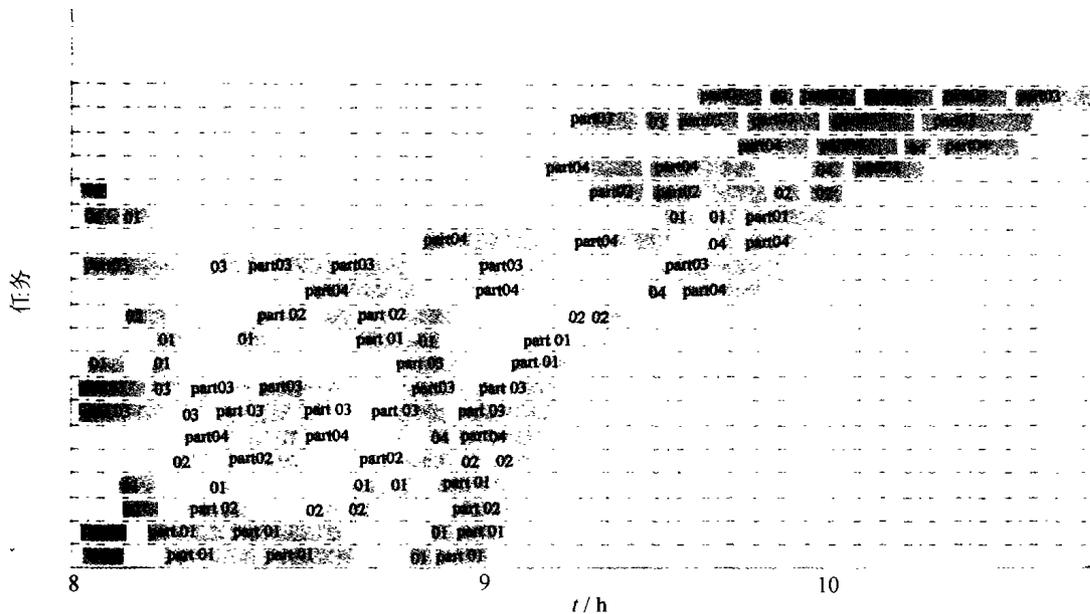


图 3 FIFO 策略仿真实验结果甘特图

Fig. 3 Gant chart obtained from the experiment which using FIFO strategy

对 RA 选择 SPT, PT + WINQ, PT + WINQ + SL 等<sup>[8]</sup>不同策略计算工件加工任务的排序优先级进行了仿真实验.结果表明当采取不同调度策略时,工件任务的平均流动时间、最大流动时间、延迟率等性能指标有所不同.例如,使用 SPT 时,任务的延迟率较低,而使用 PT + WINQ 策略时,任务的平均流动时间较优,实验结果与传统(单机)调度研究的结论<sup>[8]</sup>基本一致.所建系统的优点是能够边调度边加工,求解时间短、运行速度快(求解时间、通讯时间包含在工件的整个加工流动时间之中),能够满足调度的敏捷性要求.另外,针对急件订单的输入和设备突然损坏等情况的仿真实验也得到了较满意的结果,表明系统具有对紧急事件的处理能力,满足敏捷调度要求.

## 6 结束语(Conclusions)

基于多 Agent 和分布式规则的调度机制为实现敏捷生产调度提供了一条有效的途径.随着 Agent 和规则的分布处理,复杂的多目标、多任务敏捷调度问题转变成一系列简单的单一对象、单目标的调度问题,因而能够很容易地得到快速、有效的解决,满足敏捷调度的要求.由于建立在分布式的灵活框架之上,Agent 及调度规则可以方便地重构和重用,使得系统具有良好的扩展性和适应性.所建立的基于分布式规则的 Agent 之间的两类协调策略,克服了常规分布式系统过于松散、系统的稳定性和全局性难以保证等缺点,实现了系统全局性能与 Agent 个体目标的协调与优化.

提出的调度方法还有一些问题需要进一步解

决,其中包括分布式规则的完备性和合理性的证明方法等.另外,将研究结果应用于实际制造环境,以及推广到异地制造环境和虚拟企业也是作者下一步研究的目标.

## 参考文献(References):

- [1] RABELO R J, CAMARINHA-MATOS L M, AFSARMANESH H. Multi-agent-based agile scheduling [J]. *Robotic and Autonomous System*, 1999, 27(1-2): 15-28.
- [2] 蒋新松,张申生.敏捷竞争的挑战与思考[J]. *计算机集成制造系统 CIMS*, 1996, 2(2): 3-9.  
(JIANG Xinsong, ZHANG Shensheng. The challenge and ponderation on agile competition [J]. *Computer Integrated Manufacturing Systems*, 1996, 2(2): 3-9.)
- [3] MATURANA F P, NORRIE D H. Multi-agent mediator architecture for distributed manufacturing [J]. *J of Intelligent Manufacturing*, 1996, 7(4): 257-270.
- [4] SHEN W, XUE D, NORRIE D H. An agent-based manufacturing enterprise infrastructure for distributed integrated intelligent manufacturing systems [C]// *Proc of the Third Int Conf on the Practical Application of Intelligent Agents and Multi-Agents*. London, UK: [s. n.], 1998: 533-548.
- [5] SMITH R. The contract net protocol: high level communication and control in a distributed problem solver [J]. *IEEE Trans on Computers*, 1980, 29(12): 1104-1113.
- [6] 王艳红,尹朝万,张宇.基于多代理和规则调度的敏捷调度机制[J]. *计算机集成制造系统—CIMS*. 2000, 6(4): 45-49.  
(WANG Yanhong, YIN Chaowan, ZHANG Yu. Multi-agent and rule based dynamic scheduling system for agile manufacturing job shop [J]. *Computer Integrated Manufacturing Systems*, 2000, 6(4): 45-49.)

(下转第 536 页)

guarantees that the closed system is input-to-state stable and the  $L_2$ -gain from the disturbance input to the controlled output is not larger than a prescribed value for all admissible parameter uncertainties. Simulation results demonstrate that the controller designed by the above approach is feasible and effective.

### References:

- [1] SU W, XIE L, de Souza CARLOS E. Global robust disturbance attenuation and almost disturbance decoupling for uncertain cascaded nonlinear system [J]. *Automatica*, 1999, 35(4): 697 - 707.
- [2] POLYCARPOU M M, IOANNOU P A. A robust adaptive nonlinear control design [J]. *Automatica*, 1995, 32(3): 423 - 427.
- [3] SONTAG E D, WANG Y. On characterizations of the input-to-state stability property [J]. *Systems Control & Letters*, 1995, 25(2): 351 - 359.
- [4] ISHII C, SHEN T, QU Z. Lyapunov recursive design of robust adaptive tracking control with  $L_2$ -gain performance for electrically-driven robot manipulator [J]. *Int J Control*, 2001, 74(8): 811 - 828.

- [5] van der SCHAFT A J.  $L_2$ -gain analysis of nonlinear system and nonlinear  $H_\infty$  control [J]. *IEEE Trans on Automatic Control*, 1992, 37(6): 770 - 784.

### 作者简介:

朱永红 (1965 —), 男, 副教授, 2003年毕业于南京航空航天大学并获控制理论与控制工程专业并获博士学位, 现在江西景德镇陶瓷学院任教, 研究方向为鲁棒控制、自适应控制、智能控制等, E-mail: zyh\_patrick@tom.com;

姜长生 (1942 —), 男, 教授, 博士生导师, 1968年南京航空航天大学自控理论专业研究生毕业, 现在南京航空航天大学自动化学院任教, 研究方向为飞行器控制、鲁棒控制、自适应控制、智能控制等;

胡鸿豪 (1953 —), 男, 副教授, 1982年毕业于江西工业大学(现南昌大学)电子专业, 现在江西景德镇陶瓷学院任教, 研究方向为测量与控制等;

罗贤海 (1965 —), 男, 教授, 1989年西北工业大学工程力学专业研究生毕业, 现在江西景德镇陶瓷学院任教, 研究方向为机械设计及自动控制等。

### (上接第 530 页)

- [7] McDONNELL P, SMITH G, JOSHI S, et al. A cascading auction protocol as a framework for intelligent process planning and heterarchical shop floor control [J]. *Int J of Flexible Manufacturing Systems*, 1999, 11(1): 37 - 62.
- [8] JAYAMOHAN M S, RAJENDRAN C. New dispatching rules for shop scheduling: a step forward [J]. *Int J of Production Research*, 2000, 38(3): 563 - 586.

### 作者简介:

王艳红 (1967 —); 女, 副教授, 博士, 研究领域为控制理论与应用、先进制造、分布式信息处理与智能控制等, E-mail: wangyh@sut.edu.cn;

尹朝万 (1940 —); 男, 研究员, 博士生导师, 研究领域为控制工程与计算机应用、CIMS、分布式信息处理与协同制造等。