文章编号: 1001-0920(2007)12-1335-06

知识化制造系统中动态调度的自适应策略研究

杨宏兵, 严洪森
(东南大学 自动化研究所, 南京 210096)

摘 要: 针对知识化制造系统中的动态调度问题,结合知识化制造单元的高智能特征,提出了 B-Q 学习算法,并基于该算法构建了一种自适应调度控制策略. 针对知识化制造系统运行过程中系统状态空间较大的特点,通过提取系统状态特征,对系统状态进行合理聚类,有效地降低了系统状态空间的复杂性. 根据系统当前所处的瞬时状态,选取不同的调度规则对缓冲区中工件进行有效调度. 仿真结果验证了所提出调度控制策略的有效性.

关键词: 知识化制造单元; 动态调度; B-Q 学习; 控制策略
中图分类号: TH165 **文献标识码**: A

Adaptive strategy of dynamic scheduling in knowledgeable manufacturing system

YANG Hong-bing, YAN Hong-sen
(Research Institute of Automation, Southeast University, Nanjing 210096, China. Correspondent: YANG Hong-bing, E-mail: tonyyhb@gmail.com)

Abstract: Aiming to the problem of dynamic scheduling in knowledgeable manufacturing system, the B-Q learning algorithm is proposed by combining the high intelligent characteristic of knowledgeable manufacturing cell, and a kind of adaptive scheduling control strategy is presented based on this algorithm. In view of the characteristic of a large scale state space during KMS running, through the extraction of state feature and reasonable state clustering, the complexity of system state space is reduced effectively. According to current system transient state, different dispatch rules of scheduling the jobs are selected, and thus the effective scheduling can be obtained for the orders in buffer. Simulation results show the effectiveness of the scheduling control strategy.

Key words: Knowledgeable manufacturing cell; Dynamic scheduling; B-Q learning; Control strategy

1 引 言

知识化制造是人们于 2001 年提出的新的制造理念[1]. 该理念将一种先进制造模式看作一种先进制造知识,通过 Agent 网与知识网(KM)之间一一对应的同构映射关系,将已有的先进制造模式纳入知识化制造系统(KMS)[2],以满足各类制造企业的不同需求. 知识化制造系统是一个高智能制造系统,具有自适应、自学习、自进化、自重构、自培训和自维护等特征,能够通过自身不断的学习和进化,适应外界环境的变化[1]. 自适应作为知识化制造实现中的关键技术之一,涉及制造系统所涵盖的多个领域. 在实际生产过程中,动态调度问题通常面临带有不可预测扰动的调度环境,因此研究一种动态调度的自适应策略具有重要的意义.

目前,研究动态调度问题的主要方法[3,4]有:最优化方法、系统仿真方法、启发式方法、人工智能方法及计算智能方法等. 已有的研究表明,机器数 $m \geq 3$ 的 n 个工件的调度问题就是 NP 困难(NP-hard)的,至今尚未找到多项式复杂程度的算法解决此问题. 调度规则方法具有对 NP 特性不敏感且实时性好等优点,是实际生产中应用较为广泛的启发式方法. 文献[5]在不同的系统参数条件下,对一些常用的调度规则进行研究后发现:制造系统状态发生变化时,原来效果较好的调度规则可能变得平庸. 可见单个调度规则缺乏全局性. 为了提高调度规则的性能,文献[6]利用神经网络构建生产控制系统,选取合适的调度规则. 该方法训练时间长,对结果的解释能力较差,且问题规模增大会使网络结构变得更加

收稿日期: 2006-08-19; **修回日期**: 2007-01-08.
基金项目: 国家自然科学基金项目(60574062,50475075).
作者简介: 杨宏兵(1977—),男,安徽无为人,博士生,从事知识化制造、机器学习的研究; 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、知识化制造等研究.

复杂. 文献[7]在单件车间调度中,通过训练好的模糊系统选取调度规则. 该方法所选取的模糊规则较少,且专家知识不易获得. Piramuthu 等人在假定系统参数不变的条件下得到归纳学习的训练样本,利用归纳学习的结果动态地选取相关的调度规则[8]. 但对于系统参数频繁变化的动态调度而言,假定在稳态条件下得到训练样本显然是不准确的. 这种训练样本不易获取的问题在文献[6]中也同样存在.

为了避免上述训练样本获取困难的问题,本文结合知识化制造单元(KMC)高智能的特征,在强化学习的基础上,给出B-Q学习算法,并对其收敛性进行证明. 在研究B-Q学习算法的基础上,构建了一种能够适应环境变化的调度控制策略(BQ-ASCS),最后通过仿真实验说明了该控制策略的有效性.

2 问题描述

知识化制造系统是由多个知识化制造单元(KMC)组成的动态系统. KMC 由加工 Agent,搬运 Agent,调度决策模块,光栅以及传感器等一些检测设备组成,具有对系统进行实时监控、数据采集、信息处理及决策的能力. 考虑包含 M 个加工 Agent 的知识化制造单元,对单元中 N 个工件进行调度,使得工件平均拖期最小. 工件集合表示为 $J=\{J_1,J_2,\cdots,J_N\}$,每个工件都由多工序组成. 工件随机到达单元等待加工,工件相互独立且没有优先级,一台机器在同一时间内只能加工一个工件,工件在加工过程中不允许中断或被重新安排. 第 j 个工件的理想交货期为 d_j,实际完工时间为 C_j,则最小化平均拖期的目标函数为 $\min\left[\sum_j^N \phi\max(C_j-d_j,0)/N\right]$,其中 ϕ 是工件拖期完工惩罚因子.

3 B-Q学习算法分析

强化学习技术已在库存、机器重组及预防性维护等制造领域有一定的应用[9,10],并取得了较好效果. 但其在动态调度生产中的应用尚显不足,这是由于在系统状态不断变化的动态调度中,系统状态空间(即离散状态数目)非常庞大,传统的强化学习因为结构信用分配问题,导致其泛化能力不足,当面对大规模状态空间问题时,学习效果会变得不理想,难以满足实际生产需要. 文献[11]提出了 Q-III 学习算法,并对单件车间进行调度,在学习过程中利用了领域知识和经验,但这些知识和经验一般很难获得而且不准确,对提出的算法收敛性也缺少分析. 为了消除这种大状态空间的不良影响,笔者结合基本顺序算法方案(BSAS)[12]和Q学习算法给出了B-Q学习算法. 该算法通过BSAS对 KMC 仿真器产生的状态进行聚类,在得到聚类状态的基础上进行学习,故只需搜索较小的状态空间,有效地提高了学习效率,增强了泛化能力.

3.1 状态特征的选取

定义 1 在调度决策时刻点,计算知识化制造单元中各缓冲区中的工件剩余加工时间,将最大剩余加工时间称作最大机器负载,用 $\omega_{\max}$ 表示.

定义 2 在调度决策时刻点,计算知识化制造单元中各缓冲区中的工件剩余加工时间,将其平均值定义为平均机器负载,记为 $\bar{\omega}$.

定义 3 知识化制造单元的最大机器负载与平均机器负载的比值称为相对机器负载,用 ω 表示,即 $\omega=\omega_{\max}/\bar{\omega}$.

知识化制造单元的状态空间用集合 S 表示,有 $S=\cup s_i$,其中 s_i 表示知识化制造单元运行的实际状态. 一个完整的系统状态往往由十几个甚至几十个状态特征组成,在这种高维情况下,特征变化通过组合效应,常常导致系统状态个数呈现指数性增加,使得状态数目非常巨大. 为减少这种影响,本文选取对调度规则性能影响较大的 4 个状态特征:平均交货因子,系统利用率,相对机器负载和平均松弛时间. 因此,状态 s_i 由4元组组成,即 $s_i=(\omega,f,\mu,\bar{\zeta})$. 特征变量 ω 为相对机器负载,f 表示系统的平均交货因子,即 $f=\sum f_j/N_d$. 其中:f_j 是知识化制造单元中第 j 个工件的交货因子,反映第 j 个工件交工期的松紧程度;N_d 是系统中的工件总数;μ 是特征变量,表示系统利用率,其含义是知识化制造单元中当前非空闲加工 Agent 数与总的加工 Agent 数之比. ζ 是状态特征变量,表示平均松弛时间,若 ζ_j 表示第 j 个工件的松弛时间,则有 $\zeta_j=d_j-t-\sum_{q=k_d}^{k_j}p_{jq}$. 其中:$t$ 是当前时刻;p_{jq} 表示第 j 个工件的工序 q 所需加工时间(若工序 q 正在被加工,则 p_{jq} 为该工序的剩余加工时间);k_d 是工件正在被加工或等待加工的工序数;k_j 表示工件 j 的工序总数,则有 $\bar{\zeta}=\left(\sum\zeta_j\right)/N_d$.

定义 4 通过BSAS对系统状态进行聚类,并得到 x 个聚类,则将第 u 个聚类中所有系统状态的中心称为聚类状态 s_u^c,故共有 x 个聚类状态,记为 $S^c=\cup s_u^c(u=1,2,\cdots,x)$. 如果将第 u 个聚类中的每个状态视为单位质点,则 s_u^c 就是第 u 个聚类中所有状态的重心.

对系统状态进行聚类时,因为BSAS中的不相似测度采用欧氏距离法,而欧氏距离法与特征值的数量级有关,因此必须对状态特征值进行标准化预处理. 特征标准化方法有多种,其中比例因子法具有简单易用且能保持特征原有语义等特点,故本文采

用比例因子法对上述 4 个状态特征进行标准化处理,以平衡各特征在聚类中的作用.

3.2　B-Q学习算法的提出

任意调度决策时刻点,知识化制造单元将根据当前系统聚类状态 s_t^c 来选择正确的动作 a_t,即选择何种调度规则对缓冲区中待加工工件进行调度. 动作 a_t 是动作集 A 中的一个动作,$a_t \in A$,$A = (a_1, a_2, \cdots, a_\beta)$,动作集中每个动作对应一个调度规则. 动作 a_t 作用于系统后产生一个后继的聚类状态 s_{t+1}^c,学习器由此得到一个立即回报值 r_{t+1}. 知识化制造单元整个运行过程中,将会产生一系列的立即回报值. 为使学习器能够根据在时间轴上展开的立即回报序列学习到最优控制策略,即由检测到的聚类状态来选择最优的动作,定义了评估函数.

定义 5　知识化制造单元从 t 时刻的聚类状态 s_t^c 开始,根据某个控制策略执行动作 a_t,此后也遵循该策略执行所得的折算累积回报期望值,称之为状态 - 动作对 (s_t^c, a_t) 的评估函数,记为 $Q(s_t^c, a_t)$.

对于任意状态 - 动作对 (s_u^c, a_v),由评估函数定义可得 $Q(s_u^c, a_v)$ $(1 \le u \le x, 1 \le v \le \beta)$ 的表达式

$$Q(s_u^c, a_v) = \mathrm{E}\left\{\sum_k^{\infty} \gamma^k r_{t+k+1} \mid s_t^c = s_u^c, a_t = a_v\right\}, \quad (1)$$

式中 $\gamma(0 \le \gamma < 1)$ 是对延迟回报的折扣因子,确定延迟回报和立即回报的相对比例. 当从知识化制造单元的状态 - 动作对 (s_u^c, a_v) 开始,每步都遵循最优控制策略执行动作 a,则可得最大的折算累积回报期望值,即最优评估函数 $Q^*(s_u^c, a_v)$. 学习最优控制策略,本质上也就是学习如何得到最优评估函数. 由于评估函数 $Q(s_t^c, a_t)$ 中的状态是聚类状态,评估函数值在学习过程中变动较为剧烈,使得学习过程平稳性较差,从而影响学习效果. 为了减小这种波动,引入了评估函数阈值,用 Θ 表示. 为此,给出如下 B-Q 学习算法的迭代学习模型:

$$Q_n(s_t^c, a_t) = \begin{cases} Q_{n-1}(s_t^c, a_t) + \alpha_n(s_t^c, a_t)[r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t)], \\ \quad \mid r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t) \mid \le \Theta; \\ Q_{n-1}(s_t^c, a_t) + \alpha_n(s_t^c, a_t)[r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t) - \phi(n)\Delta_{\Theta 1}], \\ \quad r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t) > \Theta; \\ Q_{n-1}(s_t^c, a_t) + \alpha_n(s_t^c, a_t)[r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t) - \phi(n)\Delta_{\Theta 2}], \\ \quad r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t) < -\Theta. \end{cases} \quad (2)$$

其中

$$\Delta_{\Theta 1} = r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t) - \Theta, \quad (3)$$

$$\Delta_{\Theta 2} = r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t) + \Theta; \quad (4)$$

$Q_n(s_t^c, a_t)$ 表示第 n 次循环时的评估函数;$\phi(n)$ 是循环次数 n 的函数;α_n 为步长参数,在学习过程中以一定的速率减小 α_n 可以达到收敛到最优评估函数的目的. 步长参数 α_n 可用下式得到:

$$\alpha_n(s_t^c, a_t) = C_\alpha / (1 + \mathrm{visits}_n(s_t^c, a_t)). \quad (5)$$

式中:C_α 是步长参数的权系数变量;$\mathrm{visits}_n(s_t^c, a_t)$ 表示在 n 次循环中,状态 - 动作对 (s_t^c, a_t) 被访问的总次数,随着状态 - 动作对被访问次数的增加,步长参数 α_n 值将会随之减小. 在学习过程中,当知识化制造单元处于状态 s_t^c 时,学习器将采用 εgreedy 法选取动作 a,即以概率 $(1 - \varepsilon)$ 选择具有最大评估函数值 $(\max_{a_t} Q(s_t^c, a_t))$ 的动作 a_t,以概率 ε 随机选取动作集 A 中其他动作.

根据上述分析,B-Q 学习算法的具体实现步骤如下:

Step1: 初始化聚类数 $x = 1$,$i = 1$,置最大聚类数为 K,KMC 仿真器产生的状态数为 κ. 运行 KMC 仿真器,学习器得到仿真器产生的初始状态 s_1. 对 s_1 进行特征标准化处理,得到第 x 个聚类

$$C_x = \{s_1\} \cdot d(s_t, C_u) = \min_{1 \le l \le x} d(s_t, C_l).$$

Step2: $i = i + 1$,对状态 s_i $(2 \le i \le \kappa)$ 进行特征标准化处理,采用欧几里德(Euclidean)距离法计算状态 s_i 到聚类 C_l $(1 \le l \le x)$ 的不相似性测度 $d(s_i, C_l)$,得到与 s_i 不相似性测度最小的聚类 C_h,即 $d(s_i, C_h) = \min_{1 \le l \le x} d(s_i, C_l)$.

Step3: 如果 $x < K$,且 $d(s_i, C_h) > \Omega$,Ω 为基本顺序算法方案(BSAS)的不相似性阈值,则有 $x = x + 1$,聚类 $C_x = \{s_i\}$,否则将状态 s_i 聚类到 C_h 中,即有 $C_h = C_h \cup s_i$,并重新计算聚类状态 s_h^c. 返回 Step2,直至将所有 κ 个状态聚类完毕,可得到 x 个聚类 C_l 和聚类状态 s_u^c,$l = 1, 2, \cdots, x$,$u = 1, 2, \cdots, x$.

Step4: 初始化所有动作 - 状态对 (s_u^c, a_v) 的评估函数,记为 $Q_0(s_u^c, a_v)$,$u = 1, 2, \cdots, x$,$v = 1, 2, \cdots, \beta$. 置循环次数 $n = 1$. 在知识化制造单元运行的初始时刻 t_0,从动作集中任选动作 a_{t_0} 进行加工.

Step5: $n = n + 1$,学习器检测知识化制造单元当前时刻 t 的状态 s_t,计算不相似性测度 $d(s_t, C_l)$,$l = 1, 2, \cdots, x$,得到 $d(s_t, C_u) = \min_{1 \le l \le x} d(s_t, C_l)$,则 t 时刻的聚类状态 $s_t^c = s_u^c$. 学习器根据 εgreedy 法选择

动作 a_v,作用于加工 Agent,即 $a_t = a_v, a_v \in A$.

Step6: 观察 $t+1$ 时刻知识化制造单元状态 s_{t+1},计算不相似性测度,得到当前系统聚类状态 s^c_{t+1}. 此时学习器会收到一个立即回报值 r_{t+1},利用式(2)对评估函数 $Q_n(s^c_u, a_v)$ 进行迭代调整.

Step7: 用聚类状态 s^c_{t+1} 替换 s^c_t,循环 Step5 ~ Step7,直到学习到所有动作 - 状态对的最优评估函数 $Q^*(s^c_u, a_v)$.

需要说明的是,Step5 和 Step6 仅调整评估函数 $Q_n(s^c_u, a_v)$,调整一次就是一个循环,即调用式(2),才有循环次数 $n = n+1$. Step7 用聚类状态 s^c_{t+1} 替换 s^c_t 后,才开始对 $Q_n(s^c_{t+1}, a_v)$ 进行调整.

3.3 自适应调度控制策略(BQ ASCS)

基于 B-Q 学习算法的知识化制造单元,其自适应调度控制策略物理结构如图 1 所示.

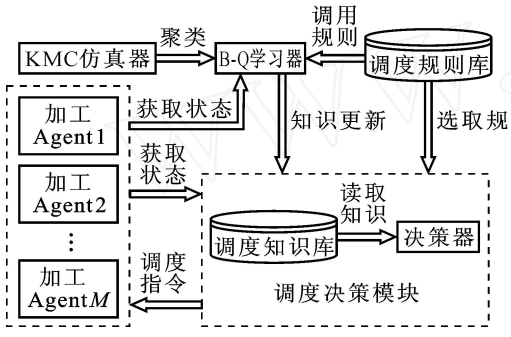

图 1 自适应调度控制策略物理结构

由图 1 可以看出 BQ-ASCS 策略工作原理:工件到达知识化制造单元并被加工时,学习器检测当前系统状态并运用 B-Q 学习算法进行学习,获取系统的动态调度知识,进而对调度知识库中的知识进行更新. 当某加工 Agent 需要调度时,决策器将根据检测到的系统状态,读取调度知识库中对应的调度知识,选择合适的调度规则对该加工 Agent 进行调度,从而保证知识化制造单元的顺利运行.

由此可知,BQ-ASCS 策略主要通过不断地学习来获取新的调度知识,面向知识化制造单元的状态来动态选取调度规则. 该控制策略对动态调度中的频繁扰动具有很强的适应性. 需要指出的是,B-Q 学习器的学习以及对调度知识库中的知识更新可通过离线学习完成.

4 B-Q学习算法收敛性分析

为了得到 B-Q 学习算法的收敛性性质,先给出如下引理:

引理 1[13] 对于随机过程 $(\alpha_t, \Delta_t, F_t)$, $t \geq 0$,有 $\alpha_t, \Delta_t, F_t: Y \to \mathbf{R}$ 满足以下方程:

$$\Delta_{t+1}(y) = (1 - \alpha_t(y))\Delta_t(y) + \alpha_t(y) F_t(y), \quad y \in Y, t = 0, 1, \cdots \tag{6}$$

设 P_t 为递增的 σ 域序列,则随机变量 α_0 和 Δ_0 为 P_0 可测的,α_t, Δ_t 和 F_{t-1} 为 P_t 可测的, $t = 1, 2, \cdots$: 如果能满足下列条件:1) Y 为有限集;2) $0 \leq \alpha_t(y) \leq 1$, $\sum_t^{\infty} \alpha_t(y) = \infty$, $\sum_t^{\infty} \alpha_t^2(y) < \infty$ 以概率 1 成立;3) $\| E\{F_t(\cdot) \mid P_t\} \|_W \leq \theta \| \Delta_t \|_W + \delta_t, \theta \in [0, 1)$, 且 δ_t 以概率 1 收敛至零;4) 方差 $\mathrm{Var}\{F_t(y) \mid P_t\} \leq C(1 + \| \Delta_t \|_W)^2$, C 是一个常量. 则 Δ_t 以概率 1 收敛至零. 式中符号 $\| \cdot \|_W$ 表示加权无穷范数,即若 $W = (w_1, w_2, \cdots, w_\pi)$, $\Delta_t = (\Delta_t(y_1), \Delta_t(y_2), \cdots, \Delta_t(y_\pi))$, $y_i \in Y$,则有 $\| \Delta_t \|_W = \max_i (\Delta_t(y_i) \mid w_i)$, $i = 1, 2, \cdots, \pi$.

为便于后面定理的证明,引理 1 中用向量范数 $\| \Delta_t \|_W$ 替代了文献[13]定理中泛函 Δ_t 的范数 $\| \Delta_t \|_W$,显然有 $\| \Delta_t \|_W = \max_i (| \Delta_t(y_i) | / w_i)$, $i = 1, 2, \cdots, \pi$ 证明略.

定理 1 令 Y 为状态 - 动作对 (s^c_t, a_t) 的有限集,对于任意动作 - 状态对 (s^c_u, a_v),有 $\Delta_n(s^c_u, a_v) = Q_n(s^c_u, a_v) - Q^*(s^c_u, a_v)$. 令向量 $\Delta_n = (\Delta_n(s^c_1, a_1), \Delta_n(s^c_1, a_2), \cdots, \Delta_n(s^c_x, a_\beta))$, $F_n(s^c_t, a_t) = r_{t+1} + \gamma \max_{a \in A} Q_n(s^c_{t+1}, a) - Q^*(s^c_t, a_t)$, P_t 是由集合 $\{s^c_t, \alpha_n, a_t, r_{t-1}, \cdots, s^c_1, \alpha_1, a_1, Q_0\}$ 中一切子集组成的 σ 域. 则有

$$\| \mathrm{E}\{F_n(s^c_t, a_t) \mid P_t\} \| \leq \gamma \| \Delta_n \|, \tag{7}$$

式中 $\| \cdot \|$ 是无穷范数.

证明

$$\| \mathrm{E}\{F_n(s^c_t, a_t) \mid P_t\} \| = \| \mathrm{E}\{r_{t+1} + \gamma \max_{a \in A} Q_n(s^c_{t+1}, a) - Q^*(s^c_t, a_t)\} \| =$$

$$\gamma \| \sum \mathrm{pr}^{a_t}_{s^c_t s^c_{t+1}} \max_{a \in A} \| Q_n(s^c_{t+1}, a) - \max_{a \in A} Q^*(s^c_{t+1}, a) \} \| \leq$$

$$\gamma \sum \mathrm{pr}^{a_t}_{s^c_t s^c_{t+1}} \max_{a \in A} \| Q_n(s^c_{t+1}, a) - Q^*(s^c_{t+1}, a) \|. \tag{8}$$

其中 $\mathrm{pr}^{a_t}_{s^c_t s^c_{t+1}}$ 是在状态 s^c_t 时动作 a_t 作用于加工 Agent 后得到状态 s^c_{t+1} 的概率,故有 $\sum \mathrm{pr}^{a_t}_{s^c_t s^c_{t+1}} = 1$. 因此

$$\gamma \sum \mathrm{pr}^{a_t}_{s^c_t s^c_{t+1}} \max_{a \in A} \| Q_n(s^c_{t+1}, a) - Q^*(s^c_{t+1}, a) \| \leq$$

$$\gamma \max_{a \in A} \| Q_n(s^c_{t+1}, a) - Q^*(s^c_{t+1}, a) \| \leq$$

$$\gamma \max_{s^c_u \in S^c, a_v \in A} \| Q_n(s^c_u, a_v) - Q^*(s^c_u, a_v) \| =$$

$$\gamma \| \Delta_n \|. \tag{9}$$

由式(8)和(9)可得 $\| \mathrm{E}\{F_n(s^c_t, a_t) \mid P_t\} \| \leq \gamma \| \Delta_n \|$. □

根据引理 1 和定理 1,可得到 B-Q 学习算法的如下性质:

定理 2 令 Y 为状态 - 动作对 (s^c_t, a_t) 的有限

集,$\Delta_n(s_u^c, a_v)$ 和 Δ_n 定义与定理 1 相同,假定系统动作 - 状态对 (s_u^c, a_v) 是有限的,且系统每个动作 - 状态对 (s_u^c, a_v) 都能被无限频繁访问,P_t 是由集合 $\{s_t^c, \alpha_n, a_t, r_{t-1}, \cdots, s_1^c, \alpha_1, a_1, Q_0\}$ 中一切子集组成的 σ 域,如果能够满足下列条件:1) 对于步长参数 α_n,有 $0 \leq \alpha_n(s_u^c, a_v) \leq 1$, $\sum_n^{\infty}\alpha_n(s_u^c, a_v) = \infty$, $\sum_n^{\infty}\alpha_n^2(s_u^c, a_v) < \infty$;2) 对于任意的立即回报 r_t,有 $|r_t| \leq C_r$,C_r 是一常量;3) 当循环次数 n 趋向于无穷大时,函数 $\phi(n)$ 以概率 1 收敛至零. 则当 B-Q 算法学习器的学习循环次数 n 趋向于无穷大时,式(2) 中的 $Q_n(s_u^c, a_v)$ 会以概率 1 收敛至最优评估函数 $Q^*(s_u^c, a_v)$.

证明　由步长参数 α_n 的定义,显然有 $0 \leq \alpha_n(s_u^c, a_v) \leq 1$. 根据式(5) 及调和级数的性质,可得 $\sum_n^{\infty}\alpha_n(s_u^c, a_v) = \infty$. 由式(5) 以及柯西收敛原理可知级数 $\sum_n^{\infty}\alpha_n^2(s_u^c, a_v)$ 收敛,即 $\sum_n^{\infty}\alpha_n^2(s_u^c, a_v) < \infty$. 将 Δ_n,α_n 和 F_n^1 代入式(6),可得

$$\Delta_{n+1}(s_u^c, a_v) = (1 - \alpha_n(s_u^c, a_v))\Delta_n(s_u^c, a_v) + \alpha_n(s_u^c, a_v) F_n^1(s_t^c, a_t). \quad (10)$$

结合式(2) 和(10),$F_n^1(s_t^c, a_t)$ 可分为 3 种情况:

1) 当 $|r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t)| \leq \Theta$ 时,有

$$F_n^1(s_t^c, a_t) = r_{t+1} + \gamma \max_{a \in A} Q_n(s_{t+1}^c, a) - Q^*(s_t^c, a_t); \quad (11)$$

2) 当 $r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t) > \Theta$ 时,有

$$F_n^1(s_t^c, a_t) = r_{t+1} + \gamma \max_{a \in A} Q_n(s_{t+1}^c, a) - \phi(n)\Delta_{\Theta_1} - Q^*(s_t^c, a_t); \quad (12)$$

3) 当 $r_{t+1} + \gamma \max_a Q_{n-1}(s_{t+1}^c, a) - Q_{n-1}(s_t^c, a_t) < -\Theta$ 时,有

$$F_n^1(s_t^c, a_t) = r_{t+1} + \gamma \max_{a \in A} Q_n(s_{t+1}^c, a) - \phi(n)\Delta_{\Theta_2} - Q^*(s_t^c, a_t). \quad (13)$$

由式(11) ～ (13) 可得

$$\begin{aligned}\| \mathrm{E}\{F_n^1(s_t^c, a_t) \mid P_t\} \| = & \Big\| \sum \mathrm{pr}_{s_t^c s_{t+1}^{c1}}^{a_t} + [r_{t+1} + \gamma \max_{a \in A} Q_n(s_{t+1}^c, a) - Q^*(s_t^c, a_t)] + \sum \mathrm{pr}_{s_t^c s_{t+1}^{c2}}^{a_t} [r_{t+1} + \gamma \max_{a \in A} Q_n(s_{t+1}^c, a) - \phi(n)\Delta_{\Theta_1} - Q^*(s_t^c, a_t)] + \\ & \sum \mathrm{pr}_{s_t^c s_{t+1}^{c3}}^{a_t} [r_{t+1} + \gamma \max_{a \in A} Q_n(s_{t+1}^c, a) - \phi(n)\Delta_{\Theta_2} - Q^*(s_t^c, a_t)] \Big\|. \end{aligned} \quad (14)$$

式中 $\mathrm{pr}_{s_t^c s_{t+1}^{c1}}^{a_t}$,$\mathrm{pr}_{s_t^c s_{t+1}^{c2}}^{a_t}$ 和 $\mathrm{pr}_{s_t^c s_{t+1}^{c3}}^{a_t}$ 分别对应式(11) ～ (13) 中状态转换的发生概率,故有

$$\sum \mathrm{pr}_{s_t^c s_{t+1}^{c1}}^{a_t} + \sum \mathrm{pr}_{s_t^c s_{t+1}^{c2}}^{a_t} + \sum \mathrm{pr}_{s_t^c s_{t+1}^{c3}}^{a_t} = 1. \quad (15)$$

由式(14) 和(15) 可得

$$\begin{aligned}\| \mathrm{E}\{F_n^1(s_t^c, a_t) \mid P_t\} \| = & \Big\| \mathrm{E}\Big[r_{t+1} + \gamma \max_{a \in A} Q_n(s_{t+1}^c, a) - Q^*(s_t^c, a_t)\Big] - \sum \mathrm{pr}_{s_t^c s_{t+1}^{c2}}^{a_t} \phi(n)\Delta_{\Theta_1} - \sum \mathrm{pr}_{s_t^c s_{t+1}^{c3}}^{a_t} \phi(n)\Delta_{\Theta_2} \Big\| \leq \\ & \Big\| \mathrm{E}\Big[r_{t+1} + \gamma \max_{a \in A} Q_n(s_{t+1}^c, a) - Q^*(s_t^c, a_t)\Big] \Big\| + \Big\| \sum \mathrm{pr}_{s_t^c s_{t+1}^{c2}}^{a_t} \phi(n)\Delta_{\Theta_1} + \sum \mathrm{pr}_{s_t^c s_{t+1}^{c3}}^{a_t} \phi(n)\Delta_{\Theta_2} \Big\|. \end{aligned} \quad (16)$$

根据定理 1 和不等式(16),有

$$\| \mathrm{E}\{F_n^1(s_t^c, a_t) \mid P_t\} \| \leq \gamma \| \Delta_n \| + \Big\| \sum \mathrm{pr}_{s_t^c s_{t+1}^{c2}}^{a_t} \phi(n)\Delta_{\Theta_1} + \sum \mathrm{pr}_{s_t^c s_{t+1}^{c3}}^{a_t} \phi(n)\Delta_{\Theta_2} \Big\|. \quad (17)$$

由于立即回报 r_t 是有界的,由式(3) 和(4) 可知 Δ_{Θ_1} 和 Δ_{Θ_2} 也是有界的. 而当循环次数 n 趋向于无穷大时,函数 $\phi(n)$ 以概率 1 收敛至零,故可知不等式(17) 最后一项也以概率 1 收敛于零.

由于立即回报 r_t 和评估函数 $Q(s_t^c, a_t)$ 均有界,根据方差的性质可知,总能找到一个常量 C 使不等式 $\mathrm{Var}\{F_n^1(s_t^c, a_t) \mid P_t\} \leq C(1 + \| \Delta_n \|_W)^2$ 成立.

根据引理 1 可知,Δ_n 是以概率 1 收敛至零,即当学习循环次数 n 趋向于无穷大时,评估函数 $Q(s_u^c, a_v)$ 会以概率 1 收敛至其最优评估函数 $Q^*(s_u^c, a_v)$. □

5　仿真实验

工件到达知识化制造单元的时间间隔服从负指数分布,平均到达率为 λ. 工件 j 的工序总数 k_j 为集合 $\{1, 2, \cdots, 6\}$ 中随机选取的整数,每道工序加工时间服从均匀分布 $U(u_{p1}, u_{p2})$. 工件被随机分配到任意机器缓冲区中等待加工,且同一工件的相邻两道工序不能由同一个加工 Agent 处理.

调度规则库中调度规则选用最早交货期优先 EDD,最短加工时间优先 SPT 和最小松弛时间优先 MST 共 3 个常用规则. 在实验中,第 j 个工件的交货期 d_j 设定为

$$d_j = rt_j + f_j \sum_q^{k_j} p_{jq}.$$

其中:p_{jq} 表示第 j 工件的工序 q 所需加工时间;k_j 表示工件 j 的工序总数;rt_j 是工件到达 KMC 时刻,交货因子 f_j 服从均匀分布,即 $f_j \sim U(u_{f1}, u_{f2})$. 由于本文的目标函数是最小化平均拖期,而 B-Q 学习算法收敛于最大值,故将目标函数乘以负数转换成最大值问题. 于是对 B-Q 学习算法中的立即回报值 r 设定如下:

$$r=\begin{cases}-(C_j-d_j), & \text{工件 } j \text{ 发生拖期;}\\ 1, & \text{工件 } j \text{ 无拖期.}\end{cases}$$

实验主要参数设定如表 1 所示.

表 1 实验主要参数

M	N	λ	u_{p1}	u_{p2}	u_{f1}	u_{f2}	ε	γ
6	2 400	1/5.5	2	13	1	6	0.15	0.7

知识化制造单元每处理完 2 400 个工件称为一个 episode,共对 500 个 episode 进行仿真. 考虑到各种随机因素的影响,依次对 50 个 episode 平均拖期的均值进行比较. 在 Pentium-2.8G 个人计算机上,采用 Matlab 6.5 作为编程语言进行实验,结果如表 2 所示,其变化趋势如图 2 所示. 选取不同松紧程度的交货因子,令 $u_{f1}=1$, $u_{f2}=5.5,6,\cdots,7.5$,对 200 个 episode 进行仿真. 同样考虑随机因素的影响,对其平均拖期的均值进行比较. 由于 BQ-ASCS 策略需要一个学习的过程,故从第 100 个 episode 开始取值,结果如图 3 所示. 同理,表 2 中总均值没有考虑其前两个平均拖期的均值.

表 2 不同调度规则工件平均拖期比较

调度规则	每 50 个 episode 平均拖期的均值										总均值 (500episode)
	1	2	3	4	5	6	7	8	9	10	
EDD	10.271	9.863	10.458	10.447	10.576	10.335	10.299	11.459	9.673	9.618	10.300
MST	10.047	10.642	11.171	9.687	9.658	9.989	10.813	10.910	11.308	10.550	10.478
SPT	16.096	15.593	17.148	15.592	14.872	16.595	14.591	16.408	16.480	16.268	15.964
BQ-ASCS	9.628	8.785	7.925	9.527	9.405	9.429	9.134	8.858	9.018	9.060	9.045

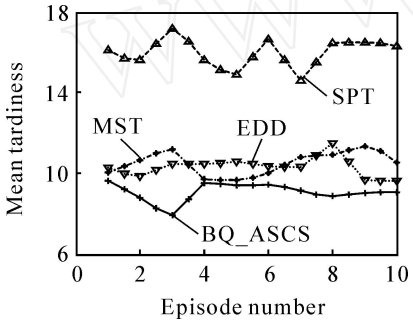

图 2 同一交货因子的平均拖期比较

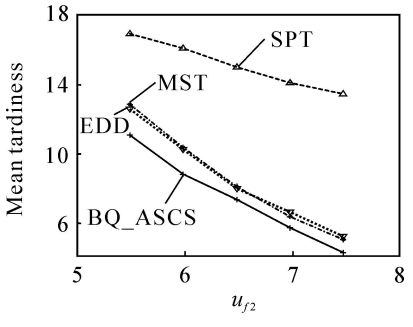

图 3 不同交货因子的平均拖期比较

由表 2 不难看出,BQ_ASCS 策略大部分时间都优于其他 3 个调度规则,对于 500 个 episode,比最好的 EDD 规则提高约 12.18 %,比最差的 SPT 规则提高约 43.34 %. 从图 3 可知,对于不同松紧的交货因子情况,BQ_ASCS 策略均优于其他 3 个调度规则,可以明显地减少工件的平均拖期.

6 结 语

本文针对动态调度中调度知识获取困难的问题,提出了对环境具有较强适应性的 B-Q 学习算法. 该算法无需任何先验知识,通过与知识化制造单元交互来获取调度知识,并对该算法收敛性进行了分析. 基于 B-Q 学习算法构建了自适应调度控制策略,该策略能够动态实时地选取调度规则对工件进行调度. 实验结果表明,在工件交货期变化的情况下,本文给出的调度控制策略明显优于单一调度规则,从而证明了该策略的有效性和可行性.

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 Hong-sen, Liu Fei. Knowledgeable manufacturing system —A new kind of advanced manufacturing system [J]. Computer Integrated Manufacturing Systems, 2001, 7(8): 7-11.)

[2] Yan H S. A new complicated-knowledge representation approach based on knowledge meshes[J]. IEEE Trans on Knowledge and Data Engineering, 2006, 18(1): 47-62.

[3] 钱晓龙, 唐立新, 刘文新. 动态调度的研究方法综述[J]. 控制与决策, 2001, 16(2): 141-145.
(Qian Xiao-long, Tang Li-xin, Liu Wen-xin. Dynamic scheduling: A survey of research methods[J]. Control and Decision, 2001, 16(2): 141-145.)

[4] 徐俊刚, 戴国忠, 王宏安. 生产调度理论和方法研究综述[J]. 计算机研究与发展, 2004, 41(2): 257-267.
(Xu Jun-gang, Dai Guo-zhong, Wang Hong-an. An overview of theories and methods of production scheduling [J]. J of Computer Research and Development, 2004, 41(2): 257-267.)

[5] Baker K R. Sequencing rules and due-date assignments in a job shop[J]. Management Science, 1984, 30(9): 1093-1104.

(下转第 1346 页)

[3] Abilio Lucena, Mauricio G C Resende. Strong lower bounds for the prize collecting Steiner problem in graphs [J]. Discrete Applied Mathematics, 2004, 141: 277-294.

[4] Mohamed Haouari Jouhaina Chaouachi Siala. A hybrid Lagrangian genetic algorithm for the prize collecting Steiner tree problem [J]. Computers & Operations Research, 2006, 33: 1274-1288.

[5] 王安顺, 郭子龙. 混沌免疫组合优化算法[J]. 控制与决策, 2006, 21(2): 205-209.
(Wang An-shun, Guo Zi-long. Novel chaos immune optimization combination algorithm [J]. Control and Decision, 2006, 21(2): 205-209.)

[6] Francisco Barahona, Ranga Anbil. The volume algorithm: Producing primal solutions with a subgradient method [J]. Mathematical Programming, 2000, (87): 385-399.

[7] Bahiense L, Barahona F, Porto O. Solving Steiner tree problems in graphs with Lagrangian relaxation[J]. J of Combinatorial Optimization, 2003, 7: 259-282.

[8] Dell'Amico M, Maffioli F, Sciomachen A. A Lagrangian heuristic for the prize collecting travelling salesman problem[J]. Annals of Operations Research, 1998, 81: 289-305.

[9] Raffensperger J F. Solving the TSP with decomposition-based pricing [C]. Talk Presented at the 18th Int Symposium on Mathematical Programming. Denmark: Copenhagen, 2003: 87-102.

[10] Rafael Marti, Manuel Laguna, Fred Glover. Principles of scatter search [J]. European J of Operational Research, 2006, 169: 359-372.

[11] Hamiez J P, Hao J K. Scatter search for graph coloring [J]. Lecture Notes in Computer Science, 2002, 23(10): 168-179.

[12] 罗家祥, 唐立新. 带释放时间的并行机调度问题的ILS&SS算法[J]. 自动化学报, 2005, 31(6): 917-924.
(Luo Jia-xiang Tang Li-xin. A new ILS & SS algorithm for parallel-machine scheduling problem[J]. Acta Automation Sinica, 2005, 31(6): 917-924.)

[13] Robert A Russell, Wen-Chyuan Chiang. Scatter search for the vehicle routing problem with time windows[J]. European J of Operational Research, 2006, 169: 606-622.

[14] Christian Blum, Matthias Ehrgott. Local search algorithms for the k-cardinality tree problem [J]. Discrete Applied Mathematics, 2003, 128: 511-540.

(上接第 1340 页)

[6] Arzi Y, Iaroslavitz L. Neural network-based adaptive production control system for a flexible manufacturing cell under a random environment[J]. IIE Trans, 1999, 31(3): 217-230.

[7] 万国华. 用模糊调度系统求解动态 Job Shop 问题[J]. 系统工程理论与实践, 2001, 21(8): 97-101.
(Wan Guo-hua. Using fuzzy scheduling system for solving a dynamic job shop problem [J]. Systems Engineering Theory and Practice, 2001, 21(8): 97-101.)

[8] Piramuthu S, Shaw M, Fulkerson B. Information-base dynamic manufacturing system scheduling[J]. Int J of Flexible Manufacturing Systems, 2000, 12(2/3): 219-234.

[9] Kim C O, Jun J, Baek J K, et al. Adaptive inventory control models for supply chain management[J]. Int J of Advanced Manufacturing Technology, 2005, 26(9/10): 1184-1192.

[10] Mcdonnell P, Joshi S, Qiu R G. A learning approach to enhancing machine reconfiguration decision-making games in a heterarchical manufacturing environment [J]. Int J of Production Research, 2005, 43(20): 4321-4334.

[11] Aydin M E, Öztemel E. Dynamic job-shop scheduling using reinforcement learning agents[J]. Robotics and Autonomous Systems, 2000, 33(2): 169-178.

[12] Theodoridis S, Koutroumbas K. Pattern recognition [M]. 2nd Edition. San Diego: Academic Press, 2003.

[13] Singh S, Jaakkola T, Littman M L, et al. Convergence results for single-step on-policy reinforcement-learning algorithms[J]. Machine Learning, 2000, 38(3): 287-308.