基于多Agent的柔性作业车间动态调度系统的设计与研究

随着全球制造化和信息化的普及，制造业面临着技术革新和竞争加剧带来的多重挑战，面对分布式的制造资源、随机订单、频繁的订单更改以及设备故障等种种动态多变的因素，制造企业需要迫切地提高自身的管理和服务水平。调度系统作为车间调度管理的重要模块，需要综合考虑多方因素，包括对制造资源、产品工艺的管理，订单调度方案的生成以及对诸如设备故障等异常因素的响应等等，系统的架构设计和调度策略、重调度策略的配置等对构建一个具有自主性、自适应性、智能化的分布式制造环境有着至关重要的作用。

现代车间调度系统趋向于智能化、自动化、信息化，而多Agent理论作为人工智能领域的分支，通过对大型复杂任务进行分解，针对每个子任务设计对应的Agent，构建多Agent系统，系统通过各Agent之间的交互协调来快速灵活地完成任务。本文在车间动态调度的研究中应用了多Agent理论，设计了一种基于多Agent的车间动态调度系统，旨在提高制造业生产效率的同时，提高车间制造的稳定性。

1. 绪论

1.1 研究背景

（经济全球化—竞争激烈—生产调度系统很关键—学者研究：传统/新型—提出Agent）

（生产调度关键—生产环境充满不确定性—提出分布式人工智能—提出从调度系统的角度解决问题）

（自己的：提出当前制造业面临的困难（竞争激烈、生产模式、分布式、动态不确定性）--生产调度很关键—当前算法和系统的不足—提出多Agent调度系统）

我国现代制造业面临着愈来愈激烈的竞争，一是由于经济全球化的影响，制造企业需要在全球化舞台上跟来自世界各地的企业竞争，这带来了更大的市场和机遇的同时，也要求企业在制造和管理上需要作出变革，适应全球化的竞争潮流；二是随着经济的发展、生活水平的提高，人们的消费观念和习惯有了很大的转变，对多样化和个性化的商品给予了更多的关注，因此商家需要根据市场需求不断推陈出新，调整经营策略，更换出售商品的种类和结构，对应的，制造企业在生产模式上也发生了很大的转变，传统的“小品种大批量”生产模式已被逐渐摈弃，取而代之的则是面向订单的小批量多品种甚至是单件生产。这导致了制造业传统的面向大批量的长时间跨度的静态生产调度方案已经不再适用，而是要充分考虑市场需求的动态变化，能够灵活响应随机订单、紧急插单等情况，同时要快速生成满足交货期的调度方案。除此之外，制造业所处的生产环境充满不确定性，面临包括急插单、原料紧缺、设备故障等多种异常因素，这对制造业在快速调整调度方案，尽可能满足订单的交货期、保证系统稳定性上无疑有了更高的要求。

为提高我国制造业的发展水平，除了需要在生产技术和设备配置上进行改进外，还需要在生产管理上进行优化，而生产调度作为车间制造的核心，一直是制造领域的研究热点。相比于渐趋成熟的生产技术，生产调度涉及的改进所需成本和人力更小，且效果同样显著，因此吸引了众多研究人员的关注。尽管如此，生产调度依旧是制造领域的难点，这是由于其涉及的方面广，问题规模庞大，异常因素多且解决难度大，因此至今仍未有十分成熟通用的调度系统能够广泛应用到制造业中。

传统的生产调度系统属于集中控制式的静态调度系统，特点是制造资源集中、生产规模庞大、时间跨度长等，主要面向种类单一、工艺结构简单的产品制造，该形式的调度系统简单地把管理、控制、监控等功能交由人力负责，因此容错能力和稳定性低，当生产过程中发生如原料短缺或设备故障等情况时，往往需要对整个生产线进行停工处理，显然集中控制式的静态调度系统是无法满足现代制造业的要求的。现代制造业使用较为广泛的调度系统为递阶控制式的调度系统，属于智能控制和动态调度理论的早期成果，该形式的调度系统的特点是分层管理，决策单元位于上层，对下层单元进行管辖和监控，信息传递通常由上层经过多层迭代传输至目标单元。对比集中控制式的调度系统，强化了管理控制职能和信息反馈速度，但是对于规模庞大的复杂系统而言，递阶控制式意味着系统的层次会过深，各层次间单元的通讯需要经过多次中转处理，灵活性较低，效率不高，这将导致系统在应对环境的动态异常因素时不够灵敏。为克服以上不足，同时还需要考虑到当今大多数企业制造资源的分布式特点，即一家企业在世界各地设有工厂车间，如何综合利用不同地理位置的制造资源和生产技术，对提高制造业的生产效率和资源利用率是非常关键的。由此，制造业的生产调度系统更趋向于向智能化、信息化的方向发展，同时应强调系统的分布式协调和控制能力，提高系统的自适应能力和稳定性，从而保证系统的容错能力。

近年来，与Agent相关的理论在制造领域一直备受关注，主要原因在于单个的Agent能够自主地获取来自外界的信息，感知周边环境的变化，从而根据自身需求作出反应。每个Agent都被赋予一定的学习能力，这取决于Agent的计算能力和配置的决策规则，在与外界环境交换信息的同时，能够收集自身需要的数据，强化自主学习能力。考虑把多Agent理论应用到车间调度系统上，根据生产管理中各个流程的结构和功能，把其中的物理实体或算法逻辑封装为具有不同功能和结构的Agent，构建多Agent系统，通过Agent之间的交互协调，能够快速灵活地完成任务，Agent的这些特点，能够很好地满足车间调度系统在灵活性、自适应性、分布式上的需求。

目前多Agent理论已经较为成熟，但能成功应用多Agent理论并正式投入市场使用的车间调度系统仍很少，这主要是由于调度问题涉及方面广，难度大，因此如何把理论与实践相结合，构建满足市场需求以及实际生产需要的调度系统，仍然是广大研究人员急需解决的一个难题。本文的研究重心在于综合考虑现代制造企业的分布式资源，所面临的动态变化的生产需求以及干扰正常生产的异常因素，应用多Agent理论设计一个分布式的动态车间调度系统。

1.2 车间调度问题的描述和分类

车间调度问题可以描述为在生产资源一定的前提下，如何把这些资源分配给特定时间范围内的不同生产任务，以达到完成一个或多个目标的目的[1]。更具体地，车间调度问题可以表示为把n个工件安排在m个设备上进行加工，其中每个工件从加工到完成需要经历若干道工序的加工，工序间存在顺序约束；每个设备能够对若干道工序进行加工，对应不同的完成时间，而调度目的就是如何定义设备的分配、工件工序的加工顺序以及工序的开始和结束时间，以达到一定的优化目标，因此车间调度问题的实质是全局考虑若干生产任务的约束条件，按照一定的策略制定资源的分配以及任务的生产顺序，最终满足生产任务在一个或多个目标上的期望值。这里的约束条件主要由生产环境以及生产任务的特点所提供，例如生产设备的加工能力，有些设备只能对单一工序进行加工，有些设备能够对若干工序进行加工，对应不同的完成时间；生产任务上的约束主要体现在工艺约束上，存在生产工艺固定的工件，也存在生产工艺可选的工件，后者在工序的前后顺序选择上存在多个可选分支，并且每道工序可以在多个设备上进行加工，对应不同的完成时间，因此具有更多的柔性。在车间调度的目的上，存在多种优化目标，包括在满足交货期的前提下，最小化最大生产时间、成本最低或者在可容忍任务延期的情况下，最小化延时总和等等。

根据制造业生产系统的特点，车间调度可以分为以下几种类型：单机调度、多机调度、流水车间调度、作业车间调度。单机调度是最简单最早期的调度种类，所有工件的生产加工都由一个设备来完成；多机调度是相对于单机调度而言的，区别体现在生产设备的数量上更多，单一工件的加工同样只需要由一个设备来完成，但是由于相同功能的设备更多，因此效率上更高。单机和多机调度灵活性较低，不存在柔性，属于早期生产力较低下的调度方式，已不适用于现代制造业；流水车间调度和作业车间调度细化了生产流程，强调设备的职能分工，对于工件中的每一道工序都会分配特定的设备来进行加工。其中流水车间调度中所有工件都有相同且固定的加工工艺，每个工件有着严格的工序顺序约束，需要按照特定的顺序经过每一个设备，这些设备就组成一条流水线，若存在功能相同的设备可并行运作，则其中的工序可选择任意一个设备，可选择的设备数量越大，则系统具备的柔性越多；而作业车间调度中，每个工件的工艺路径可以是不同的，工件与工件间也不存在排队间的顺序约束，由于可能存在多个可选工艺路线的情况，甚至即使是同一种工件，在设备选择上也存在多种形式，因此当设备规模较为庞大、工件加工工艺复杂时，作业车间调度可看作是柔性作业车间调度。

流水车间调度在上世纪八十年代以来一直是众多车间工厂最主要的调度方式，尤其适用于大批量小品种的长期静态生产，极大地降低了人力成本，使生产力得到高速提升。但由于现代市场商品生命周期更短，需求动态变化，制造业的生产模式逐渐转向面向订单的小批量多品种生产模式，因此具备更高灵活性和更多柔性的作业车间调度已逐渐成为主流。加上生产环境中存在订单随机、紧急插单、资源短缺、设备故障等诸多异常因素，车间调度要具备快速响应能力，本论文的研究重心正是面向柔性车间调度的动态调度系统。

1.3国内外研究现状

自1954年，Johnson开展对基于两台机床的flow shop型调度问题的研究后，便逐渐引起实践人员的广泛研究。但随着时代的变迁以及生产力的发展，车间调度的种类越来越多，从最原始的单机调度、多机调度过渡到广泛的流水车间调度和作业车间调度，对应的调度模型越来越复杂，难点越来越多，因此关于车间调度的大部分研究是针对某个方面展开的，例如调度算法、重调度算法的优化、调度系统结构的调整等等。

在研究方法上，最初是集中在整数规划、多目标优化、仿真和简单规则上，这些经典的调度方法在应用中存在很大的局限性，如难以建立准确约束条件下的数学模型，并且求解最优解时间是随问题规模呈指数成倍增长的NP一hard型，因此只能对小规模的系统求解。近年来，随着各种新的相关学科与优化技术的建立与发展，出现了许多新的优化方法，如神经网络、遗传算法、基于知识的方法、基于离散事件系统(DED)S的解析模型和方法等。值得关注得是，人们将这些优化方法有机地结合起来，如遗传算法与调度规则、仿真与基于知识的方法等结合，在一定程度上达到了一种较为科学有效的调度效果。  
但是随着顾客需求的个性化及要求企业响应市场的敏捷化，往往在生产加工过程中加入了更多的不确定性及复杂性约束，这些使得上述的调度思想难以适合敏捷化制造的要求。

近年来，随着对DAI和先进制造系统的进一步研究，表明拥有较高的智能性、灵活性和良好的合作能力的Agent之间彼此协作的分布式制造系统可以快速地响应环境变化。多Agent放松了对集中式、规划和顺序控制的限制，提供了分散控制、应急和并行处理，提供更快速的问题求解。这样，制造系统便可以通过分布在车间里的资源和个体Agent之间的联合来实现系统的集成。近些年来，各国纷纷将Agent理论和多AGENT技术应用于制造车间调度问题的研究上，己经取得了一定的研究成果，建立了各种各样基于Agent的调度原型或仿真系统。

国外的研究机构已展开了对基于Agent调度系统的研究，日本、美国、和欧洲在DIA及其应用研究处于世界前列。Hadvai等出的分布式实时调度的多Agent系统结构REDS;Baker等研究了包括调度在内的工厂控制算法在分布分层结构的多Agent制造环境中的实施问题；Shaw提出用Agent方法进行制造调度和控制，指出制造单元可以通过投标机制，将作业以子合同的形式转包给其它制造单元;Chen等采用Agent技术进行制造调度与供应链协调研究;Butler等提出了用于分布式动态调度的多Agent体系结构，将调度分为两个层次，第一层通过Agent的协商机制，以一种分布式的方法制造单元分配给作业，第二层对共享制造资源进行动态分配;这些系统都引入了Ageni概念对制造车间生产过程加以分析，但并没有一个切实可行的基于多Agent的调度原型系统。

Norrie等列举了近30个国际上已经开展或正在进行的运用Agent技术进行制造规划、调度及控制的研究项目。从这些研究项目可以看出，运用Agent技术进行制造规划与调度等问题的研究，其常见的做法是用Agent来表示制造中的各种物理资源或逻辑资源，如机床、物料运输设备、机器人、订单、BOM等，并根据制造过程的组织形式，通过网络及Agent通讯协议将它们连接成一个多Agent系统，然后，采用多Agent系统理论和方法对Agent的协作、协商、调度、控制、冲突消解、智能显现等行为进行研究。

在1994年的一项国际合作研究项目“智能制造系统研究计划”提出的六个研究方向当中，合弄制造系统HMS是比较有影响的一种思想，它偏重于利用人工智能技术研究新的未来灵活制造模式，具有合作性、自治性和智能性。在具体实现方式上与分布式人工智能中Agent十分相似，目前的HMS的实验系统全部是借助于多Agent系统技术实现的，研究尚处于起始阶段，还无法在大中型制造系统中实际应用。虽然HMS的自治性及合作能力有赖于DAI技术的支持，但Agent建模协作机制在其中并没有太多体现。

国内的对Agent的研究近年来也相继开展开来，尤其是大学己展开这方面的研究。华中理工大学在国家自然科学基金重点资助项目(5935090)支持下，成功地以数控加工系统为背景，开发了一个基于Agent分布式网络化IMS调度原型系统，实现了制造单元的柔性智能化与基于网络的制造系统柔性智能化集成。此外，沈阳自动化所在多智能控制器研究与设计中，把Agent及多Agent模式应用于多机械手、多机器人的协调与协作设计中，开发了基于多Agent的多机器人协调调度系统，已取得了阶段性成果。我国“八五”期间也有一些高校和研究机构如清华大学、上海交通大学、西安交通大学、北京机械工业自动化研究所等进行此类问题的研究，并己开发出相应的计算机宿主生产调度与管理系统，逐步从理论研究阶段走向应用阶段。这表明国内越来越多的研究人员开始进驻Agent技术和应用制造系统调度问题的研究。

回顾以往的研究历程，我们发现在早期的研究中，生产调度往往被抽象描述为一个有性能指标、有约束条件、寻求最优解的数学问题。但由于其实质上是一个NP难题，即随着问题规模增大，寻求最优解的计算量呈指数增长，使得一些常规得优化方法难以奏效。因此，激发了很多研究者致力于“优化调度”的研究，试图解决调度算法的求解速度和求解质量等问题。

随着研究的深入和制造系统的发展，特别是柔性制造系统、计算机集成制造系统的应用，实际生产系统的复杂性和动态性增加，促使人们进一步研究能够处理生产环境的不确定性和意外时间的生产调度技术，使得调度目标变成柔性调度。可见,面对现代制造的复杂要求,如果依然遵循传统的调度研究方法和思路,单纯通过研究某些算法的改进和扩展等努力来解决,是达不到预期的效果的,需要从系统的角度出发,采用新的思路研究理想的调度系统模型、合理的调度系统结构以及能够集成经典的调度方法的先进调度策略和方法。这是生产调度系统适应未来制造需要的必然发展趋势,也是制造控制理论和技术发展的必然趋势

2 基于多Agent的柔性作业车间调度系统结构的设计

一个车间调度系统是否具备实用价值，可以从系统的通用性去衡量，即调度策略能否广泛应用于多种调度模型，当出现新的调度模型时，调度算法和重调度算法是否可以无缝替换甚至只需少量修改便可重新适应；系统在产品信息、工件工艺约束的表现形式能否适用于各种产品；系统能否快速响应工艺信息、设备节点信息的增删查改等等，这些都要求系统能在各类型的生产环境下均能保持高稳定性、可适应性和可扩展性，因此关键在于对系统功能的划分和结构的设计是否合理，本章将主要阐述Agent和多Agent的基本理论，同时根据调度系统在订单任务输入、调度和重调度、故障检测等多个方面对系统的结构进行分析，最终应用多Agent理论设计分布式的动态调度系统。

2.1 Agent与多Agent理论

2.1.1 Agent理论

尽管Agent在众多研究中应用非常广泛，在理论上已渐趋成熟，但目前对Agent并未有统一的标准。Agent通常也被称为智能体，属于人工智能领域，因此人们关注的主要是Agent的自主性，从这个角度而言，Agent比较主流的定义是Wooldrige和Jennings提供的：Agent属于由硬件和软件两者综合组成的实体，能够和所处的环境进行信息交换，并能够按照特定目标自主采取相应的行动。Agent通常也被认为是“可移动的”，能够根据自身的需求并在在外界环境信息的刺激下作出相应的判断，改变自身的位置，还能够影响周边环境以及其他Agent。尽管现在Agent尚未大范围推广，实际的应用实例也比较少，但是Agent具有大家所公认的以下一些特点：

1. 自主性：拥有支撑自身行为的决策库，具备学习能力，可根据与周边环境的接触扩充决策库，根据决策规则选择对应的行为，可连续动态地改变自身的状态。
2. 社交性：具有通信标识和通信语言，Agent之间具有通信的能力，遵循特定的通信规则可相互交换信息。
3. 反应性：能够时刻感知来自外界的信息而无需特定的信息进行唤醒，能够分析信息的种类和价值，把有用的信息扩充自身数据库。
4. 能动性：不仅能接收外界的信息，同时还能够发起对周边环境的探索，通过主动获取环境信息以及接收其他Agent的信息更改决策库以及位置。

可见，Agent有一定的智能和学习能力，能够根据自身的目的获取、交换信息，从而更改自身的状态，包括决策库的决策规则、数据库信息、地理位置等等。Agent的学习能力主要有决策库提供，库中配备的决策规则的完善程度决定了Agent学习能力的强弱，并由此可分为三种类型的Agent：

1. 慎思型：需要人工智能领域发展到较高阶段，赋予Agent以较高的智能，对应的Agent具备最强的学习能力，拥有对自身和环境的认知能力。该认知能力关键在于能够根据所接受的信息、行动导致的结果与目标的一致性来调整所配备算法、规则的相关参数，从而改变后续的行为模式，提高目标的完成度以及效率。因此慎思型Agent需要由强大的推理决策能力，无需人工干预。
2. 反应型：不具备或者具备很低的学习能力，在功能上更偏向于是一个程序编程接口，内部封装了提供各种服务的函数，简单地根据决策库中定义的规则来决定自身的行为，运行过程中不会涉及对原有规则的优化，较为被动地等待接收外界的信息请求。
3. 混合型：慎思型拥有最强的学习能力，但运行过程中需要结合多方面的信息和规则来进行推理，从而决定自身行为，因此需要更多的计算时间；反应型学习能力低，但是由于行为模式已经事先确定，因此处理速度快。混合型就是结合了两种Agent的优点，兼顾了计算能力以及处理速度，在本文所设计的Agent属于混合型Agent。

根据以上Agent的描述，可见Agent不仅具有学习能力，同时具有可移动性，在分布性和扩展性的环境中易于迁移，因此由若干Agent组成的多Agent系统非常适合用于部署分布式系统，对于分布式系统中的各个功能模块或子系统，可以交由单独的Agent或者由若干Agent组成的Agent模块负责，通过多Agent系统间Agent的信息交互，来协同完成各项任务。

2.1.2 多Agent系统的概念

多Agent系统是由Agent组成的具有一定结构和功能的分布式系统。其中的Agent需要承担系统中的部分功能，功能相似或相同的Agent可组成一个Agent组。系统需指定Agent间的通信协议和通信方式。

多Agent系统能够对复杂问题进行分解，分配到相关的Agent，各Agent根据自身的计算能力求解子问题，通过Agent间的通信进行交互协调，最终获取对复杂问题的完整求解结果。与传统集中控制式、递阶控制式的系统而言，多Agent系统部署和维护所需要的成本更低，系统将计算能力和资源分散到一组Agent中，单一的Agent在子问题的计算中对硬件性能的要求较低，减少了对高性能的大型服务器的依赖。同时，Agent能独立于中央服务器运行，能并行处理消息请求以及进行问题求解，因此能有效提高系统整体的稳定性和效率。

Agent的自主性、反应性、能动性等决定了多Agent系统具有更强的灵活性，对外界的变化反应更为灵敏，在环境动态变化频繁的场景中具有更大的优势。Agent作为分布式系统中的节点，由于其具有可移动性，因此对于节点的增删处理反馈更及时，保证系统的高可扩展性，再者Agent的学习能力能够调整自身的状态和行为，随着时间的推进，可以优化求解过程、提高系统整体的求解质量，这相当于延长了系统的生命周期，减少人工干预、维护的成本。

2.1.3 多Agent系统的结构

多Agent调度系统一般存在两种较广泛使用的系统结构，分别是层次型和分布型：

（1）层次型：该结构强调的是节点间的从属关系，属于递阶控制式，Agent间通过分层的形式进行管理，上层Agent负责对下层Agent进行管理和控制，不同层次的Agent进行通信是必须通过层次间逐层传递，位于同一层次的Agent之间可平等自由地通信，因此系统主要通过上层Agent对下层Agent的控制管理来完成任务。该结构具有很多优点，首先是易于扩展，层次的增删或者在某一层次中Agent的增删对系统其它部分影响很小。其次易于系统功能的划分和管理，层次间的从属关系能够规范下层Agent的行为，相同层次的Agent间易于数据共享。但该结构的缺点同样明显，首先是通信效率较低，当系统规模扩大，层次加深时，最顶层和底层之间的通信需要跨越多个层次，延时明显。而且当同一层次的Agent数量过多时，会加重上层节点控制和协调的负担，最严重的问题是，当上层Agent失效时，层次间的通信将被阻断，因此一般通过增加功能相近的可替换节点来预防该情况的发生，可以一定程度上提高系统的稳定性。

（2）分布型：该结构中各节点间不存在从属关系，不需要经过消息中转，相互间可以平等自由通信，系统通过各Agent之间的协调完成任务。对比层次型结构，分布型结构拥有更高的灵活性和扩展性，由于Agent间自由通信，信息可直达，因此效率更高，并且不存在由于某一Agent失效导致局部或全局通信瘫痪的情况。该结构同样存在缺点，由于不存在Agent间的控制从属关系，因此Agent之间数据共享、状态同步更困难，当任务对数据一致性要求很高时，分布型系统效率反而会下降，因此通常会添加专门用于进行数据存取和共享操作的Agent。再者由于信息在任意的Agent之间可达，提高通信效率的同时也可能导致占用过多的通信带宽，这是由于Agent具有自主性和能动性，可以发起对环境以及Agent的探测，若不以约束，可能导致通信风暴，网络阻塞。

通常情况下，可以综合两种结构的优点，兼顾两者在控制管理和通信效率上的性能，因此就有了混合型的多Agent系统结构，该结构为功能相近以及依赖数据共享的Agent组建Agent组，实施局部的控制管理，并且由组内具有管理职能的Agent发起对环境的探索，能够约束系统整体的通信量，同时功能职责分明的Agent之间不设置从属关系，可自由通信，保证系统的灵活性和高效性。

2.2 动态车间调度系统功能与结构分析

车间调度系统最基本的功能在于满足生产任务在时间和经济效益上的要求，尽可能地给出最优的调度方案，并按照方案把任务安排到对应的加工设备上。目前市场需求动态变化、商品更新迭代的周期不断缩短，制造环境也充满诸多异常因素，因此系统的结构要能够兼顾灵活性和稳定性，当生产任务发生变化时，能够动态调整系统结构；当发生各种异常情况时，系统结构中的各单元能够灵活配合，快速反馈。

完整的柔性作业车间动态调度系统应当具备以下功能：一是能够管理订单任务，包括对任务合法性和优先级的判定；二是对订单任务进行分解分配，把任务分解成工件的工序集合，基于各设备原有的加工任务计算各工序的分配方案；三是能够快速响应异常因素，对原有调度方案进行调整。根据以上功能，本文把柔性车间动态调度系统生产管理涉及的工作流程总结为：订单的输入和处理、产品工艺信息的管理、订单任务的分解分配、计算和筛选调度方案、异常检测和开启异常调度。对应以上流程，本文基于多Agent理论，设计了五类Agent，包括管理Agent、资源Agent、工艺Agent、算法Agent、和监控Agent，其中管理Agent负责订单任务的处理以及对系统内的Agent进行管理；资源Agent分为车间Agent和设备Agent，设备Agent对应系统内的加工设备，负责模拟设备的工作进度和状况，车间Agent负责管理分布其中的设备Agent；工艺Agent负责管理系统内产品的工艺信息，为其他Agent提供数据查询接口；算法Agent是对系统内各种调度算法和重调度算法的封装；监控Agent对应系统的监控设备，通过分析监控数据，反馈系统内设备的工作状况。调度系统把其中的物理实体、数据或算法逻辑封装为以上五类Agent，各Agent各司其职的同时，通过Agent间灵活自由的通信，共同完成生产管理的完整流程。

为了保证系统生产管理中各流程能够平滑衔接，信息高效传递，本文对基于多Agent的柔性作业车间动态调度系统的体系结构设计如下。由于现代制造业的资源具有分布式的特点，在世界各地可能均设有加工工厂，各工厂根据生产技术特点有着各自的车间和设备配置，形成了相对独立的生产系统。为了能够联合各个地理位置的制造资源和生产技术，设置了车间Agent对其中的设备Agent进行管理，同时分配子管理Agent协调工厂内的各个车间Agent，因此子管理Agent、若干车间Agent以及对应的设备Agent形成了一个具有层次型结构的资源Agent组。由图可知，全局管理Agent与各资源Agent组存在从属关系，上层Agent对下层Agent具有控制协调的功能，而全局管理Agent、算法Agent、工艺Agent和监控Agent间是平等的关系，能够自由通信，因此本文调度系统的体系结构属于典型的混合型结构，柔性车间调度系统应用该结构具有如下优点：

（1）易于资源管理和系统扩展。资源Agent组的结构符合现代制造系统的资源分布特点，制造业的资源结构基本是以“工厂-车间-设备”的方式进行呈现，一个加工工厂内设置若干车间，每个车间根据所生产加工的产品、零件来配备对应的生产设备，而企业可能在多个地理位置分布有加工工厂，制造业资源的这种分布特点是与层次型结构一致的。系统是基于各设备Agent的仿真数据对订单任务执行分解分配过程的，该过程涉及诸多数据和状态的共享，通过上层Agent对下层Agent控制管理的能力，能够更有效率协调资源Agent间的工作。

（2）系统的成本降低。车间调度系统把生产管理的各个功能进行划分并分配到对应的Agent，相当于把系统所需的计算量和控制量分散到各个Agent中，降低了系统对大型高性能服务器等硬件的依赖。

（3）系统运行效率高，稳定性强。资源Agent组属于层次型结构，有全局管理Agent进行统一的管理。全局管理Agent、算法Agent、工艺Agent和监控Agent之间是松耦合关系，属于分布式结构，各Agent之间不存在从属关系，消息直接可达，无需经过层次间或中央服务器的消息中转，通信效率高，在协同解决大型复杂的调度任务时更灵活高效。

管理模块：整个系统的管理中枢，为各模块之间的信息传输提供消息中转功能，协调模块间的交互，同时对订单任务进行合法性判断、排序管理，并由此开启调度和重调度流程。

1. 资源模块：表征整个系统中制造资源的分布和结构。该资源专指系统中的制造资源，如机床、机械臂等加工设备，其次资源模块呈现了系统中工厂、车间和设备的分布和组织结构，同时还要对设备的工作状态进行模拟仿真，包括设备的历史工作记录，当前工作进度以及运行状况等。
2. 调度模块：封装了系统所需要的各种算法，用于计算最优的调度方案以及重调度方案。
3. 工艺模块：表征系统所能生产的产品信息，包括产品的基本信息（名称、组成、规格、所需原材料等）、工件的加工工艺。工艺模块在功能上偏向于数据库，管理、更新产品信息的同时为其他模块提供数据查询接口。
4. 监控模块：负责监控设备的工作状态，当存在设备故障时，通知管理模块调整原有调度方案。

从调度流程的角度对调度系统进行模块划分后，需要把各模块相互连接最终形成一个完整的调度系统，这种连接关系呈现的是各模块间的信息流动方向。考虑到柔性作业车间调度系统中工厂的车间结构、车间的设备配置、产品的更新迭代可能频繁发生，因此多Agent系统的结构能够保证系统的扩展性，能够轻易对其中的节点进行增删操作；其次，柔性作业车间调度是典型的NP-hard难题，随着调度规模的增长，需要的计算时间将呈指数增长，在系统易于扩张的同时，还要保证系统原有的灵活性，如模块间以及模块内部个体间的交互仍可维持较高的效率。

2.3 基于多Agent的柔性作业车间动态调度系统的建模

2.3.2 基于多Agent的柔性作业车间动态调度系统的结构设计

本文所研究的调度系统使用的是综合性的混合型系统结构，车间调度系统在功能上可以分为资源、管理、算法、工艺以及监控模块,如图所示，五个模块以分布式的结构构件了完整的调度系统。各模块中的物理实体或算法逻辑均封装为相应的Agent，功能的实现和任务的执行由各个Agent间相互协调完成。与各功能模块相对应，系统中定义了五类Agent：管理Agent，资源Agent、工艺Agent、算法Agent和监控Agent。

资源Agent：对应资源模块，代表系统中的制造资源。资源Agent分为车间Agent和设备Agent。每个设备Agent对应一个加工设备，如机床、机械臂等，用于对设备的加工进度以及状态进行模拟仿真，记录设备的任务队列，即设备当前加工工序的完成进度以及待加工工序安排的开始时间和结束时间。系统的调度过程是基于各设备Agent的仿真数据进行的。每个车间Agent对应工厂内的一个车间，负责对分布于其中的设备Agent进行管理。

管理Agent：对应管理模块，负责

2.4 Agent建模

[009] 所述动态调度系统包括：工艺Agent、管理Agent、监控Agent、资源Agent模块和算法Agent。

[010] 所述工艺Agent对系统的产品工艺信息进行管理，为管理Agent和算法Agent提供数据查询服务。

[011] 所述监控Agent负责采集并分析来自机械状态监测设备、机械故障诊断仪、RFID读取器等设备所采集的工作设备工作进度和状态信息，判断设备的故障类型以及对应的影响，并把相关信息发送至所述管理Agent。

[012] 所述管理Agent负责对全体Agent进行管理监控。管理Agent分为全局管理Agent和子管理Agent。由于企业可能在不同的地理位置设有工厂，为了联合各工厂部门的技术和制造资源，因此系统需为每个工厂分配一个子管理Agent,子管理Agent负责对车间内的所有Agent进行管理。全局管理Agent能够接收客户的生产订单任务，然后对任务的合法性进行判断；子管理Agent能够接收来自监控Agent关于设备工作进度信息，然后发送至对应的设备Agent，与设备Agent的模拟数据进行同步；子管理Agent同时也能接收来自监控Agent的故障信息，把信息发送至

故障设备所属的资源模块后，由资源模块和算法Agent对原有调度方案进行调整，即启动重调度。

[013] 所述资源模块由若干资源Agent组成，资源Agent分为车间Agent和设备Agent。设备Agent负责对其所对应的机床、机械臂等的工序任务队列进行模拟。车间Agent对应工厂内的一个车间，负责对车间内加工设备对应的设备Agent进行管理。当车间Agent接收到来自管理Agent的任务时，根据其下的设备配置与任务所要求的设备进行匹配，若匹配成功，由算法Agent生成调度方案。

[014] 所述算法Agent封装了调度系统内的逻辑方法，可根据系统需求灵活地增加、删除、替换、调整对应算法逻辑。在该系统中，算法Agent至少需要封装两种方法，分别是调度方法和重调度方法，前者计算任务的最小完成时间，生成调度方案，后者当发生订单更改、设备故障等异常因素时，调整原有调度。

[015] 所述工艺Agent管理的工艺信息用于表示系统所能生产加工的产品种类、产品需要的原材料、产品中各零部件的加工工艺流程以及其中各个工序的机器选择等信息。一种产品在结构上由若干零部件装配而成；一个零部件从投入生产到完成需要经过多道工序的加工；每道工序能在若干机器上进行加工，对应不同的完成时间。因此根据产品的结构，工艺Agent对其所存储管理的产品工艺信息分为三个级别：产品工艺信息、零部件工艺信息和工序加工信息。

（1）所述产品由若干零部件组成，对应的产品工艺模型由三元组<I, S,R>来表示，I代表产品信息，S代表产品的零部件组成，R代表约束。产品信息包括产品

名称、产品ID、规格和精度等；零部件组成声明产品的结构；约束表示零部件在生产上的约束关系，若零部件可并行生产，代表该产品的生产具有一定的柔性。

（2）所述零部件需要经历若干道工序的加工，对应的零部件工艺模型由三元组<I,S,R>表示，I代表零部件信息，S代表零部件的工序组成，R代表约束。零部件信息包括零部件名称、零部件ID、需要的原材料、规格和精度等；零部件的工序组成表示该零部件从生产到完成需要经历的工序；约束表示工序的顺序约束。在零部件的若根据该约束可以得到多条可选的工艺路线，表示产品的生产具有一定的柔性。

（3）所述工序能够在若干机器上进行加工对应不同的加工时间，对应的工序加工信息模型由二元组<I,S>组成，I代表工序信息，S代表工序的机器选择。工序信息包括工序名称、ID；机器选择代表能够完成该工序的设备集合以及对应的加工时间。

[016] 本发明所述的工艺Agent通过修改产品工艺模型的约束、零部件工艺模型的约束或工序加工信息模型的约束，能够适用具有柔性或不具有柔性的产品生产，因此具有通用性。

[017] 所述动态调度是通过各Agent间的交互和配合来共同解决任务分解、调度和重调度流程的，各Agent之间的信息交互如下：

[018] 所述工艺Agent能够接收处理如下两类消息请求：

（1）工艺Agent能够响应来自系统管理员对工艺信息的增删查改操作。当系统管理员执行添加新产品操作时，工艺Agent需执行的步骤如下：

S1、若新产品是由系统原有零部件组成的，工艺Agent根据系统管理员对零部件的选择以及指定的约束关系，以所述产品工艺模型三元组<I,S,R>的形式录入

说 明 书

自身数据库中，而后添加完成，结束操作。若新产品在零部件组成上包含工艺Agent所不具有的零部件，执行步骤S2。

S2、若新的零部件是由系统原有工序组成的，工艺Agent根据系统管理员对工序的选择以及指定的工序约束，以所述零部件工艺模型三元组<I,S,R>的形式录入自身数据库中，而后把新零部件与原有零部件组成的新产品录入数据库中，最终新产品添加完成，结束操作。若零部件在工序组成上包含工艺Agent所不具有的工序，执行步骤S3。

S3、工艺Agent根据系统管理员对新工序的信息和机器选择，以所述工序加工信息模型三元组<I,R>的形式录入自身数据库中，而后经过对新零部件、新产品的相继封装得到新产品的信息，至此产品添加完成，结束操作。

[019]（2）工艺Agent能够接收来自管理Agent对产品信息的获取请求，工艺Agent最终返回产品中各零部件所需的设备集。

[020]（3）工艺Agent能够接收来自算法Agent对零部件工艺信息的获取请求，工艺Agent最终返回对应零部件的加工工艺流程，其中主要是工序间的顺序约束以及机器选择。

[021] 所述监控Agent能够发送如下三类消息请求：

（1）监控Agent需定时读取对应设备的工作进度信息，进度信息是指设备正在加工的工序已经完成的百分比，并把进度信息发送至该设备对应的设备Agent，通知设备Agent进行进度同步。

[022]（2）当机械状态监测设备、机械故障诊断仪等监控设备有新数据读入时，监控Agent需要对数据进行分析，得到设备的故障类型以及故障的影响，而后把故障信息发送至全局管理Agent通知其启动重调度。

[023]（3）当故障设备维修完毕可重新投入使用后，返回该信息通知全局管理Agent再次启动重调度。

[024] 所述管理Agent能够接收处理如下三类信息请求：

（1）全局管理Agent能够接收来自客户的产品订单任务，订单中标注了所需生产的产品、产品数量、产品的规格和精度等等。管理Agent需对订单合法性进行判断，先向所述工艺Agent获取任务所要求的产品工艺信息，根据系统的设备配置和库存原材料判断是否匹配，若匹配成功，表示系统有能力生产该产品，然后把任务发送至其下各个子管理Agent;若匹配失败，向客户返回无法完成订单的结果。

[025]（2）子管理Agent能够接收来自全局管理Agent的任务请求，而后把任务分别发送至其下的各个车间Agent，获取到各车间Agent的调度方案后，从中选择用时最短的方案；若不存在车间Agent能够完成任务，子Agent执行基于订单任务的分解与分配方法。

[026]（3）全局管理Agent能够接收来自所述监控Agent的设备故障信息，而后清空故障设备上原有的工序任务队列，向算法Agent发送重调度请求。

[027]（4）全局管理Agent能够接收来自所述监控Agent的故障修复信息，而后把原故障设备重新加入调度集合，再次启动重调度流程。

[028] 所述资源模块能够接收如下两类消息请求：

（1）车间Agent能够接收来自其所属的子管理Agent的任务请求，该任务可能是产品级、零部件级或工序级别的加工任务，车间Agent把任务所要求的加工设备

说 明 书

与自身拥有的设备配置进行匹配，表示该车间有能力完成该任务，然后向算法Agent发起调度请求，计算调度方案；否则，向所述子管理Agent返回任务无法完成的判断结果。

[029]（2）设备Agent能够接收来自监控Agent的的设备工作进度信息请求，而后设备Agent需进行进度同步，即根据该进度信息调整自身所模拟的设备工作进度，以保证模拟数据与设备实际工作进度的误差控制在可接受范围内。

[030] 所述算法Agent能够接收处理如下两类消息请求：

（1）算法Agent能够接收来自所述车间Agent的调度请求，该请求附有参与调度的任务以及设备Agent所模拟的设备工序任务队列，算法Agent先向所述工艺Agent获取任务所要求的零部件加工工艺流程，流程中标注了各工序的顺序约束关系以及对应的加工设备，而后根据设备Agent所模拟的工序任务队列，执行调度方法对零部件任务的工序进行分配，最终得到用时尽可能少的调度方案，该方案最终返回到所述子管理Agent。

[031]（2）算法Agent能够接收到来自所述子管理Agent的重调度请求，该请求附有故障设备集合和故障信息，算法Agent需根据设备故障的类型以及影响程度采用对应的重调度方法，并把重调度方案返回到全局管理Agent。

[032] 同时，本发明还提供了一种基于订单任务的分解分配方法以及面向设备故障的重调度方法。

[033] 基于订单任务的分解分配方法根据企业分布在各地的工厂的生产能力把

订单任务进行分解，以达到联合各工厂的技术和制造资源，提高生产效率的目的，该方法的执行步骤如下：

[1] 基于多Agent的可中断作业车间调度问题研究\_万明谊