

文章编号: 1001-0920(2004)04-0388-05

知识化制造系统的任务分配决策

董 昊, 严洪森

(东南大学 自动化研究所, 江苏 南京 210096)

摘 要: 为解决知识化制造系统(KM S)的任务分配问题, 给出一种基于 Agent 的 KM S 体系结构, 提出一种特征控制的概念和一种基于 Tabu 搜索的优化算法. 在建立的数学模型基础上, 对特征控制的动态过程进行描述, 并给出算法的详尽步骤. 该算法和特征控制的结合使任务分配决策系统能够动态地响应任务和 Agent 的变化. 最后的仿真实例验证了该方法的有效性.

关键词: 知识化制造系统; 任务分配; 特征控制; Tabu 搜索

中图分类号: TH165; TB114.1

文献标识码: A

Decision-making of task distribution in knowledgeable manufacturing system

DONG Hao, YAN Hong-sen

(Institute of Automation, Southeast University, Nanjing 210096, China. Correspondent: DONG Hao, E-mail: zwang@seu.edu.cn)

Abstract: An agent-based system structure of the knowledgeable manufacturing system (KM S) is proposed to realize the decisionmaking of task distribution. The concept of character control and an optimization algorithm based on Tabu search are presented. Based on the mathematical model, the dynamic process of character control is described, and the detailed steps of algorithm are given. The integration of this algorithm and character control makes the decisionmaking system of task distribution respond to the variation of tasks and agents dynamically. The simulation results testify the validity of this method.

Key words: knowledgeable manufacturing system; task distribution; character control; Tabu search

1 引 言

知识化制造系统(KM S)是一种基于知识网的高智能制造系统. 它能够吸纳各种先进制造模式和理念, 即通过知识网将其转换为相应的先进制造知识, 并用此指导制造和组织生产. 知识化制造是 2000 年提出的新概念^[1], 并已受到关注^[2].

KM S 是一种以自适应、自学习、自进化、自重构、自培训和自维护为主要特征(6 自特征), 以 T (时间)、 Q (质量)、 C (成本)、 S (服务)、 E (环境)为主

要目标的高智能制造系统^[1]. 制造系统在运行中存在许多任务分配的问题. 例如, 制造企业由于产品多样化、多功能化和更新换代周期短等特点, 对于不同的产品设计任务, 要求有不同的设计 Agent 与之相适应, 以便能快速高效地设计出产品. 另一方面, 由于现代制造系统的运行环境充满了不确定性, 系统的制造任务经常是动态变化的. 如不可预知任务的增加和减少, 某些制造资源的紧缺和引入, 制造任务处理时间的变化等. 这些诸多的不确定因素, 要求制

收稿日期: 2003-04-21; 修回日期: 2003-06-09

基金项目: 国家自然科学基金资助项目(60443001); 江苏省高校“青蓝工程”第二期省级中青年学术带头人培养计划

作者简介: 董昊(1974—), 男, 江苏淮安人, 博士生, 从事知识化制造、并行工程的研究; 严洪森(1957—), 男, 浙江江山人, 教授, 博士生导师, 从事生产计划与调度、并行工程等研究

造系统的任务分配模块能动态地响应这些变化,使资源得到优化配置,达到制造系统运行的整体优化

为此,本文在面向 Agent 的 KMS 体系结构基础上,采用特征控制方式和一种基于 Tabu 搜索的算法来实现 KMS 各个子系统的任务分配决策

2 KMS 的体系结构

2.1 KMS 体系结构

KMS 可以看成是一个整体分布式自治和局部集中控制的系统,由若干个 Agent 群和一个 KMS 控制器组成。在整体上这些 Agent 群是分布自治的,在一定的通讯协议(TCP/IP or CORBA)下通过 internet/intranet 连接,在交互和合作中使制造系统正常运行并得到整体优化。Agent 技术是一种进行分布式工业系统建模的重要方法^[3],本文用它来构建知识化制造系统 KMS 控制器结构参见文献[1],本文不再赘述。KMS 控制器主要是在宏观上对 KMS 进行组织和对 KMS 的运行进行指导

2.2 Agent 群的内部结构

每个 Agent 群作为一个子系统,由多个功能 Agent 和一个控制 Agent 组成。功能 Agent 是能够完成一定功能的相对独立的实体;控制 Agent 在子系统内对各功能 Agent 进行集中控制。本文中,定义能独立完成如设计、工艺、制造或其他与企业生产相关的某类任务的功能实体为一个 Agent。例如,一个设计 Agent 应该包括设计人员、完成设计任务所需的设备,以及协同的工作环境。具有相似功能的 Agent 组成 Agent 群,每个 Agent 群除了各个功能 Agent 外,还有一个控制 Agent

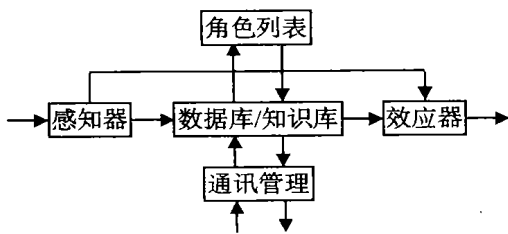


图 1 Agent 结构

图 1 为一种目前普遍接受的 Agent 结构^[4]。在本文中,每个 Agent 群包括功能和控制两种 Agent。在功能 Agent 的角色列表中包括投标模块和功能单元。投标模块在其所在功能 Agent 空闲且控制 Agent 招标进行任务分配时进行投标;功能单元表示的是 Agent 所具有的完成某类任务的能力。控制 Agent 角色列表包括任务分解、招标、评价和决策 4 部分。任务分解模块根据 Agent 群的当前工作情况

将任务进行分解;招标模块针对分解后的任务面向各功能 Agent 进行招标;评价模块对投标的各功能 Agent 进行能力评价和任务完成成本估算;决策模块根据评价结果进行任务分配的决策

3 KMS 的规划、调度与控制

制造系统的运行过程是:任务通知到达各 Agent 群后,控制 Agent 在任务分解后通过与群内所有功能 Agent 的招标-投标-评价决策-中标过程^[5]进行任务分配,之后各功能 Agent 开始运作完成分配到的任务

在各个子系统中各功能 Agent 具有相似的任务完成功能,可对各功能 Agent 采用某些共同的特征进行描述,只是因为个体的差异,这些特征值各不相同。另外,完成的任务也可用特征进行描述,对于某个子系统面对的任务,这些特征是共同的,所不同的是特征值的差异。因此,本文提出一种新的特征控制方式,如图 2 所示

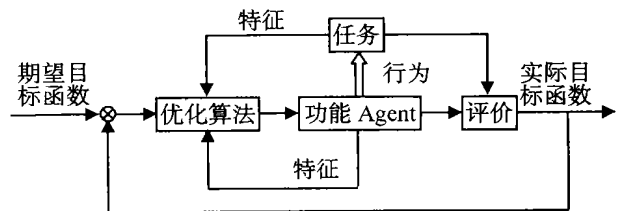


图 2 Agent 行为的特征控制

特征控制的核心是优化算法模块,控制的目标是使目标函数到达期望值。首先抽取影响目标函数的各功能 Agent 和任务的特征,把这些特征值作为参数送入优化算法模块,根据目标函数产生功能 Agent 的任务分配方案。评价模块在当前的分配方案下,根据目标函数对任务的完成情况和功能 Agent 的行为成本作出评价,得到实际的目标函数值

特征控制给出了一个 KMS 各子系统通用的控制模式。因特征控制中各个模块内容的不同,使其针对不同的子系统。例如,功能 Agent 在设计 Agent 群中由各个设计 Agent 组成,而在工艺 Agent 群中由各个工艺 Agent 组成。任务模块由子系统内具有相似特征的任务组成。优化算法模块针对各子系统求解问题的特点,可采用不同的优化算法。本文给出的是一种基于 Tabu 搜索的优化算法,与传统的基于 Agent 的完全分布自治的制造系统相比,引入特征控制的系统具有响应速度快和局部优化性能好的优点

特征控制中任务和各功能 Agent 特征的选取原则如下: 1) 完整性, 应选取对 Agent 群运行优化有影响的所有任务和功能 Agent 的特征; 2) 可操作性, 特征应易于评估和表示; 3) 非冗余性, 不同的特征因素应当是从不同方面对功能 Agent 和任务进行描述; 4) 可比性, 特征的确定应便于对不同的功能 Agent 和不同的任务进行区别

4 KMS 的任务分配决策

4.1 通过特征控制实现的 Agent 群内部任务分配

Agent 群作为一个子系统在接到任务通知时, 控制 Agent 对任务进行分解, 在与各功能 Agent 进行招投标后, 根据任务特征和各功能 Agent 的特征, 调用优化算法模块得到任务分配方案

任务分配决策应对两个方面的变化作出响应: 任务的变化和功能 Agent 的变化 任务变化——对任务完成时间和完成成本有影响的任务属性的改变; Agent 变化——改变 Agent 任务完成能力的情况发生, 如 Agent 故障、缺席或干扰等 通过特征控制可实现对这两方面变化的响应 任务和 Agent 的变化可反映在它们特征值的改变, 优化算法模块接收这些改变后产生新的任务分配方案以指导各功能 Agent 的行为, 响应上述变化达到系统的优化运行

4.2 数学模型

某制造 Agent 群内有一个控制 Agent 和 m 个制造 Agent 每个制造 Agent 有一个输入缓冲区和 3 个属性(即标志、执行速度和单位时间运行成本), 如表 1 所示 每个 Agent 在每一时刻只能执行一项任务, 如果 Agent 正忙, 则新分配的任务在输入缓冲区顺序存放并等待执行, 每项任务只需在一个 Agent 上完成

表 1 制造 Agent 属性

标志	执行速度	单位时间运行成本
Agent-1	10	65
Agent-2	10	70
Agent-3	40	38
Agent-4	10	48
Agent-5	10	55
Agent-6	40	60
Agent-7	10	140
Agent-8	10	110
Agent-9	10	55

一个任务单元到达制造 Agent 群, 经任务分解

后, 共有 n 个任务, 每个任务具有给定的任务长度 其完成时间= 任务长度/执行速度; Agent 的运行成本= 完成时间 \times 单位时间运行成本 当一个任务单元完成后, 另一个任务单元到达, 如此方式任务陆续到达、完成 如果任务数量 n 或者任务长度发生了变化, 则代表任务需求改变

特征控制的目标是形成任务在各制造 Agent 上的优化分配, 使在满足任务完成时间的条件下, 各 Agent 的运行成本尽可能小, 并且在突发事件(如任务需求变化、Agent 故障)发生时, 能够相应地自动调整分配方案

综上所述, 建立数学模型如下:

制造 Agent 的执行速度为 $v_i (i = 1, 2, \dots, m)$; 单位时间的运行成本为 $c_i (i = 1, 2, \dots, m)$; 任务长度为 $l_j (j = 1, 2, \dots, n)$. 得到任务 j 在 Agent- i 上的完成时间为

$$t_{ij} = l_j/v_i, i = 1, 2, \dots, m, j = 1, 2, \dots, n. \quad (1)$$

任务 j 在 Agent- i 上的运行成本为

$$c_{ij} = c_i t_{ij}, i = 1, 2, \dots, m, j = 1, 2, \dots, n. \quad (2)$$

令矩阵 X 中的元素 x_{ij} 表示任务 j 在 Agent- i 上的分配方案, 即

$$x_{ij} = \begin{cases} 1, & \text{任务 } j \text{ 分配给 Agent- } i; \\ 0, & \text{任务 } j \text{ 没有分配给 Agent- } i; \end{cases} \quad (3)$$

目标函数

$$\min F = w_1 \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} + w_2 \max_i \sum_{j=1}^n t_{ij} x_{ij}; \quad (4)$$

$$\text{s.t. } x_{ij} \in \{0, 1\}, \sum_{j=1}^n x_{ij} = 1. \quad (5)$$

式(3)~(5)中: $i = 1, 2, \dots, m; j = 1, 2, \dots, n;$ w_1 和 w_2 是目标函数中成本因素和时间因素的权重

这是一个组合最优化问题, 是 NP 困难的并且其可行解的数量不超过 2^{mn} .

4.3 特征选取

特征选取根据前面所述的原则, 确定如下: 任务特征: 任务数 n , 任务长度; 功能 Agent 特征: 执行速度, 单位时间运行成本

4.4 对任务和 Agent 变化的响应

如前所述, 要求系统在突发事件下, 能够自动调整分配方案 如任务需求的变化, 反映在任务特征上就是任务数 n 或任务长度的改变 而 Agent- i 发生故障时可将其执行速度设置为一个接近于 0 的正值, 使其对任务的完成时间和运行成本都趋于极大

特征控制在上述突发事件发生时, 任务特征或

Agent 特征的改变触发优化算法模块, 重新对分配方案状态空间进行搜索, 找到满意解

5 基于 Tabu 搜索的优化算法

Tabu 搜索方法是由 Glover F 提出的, 其优点是提供了一个非常灵活的框架, 可方便地把一些知识和规则通过 Tabu 表中的元素体现出来, 实现搜索的集中性和广泛性. Arts 等人针对作业调度 (job shop) 问题, 比较了模拟退火算法、遗传算法和 Tabu 搜索等算法的性能, 得出 Tabu 搜索算法在最短的时间内给出了最好的结果^[6]. 在其他的一些问题上, Tabu 搜索也取得了较好的结果. 因此本文在特征控制的优化算法模块中采用基于 Tabu 搜索的算法

5.1 Tabu 搜索关键词定义

定义 1 (移动) 分配给 Agent_p 的任务 q 重新分配给与 Agent_p 相邻的 Agent, 而其他 Agent 承担的任务不变, 称作 X 的一次移动. 即在以横向为待分配任务、纵向为各功能 Agent 张成的任务分配矩阵 $x_{ij} (i = 1, 2, \dots, m; j = 1, 2, \dots, n)$ 中, 若 $x_{pq} = 1$, 则

$$\begin{cases} x_{mq} = 1, x_{pq} = 0, & \text{若 } p = 1, \\ x_{p-1,q} = 1, x_{pq} = 0, & \text{若 } p = 2, 3, \dots, m, \\ x_{ij} = x_{ij}, & j \neq q, \end{cases} \quad (6)$$

称作 X 关于任务 q 的一次向上移动;

$$\begin{cases} x_{p+1,q} = 1, x_{pq} = 0, & \text{若 } p = 1, 2, \dots, m - 1, \\ x_{1q} = 1, x_{pq} = 0, & \text{若 } p = m, \\ x_{ij} = x_{ij}, & j \neq q, \end{cases} \quad (7)$$

称作 X 关于任务 q 的一次向下移动

定义 2 (邻域) X 进行一次移动后得到一个新的任务分配矩阵, 当前任务分配矩阵 X 的邻域 $N(X)$ 定义为 X 的所有可能单次移动后产生的任务分配矩阵构成的集合

定义 3 (禁忌表) 为了避免循环和陷入局部最优, 引入一定长度的禁忌表 T , 表中记录了最近进行的若干个移动, 在当前的迭代步骤中, 这些移动是被禁止的

定义 4 (特赦规则) 在当前的迭代步骤中, 如果候选解虽然在禁忌表 T 中, 但候选解的目标函数值低于目前为止所搜索到的最小值, 则禁忌约束无效, 选取该候选解作为当前解

5.2 初始可行解的获得

Tabu 搜索是一种迭代算法, 一个好的初始解可以提高搜索速度. 为减少计算量, 不考虑运行成本而只按照完成时间得到初始可行解. 首先分别对任务

和 Agent 根据任务长度和执行速度由高到低进行排序, 按照较高长度的任务分配给执行速度较高的 Agent 来完成的原则, 取出经排序后的前 m 个任务进行分配, 结束后每个 Agent 接受到一个任务; 再取出从第 $m + 1$ 个任务开始的 m 个任务按照上述原则进行分配, 如果不足 m 个则只分配取出的任务; 依次进行直到所有任务分配完成. 这样就得到了初始可行解

5.3 算 法

算法 1 基于 Tabu 搜索的优化算法如下:

Step 1: 初始化: 定义 Tabu 表 T 长度 T_Length 和最大迭代次数 $iter_Max$;

Step 2: 读入任务特征值和 Agent 特征值; 按照 5.2 的方法得到初始可行解; 设置迭代次数 $iter = 0$;

Step 3: $iter = iter + 1$; 若满足终止条件 $iter = iter_Max$, 转 Step 5;

Step 4: 对当前可行解 X , 寻找 $N(X)$ 中的最好解 X^* . 若 X^* 不在 T 中, 以 X^* 作为当前解, 转 Step 5; 若 X^* 存在于 T 中且满足特赦规则, 解禁 X^* 并以之为当前解, 转 Step 5; 否则在 $N(X)$ 中去掉该点, 重复 Step 4;

Step 5: 更新 T , 把 X^* 置于 T 顶部, 并且若 T 长度大于 T_Length , 则删除 T 底部最老的一个元素, 返回 Step 3;

Step 6: 结束, 返回满意解和目标函数值

6 仿真实例

某制造 Agent 群内有 1 个控制 Agent 和 9 个制造 Agent, 制造 Agent 的属性如表 1 所示. 一个任务单元到达制造 Agent 群, 经任务分解后, 共有 20 个任务, 任务长度如表 2 所示

表 2 任务属性

任务	1	2	3	4	5	6	7	8	9	10
任务长度	40	40	40	40	40	40	160	160	160	160
任务	11	12	13	14	15	16	17	18	19	20
任务长度	160	160	160	160	160	160	160	160	160	160

6.1 仿真结果

采用 5.3 的优化算法, 并作如下设置: 目标函数中, $w_1 = 1, w_2 = 1000$; T 的长度 $T_Length = 50$; $iter_Max = 1000$

经过 1000 次迭代后可得到一个满意解 (见表 3). 满意解是在第 53 次迭代中获得的, 在该满意解的分配方案下, 目标函数为 28008. 算法用 C 语言编写, 在一台有 256M 内存的 Pentium IV 1.2 GHz

计算机上运行,总搜索时间为 4.12 s

表3 满意解下的分配方案

任务	Agent								
	1	2	3	4	5	6	7	8	9
1	0	0	0	0	0	1	0	0	0
2	0	0	0	0	0	1	0	0	0
3	0	0	0	0	0	0	0	0	1
4	1	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0
6	0	0	0	0	0	1	0	0	0
7	0	0	1	0	0	0	0	0	0
8	0	0	0	0	0	1	0	0	0
9	1	0	0	0	0	0	0	0	0
10	0	0	1	0	0	0	0	0	0
11	0	0	1	0	0	0	0	0	0
12	0	0	0	0	0	1	0	0	0
13	0	0	0	0	0	1	0	0	0
14	0	0	0	0	0	0	0	1	0
15	0	0	0	0	0	0	0	0	1
16	0	0	1	0	0	0	0	0	0
17	0	0	0	0	0	1	0	0	0
18	0	1	0	0	0	0	0	0	0
19	0	0	1	0	0	0	0	0	0
20	0	0	0	1	0	0	0	0	0

6.2 对任务特征变化的响应

假设在完成一定数量的任务单元后,新的任务单元达到,经任务分解后共有 18 个任务,任务数量和部分任务长度都发生了变化,如表 4 所示。任务特征的改变启动优化算法模块,读入任务特征值和 Agent 特征值,获得初始可行解后重新进行搜索。经过 4.88 s,54 次搜索后得到新的分配策略(略),目标函数为 21.01。

表4 任务变化后的特征

任务	1	2	3	4	5	6	7	8	9
任务长度	40	40	40	40	40	60	60	60	60
任务	10	11	12	13	14	15	16	17	18
任务长度	160	160	160	160	160	160	160	160	160

6.3 对 Agent 故障的响应

在原来的条件下,若 Agent-6 发生故障,优化算法模块接到消息后,把 Agent-6 的执行速度设置为 $1.0e^{-6}$,其他条件不变,重新对解空间进行搜索。经过 5.22 s,在第 25 次迭代时得到新的满意解

(略),目标函数为 39.124。这样系统就找到了 8 个 Agent 正常工作情况下新的分配策略。

7 结 语

本文在知识化制造这一新理念基础上,给出一种基于 Agent 的 KMS 体系结构,并提出特征控制的概念和一种基于 Tabu 搜索的优化算法,解决 Agent 子系统内部的任务分配问题。最后给出一个包括 9 个功能 Agent 和 1 个控制 Agent 的仿真系统,分析了系统内的任务分配决策过程,并在任务变化和某个功能 Agent 发生故障时,进行响应以得到新的分配方案。仿真结果验证了本文方法的有效性。

参考文献(References):

- [1] 严洪森,刘飞. 知识化制造系统——新一代先进制造系统[J]. 计算机集成制造系统, 2001, 7(8): 7-11.
(Yan H S, Liu F. Knowledgeable manufacturing system —— A new kind of advanced manufacturing system [J]. *Computer Integrated Manufacturing Systems*, 2001, 7(8): 7-11.)
- [2] 范玉顺. 中国运用现代集成制造技术改造传统产业的经验 and 前景[J]. 制造业自动化, 2002, 24(4): 1-7.
(Fan Y S. Experience and prospect of using CMS promote the manufacturing industry of china [J]. *Automation Panorama*, 2002, (3): 1-7.)
- [3] 刘金琨,尔联洁. 多智能体技术应用综述[J]. 控制与决策, 2001, 16(2): 133-140.
(Liu J K, Er L J. Overview of application of multi-agent technology [J]. *Control and Decision*, 2001, 16(2): 133-140.)
- [4] 乔兵,朱剑英. 多 Agent 智能制造系统研究综述[J]. 南京航空航天大学学报, 2001, 33(1): 1-7.
(Qiao B, Zhu J Y. Agent-based intelligent manufacturing systems: A state-of-the-art survey [J]. *J of Nanjing University of Aeronautics & Astronautics*, 2001, 33(1): 1-7.)
- [5] Sikora R, Shaw M J. Coordination mechanisms for multi-agent manufacturing systems: Applications to Integrated Manufacturing Scheduling [J]. *IEEE Transactions on Engineering Management*, 1997, 44(2): 175-187.
- [6] 张鸿宾. 组合优化问题的启发式搜索[J]. 计算机科学, 1998, 25(2): 13-16.
(Zhang H B. Heuristic Search in Combinatorial Optimization [J]. *Computer Science*, 1998, 25(2): 13-16.)