文章编号: 1001-0920(2019)07-1456-07 DOI: 10.13195/j.kzyjc.2017.1734

基于RBF-Q学习的多品种CSPS系统前视距离控制

唐 昊†, 杨 羊, 戴 飞, 谭 琦

(合肥工业大学 电气与自动化工程学院，合肥 230009)

摘 要: 研究一类多品种工件到达的传送带给料加工站系统(CSPS)的前视距离(Look-ahead)优化控制问题,以提高系统的工作效率. 在工件品种数增加的情况下,系统状态规模会呈现指数性增长,考虑传统Q学习在面对大规模离散状态空间所面临的维数灾难,且难以直接处理前视距离为连续化变量的问题,引入了RBF网络来逼近Q值函数,网络的输入为状态行动对,输出为该状态行动对的Q值. 给出RBF-Q学习算法,并应用于多品种CSPS系统的优化控制中,实现了连续行动空间的Q学习. 针对不同的品种数情况进行仿真分析,仿真结果表明,RBF-Q学习算法可以对多品种CSPS系统性能进行有效优化,并且提高学习速度.

关键词: RBF网络；Q学习；多品种工件；传送带给料加工站；前视距离

中图分类号: TP278 **文献标志码:** A

Look-ahead control of multi-type products CSPS system based on RBF-Q learning

TANG Hao†, YANG Yang, DAI Fei, TAN Qi

(School of Electrical Engineering and Automation，Hefei University of Technology，Hefei 230009，China)

Abstract: This paper studies the look-ahead optimal control problem of the conveyor-serviced production station (CSPS) system for a class of varieties of parts arrival to improve the efficiency of operations. When the number of varieties of the system increases, the system state scale will show exponential growth. Considering the dimension disaster problem of traditional Q-learning in the face of the large-scale discrete state and the difficulty of dealing with the look-ahead as a continuous variable directly, the RBF network is introduced to approximate the Q value function, the input of the RBF network is the state action pair, and the output is the Q value of the state action pair. The RBF-Q learning algorithm is proposed, and applied to the optimal control of multi-type products conveyor-serviced production station, realized the continuous action space Q-learning. The simulation analysis is carried out for different varieties, and results show that the method can effectively optimize the processing of CSPS system and improve the learning speed.

Keywords: RBF network；Q-learning；multi-type products；conveyor-serviced production station (CSPS)；look-ahead

0 引 言

当前我国制造业迅速发展,而制造业是国民经济支柱,因此2015年国务院正式印发了《中国制造2025》,以推进智能制造为主攻方向,将我国智能制造装备业培育成为具有国际竞争力的先导产业[1]. 智能制造装备在制造过程中能进行智能活动,如分析、推理、判断和决策等[2]. 自动化生产线是智能制造设备的重要内容,在现代化的生产加工企业中,存在一类由生产加工站作为加工主体且物料由传送带传输的生产线,称为传送带给料生产加工站(CSPS)[3-8]. 这类系统可以通过学习一个前视距离(Look-ahead)控制策略来协调站点对工件的捡取与加工,使系统的长期运行代价最小,提高系统工作效率.

随着社会需求的多样化,多品种生产方式成为一种趋势[9]. 文献[10]建立了多品种CSPS系统的SMDP模型,并用策略迭代算法进行求解,该方法依赖于精确的系统模型参数;文献[11]使用基于性能势的多Agent Q学习算法优化两类品种情况下的多站点协同控制问题,该方法以表格的形式存储状态行动对的Q值信息,存在离散粒度难以控制及存储空间消耗过大的问题. 为了实现连续化行动变量的学习,很多研究者作了将神经网络与强化学习相结合的相

收稿日期: 2017-12-20；**修回日期:** 2018-05-12.

基金项目: 国家自然科学基金项目(61573126, 71231004)；中央高校基本科研业务费专项基金项目(JZ2016YYPY0052)；高等学校博士学科点专项科研基金项目(20130111110007).

责任编委: 卢剑权.

†通讯作者. E-mail: htang@hfut.edu.cn.

关研究[12-15]. 例如, 文献[14]将CMAC网络与Q学习相结合, 并应用于单站点CSPS系统的优化控制, 通过CMAC网络来逼近具有连续行动值的Q函数, 但是网络精度有待提高; 文献[15]基于模糊推理系统提出了一种自适应模糊Q学习算法, 该方法根据学习过程中任务的复杂性自动生成和调整模糊规则, 但计算量也相应增加.

在多个品种工件随机到达的CSPS系统中, 系统状态由所有缓存库状态联合确定, 具有系统状态规模会随缓存库增加呈现指数性增长的特点. 对于大规模的离散状态空间系统, 行动空间进行离散化后, 表格存储占用的内存资源过于庞大, 在计算量上过于复杂, 将产生维数灾难问题, 并且行动离散粒度的选择一般缺乏先验性知识.

本文解决问题的思路是将具有较好泛化和逼近能力的RBF网络[16]引入到Q学习当中, 实现行动变量的连续化, 克服理论求解对模型参数的依赖和Q学习占用内存空间大且难以直接处理连续变量情况的缺点. 网络的输入为状态行动对, 输出为状态行动对的Q值. 状态由CSPS系统的多个缓存库空余量联合状态确定, 行动为站点可控的前视距离.

1　多品种CSPS系统

图1为多品种CSPS系统物理模型, 不同品种工件沿传送带运输, 站点可以将工件捡取至相应的缓存库或者对缓存库中工件进行加工, 定义N_m和n_m分别为m品种工件缓冲库的容量和空余量, 其中$m = 1, 2, \cdots, M$, M为工件品种的总个数. 系统状态为所有工件缓存库空余量的联合状态, 其状态空间记为$\Phi = \Phi_1 \times \Phi_2 \times \cdots \times \Phi_M$. 记在系统状态$s = (n_1, n_2, \cdots, n_M)$下采取的行动为$d_s$, 其中, 对于任意$m$, 有$d_{s=(n_1,\cdots,n_m=0,\cdots,n_M)} \equiv 0, d_{s=(N_1,N_2,\cdots,N_M)} \equiv \infty$. 其他情况下, $d_s \in D = (0, l_{\max}]$, D为系统的行动空间, $l_{\max}$为最大前视距离行动.

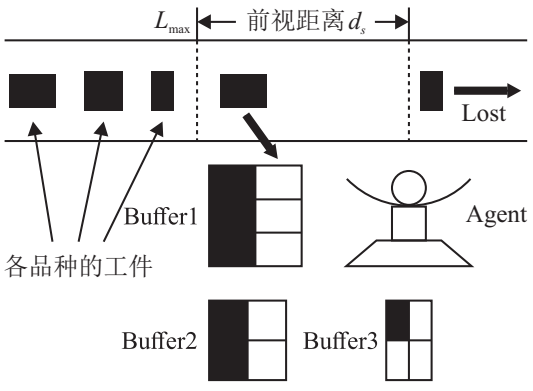

图1　多品种CSPS系统物理模型

假设系统当前控制策略为v, 记初始决策时刻$T_0 = 0$, 系统的状态演变过程表示为$s_t (t \geqslant 0)$, 在第n个决策时刻T_n, 系统状态$s_{T_n} = (n_1, n_2, \cdots, n_M)$, 以下简记为$s_n$, 采取行动$d_{s_{T_n}}$, 以下简记为$d_{s_n}$. 若在前视距离$d_{s_n}$内至少有一个工件到达, 则等待工件到达捡取点并捡取. 记第一个到达的工件品种为m, 到达时间为ς_n, 则下一个决策时刻为$T_{n+1} = T_n + \varsigma_n$, 下一时刻状态为$s_{n+1} = (n_1, \cdots, n_m - 1, \cdots, n_M), m \in \{1, 2, \cdots, M\}$. 若在前视距离$d_{s_n}$内没有工件到达, 则从相对空余量最小的缓冲库中选取一个工件进行加工. 记选取的工件品种为m, 服务时间为τ_n, 则下一个决策时刻$T_{n+1} = T_n + \max\{\tau_n d_{s_n}\}$, 下一时刻状态为$s_{n+1} = (n_1, \cdots, n_m + 1, \cdots, n_M), m \in \{1, 2, \cdots, M\}$. 在加工过程中, 任何到达捡取点的工件都会流失.

令$f(s_n, d_{s_n}, s_{n+1}, t)$表示在行动$d_{s_n}$下系统从状态$s_n$转移到下一状态$s_{n+1}$的期望代价函数, $T_n \leqslant t < T_{n+1}$, 由工件的存储代价、加工代价、系统等待代价和加工报酬组成[5]. 定义在策略v下系统无穷时段平均性能代价为

$$\eta^v = E\Big[\lim_{N\to\infty} \frac{1}{T_N} \sum_{n=0}^{N-1} \int_{T_n}^{T_{n+1}} f(s_n, d_{s_n}, s_{n+1}, t)\mathrm{d}t\Big]. \tag{1}$$

多品种CSPS系统优化的目标是找到一个最优策略v^*, 使得系统的无穷时段平均性能代价η^{v^*}最优.

不失一般性, 本文作以下假设:

1) 系统运行过程中, 工件的捡取时间和工件加工完成后放入成品库的时间忽略不计;

2) 传送带匀速运行, 前视距离可等效为前视时间;

3) 任一缓存库为满时, 站点直接从满的缓存库取一个工件加工, 故系统不会出现多个缓存库为满的情况;

4) 为了平衡生产, 站点从相对空余量最小的缓存库中取一个工件加工, 若有多个相对空余量相同时, 则从其中选取一个品种缓存库工件加工(相对空余量定义为缓存库空余量与缓存库容量的比值).

2　RBF-Q学习网络

多品种CSPS系统具有离散状态规模庞大、行动连续的特点. 其离散状态规模与缓存库个数和各缓存库容量有确定的关系, 即状态个数$n_s = (N_1 + 1) \times (N_2 + 1) \times \cdots \times (N_M + 1) - I$, I是假设中不存在的多个为满的状态个数. 通常的优化方法会存在一些缺点, 例如理论求解算法不仅模型参数难以完全

获取,转移概率函数也很难建立. 而离散化方法需选择一个离散粒度Δ对连续行动进行离散,离散行动数$n_d = l_{\max}/\Delta + 2$,且需要用查询表存储所有的状态行动对信息,内存资源消耗巨大. 例如,在后文仿真案例中,当品种数$M = 4$时,根据相应参数设置,可计算出系统状态个数$n_s = (N_1+1)\times(N_2+1)\times(N_3+1)\times(N_4+1) - 122 = 488$,相应行动数$n_d = 1/0.05 + 2 = 22$,构成的状态行动对个数$488\times 22 = 10\,736$,由此构成的策略空间非常庞大. 如采用直接搜索法,则需要搜索的策略总数为$22^{240}+248$(部分状态行动唯一),计算量相当复杂,搜索效率和收敛性难以得到保证,且离散粒度难以控制,量化不当会严重影响CSPS系统的优化性能.

1988年,Broohead等[16]提出了径向基函数(Radial basis function,RBF)神经网络,由于RBF网络具有较好的逼近能力和泛化能力等特点,在函数逼近中得到广泛的应用. 因此,本文将RBF网络与Q学习算法相结合,用于多品种CSPS系统前视距离连续变量的优化控制中.

2.1 网络结构

利用RBF网络逼近Q值函数的网络结构如图2所示,该网络为4层组织结构.

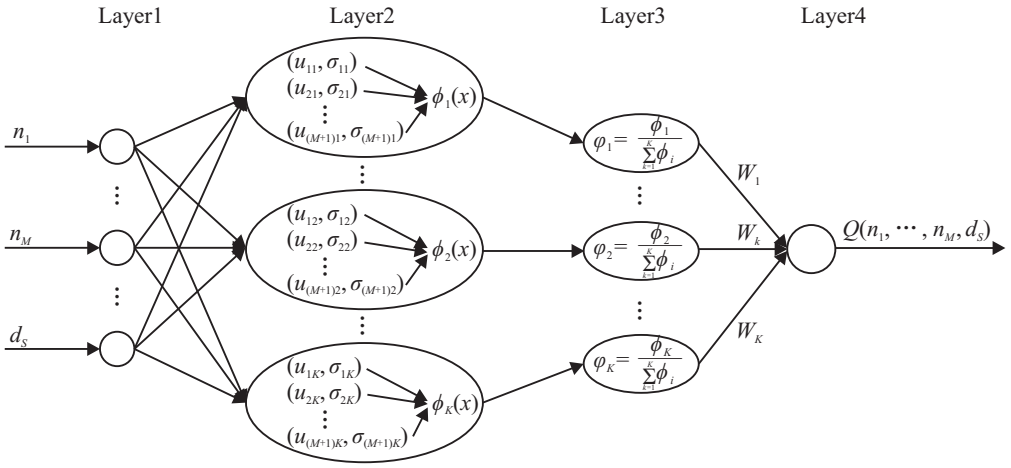

图2 RBF-Q学习网络结构

第1层为输入层,该层共有$M+1$个神经元,每个神经元对应一个输入分量. 其中:前M个分量为多品种CSPS系统的状态变量$s = (n_1, n_2, \cdots, n_M)$,最后一个分量是在该状态下的行动$d_s$. 记网络的输入为$x = (x_1, x_2, \cdots, x_M, x_{M+1}) = (n_1, n_2, \cdots, n_M, d_s)$,即状态行动对.

第2层为隐层,该层共有K个节点,每个节点为$M+1$维高斯函数,对应第一层$M+1$维输入. 第k个节点函数输出表示为

$$\phi_k(x) = \exp\left(-\sum_{i=1}^{M+1}\frac{(x_i-u_{ik})^2}{2{\sigma_{ik}}^2}\right),\quad k = 1,2,\cdots,K, \tag{2}$$

其中u_{ik}和σ_{ik}分别为第k个节点函数的中心和宽度.

第3层为归一化层,与隐层对应为K个节点,对规则进行归一化操作,有

$$\varphi_k(x) = \frac{\phi_k(x)}{\sum_{k=1}^{K}\phi_k(x)},\ k = 1,2,\cdots,K. \tag{3}$$

第4层为输出层,该层只有一个节点,用于逼近状态-行动对值函数. 第k个隐节点与输出节点的连接权值为w_k,网络输出为

$$Q(\Theta, s, d) = \sum_{k=1}^{K} w_k\varphi_k(x). \tag{4}$$

其中:Θ是网络参数合集,包括输出权重w、数据中心u、宽度σ.

对于上面给出的4层网络结构,只要输入当前的CSPS系统缓存库的状态和对应的行动,就可以输出该状态行动对的Q值. 除了前文提到的特殊情况(存在任意一品种缓存库为满或者缓存库全空)行动唯一,在与环境的交互学习中,行动的选择根据所学的Q值函数按一定的策略给出. 在系统状态为$s = (n_1, n_2, \cdots, n_M)$时,选择的贪婪行动$d_s^{\text{greeedy}}$为

$$d_s^{\text{greedy}} = \arg\min_{d\in D} Q(\Theta, s, d) = \arg\min_{d\in D}\frac{\sum_{k=1}^{K} w_k \mathrm{e}^{-\sum_{i=1}^{M}\frac{(x_i-u_{ik})^2}{2\sigma_{ik}^2}}\mathrm{e}^{-\frac{x_{M+1}-u_{(M+1)k}}{2\sigma_{(M+1)k}^2}}}{\sum_{k=1}^{K}\mathrm{e}^{-\sum_{i=1}^{M}\frac{(x_i-u_{ik})^2}{2\sigma_{ik}^2}}\mathrm{e}^{-\frac{x_{M+1}-u_{(M+1)k}}{2\sigma_{(M+1)k}^2}}}.$$

网络输入的最后一个分量为行动变量,即$x_{M+1} = d, d\in D$. 在知道当前状态$s = (n_1, n_2, \cdots, n_M)$时,可以令$c_M = \mathrm{e}^{-\sum_{i=1}^{M}\frac{(x_i-u_{ik})^2}{2\sigma_{ik}^2}}$,$c_M$是一个常数项,于是有

$$d_s^{\text{greedy}} = \arg\min_{d\in D}\Big(\frac{\sum_{k=1}^{K} w_k c_M \exp\Big(-\frac{d-u_{(M+1)k}}{2\sigma^2_{(M+1)k}}\Big)}{\sum_{k=1}^{K} c_M \exp\Big(-\frac{d-u_{(M+1)k}}{2\sigma^2_{(M+1)k}}\Big)}\Big). \quad (5)$$

行动探索对学习而言是非常重要的,特别是在初始阶段,为了平衡学习过程中对行动的探索与利用,本文采用ε-greedy贪婪策略,即在当前状态$s=(n_1, n_2, \cdots, n_M)$下,采取的行动$d_s$以$\varepsilon$的概率选择随机的探索行动$d^*$,以$1-\varepsilon$的概率选择贪婪行动$d_s^{\text{greeedy}}$. 本文的仿真实验中,探索概率$\varepsilon$是随着学习步数按一定曲线衰减的,下式给出了具体曲线参数:

$$\varepsilon_n = \begin{cases} \varepsilon_1,\ N \leqslant 0.2N; \\ \varepsilon_1 \exp(-\gamma_\varepsilon(n-0.2N)),\ n > 0.2N. \end{cases} \quad (6)$$

其中: $\gamma_\varepsilon = -2\ln(\varepsilon_2/\varepsilon_1)/N$, $\varepsilon_1 = 0.6$表示初始阶段探索率, $\varepsilon_2 = 0.15$表示中期阶段探索率, N表示总的学习步数.

2.2 网络参数学习

神经网络结构一般需要根据网络的输出与目标函数或者目标样本之间的误差来调整网络的参数. 在学习当中,需要通过与环境的交互来获得相应状态行动对的Q值误差,并以Q值误差作为与目标输出的误差来调整网络参数,实现值函数的逼近. 假设CSPS系统在运行时刻T_n,状态为s_n,采取行动d_{s_n},转移到下一状态s_{n+1},可以观测到一个转移样本轨道$\langle s_n, d_{s_n}, s_{n+1}, \omega_n, \tau_n\rangle$, ω_n为系统状态转移过程中实际的逗留时间, τ_n为在服务工件过程中的加工时间. 根据文献[5],转移过程的即时差分公式c_n为

$$c_n = f'(s_n, d_{s_n}, s_{n+1}) - T_\alpha(\omega_n)\eta_n + \mathrm{e}^{-\alpha\omega_n}\min_{d\in D} Q_\alpha(\Theta, s_{n+1}, d) - Q_\alpha(\Theta, s_n, d_{s_n}). \quad (7)$$

其中: $T_\alpha(x) = \int_0^x \mathrm{e}^{-\alpha t}\mathrm{d}t, x \geqslant 0$,显然$T_0(x) = x$, α为折扣因子, $\alpha > 0$; η_n是平均代价η^v的估计值; $f'(s_n, d_n, s_{n+1})$是系统从T_n时刻到T_{n+1}时刻转移过程中的累积折扣代价.

系统进行工件捡取时,有

$$f'(s_n, d_{s_n}, s_{n+1}) = T_\alpha(\omega_n)\Big[\sum_{m=1}^{M}[k_1^m\cdot(N_m - n_m)] + k_3\Big]. \quad (8)$$

其中: k_1^m表示单位时间存储m品种工件的代价, k_3表示单位时间站点的等待代价.

系统进行工件加工时,有

$$f'(s_n, d_{s_n}, s_{n+1}) = T_\alpha(\omega_n)\Big[\sum_{t=1,t\neq m}^{M}[k_1^t\cdot(N_m - n_m)] + k_1^m(N_m - n_m - 1)\Big] + k_2^m\cdot T_\alpha(\tau_n) + k_3\cdot(T_\alpha(\omega_n) - T_\alpha(\tau_n)) + k_4^m\cdot \mathrm{e}^{-\alpha\tau_n}. \quad (9)$$

其中: k_2^m表示单位时间加工m品种工件的代价; k_4^m表示加工完成m品种工件的即时报酬,为一个负数.

基于式(7)的即时差分,利用梯度下降法进行可调参数学习[17],包括隐层单元各个基函数的中心、宽度以及隐层至输出层之间的连接权值. 结合式(2)~(4),可得各参数的更新公式如下所示:

$$w_k(n+1) = w_k(n) + \gamma_w c_n \varphi_k,\ k = 1, 2, \cdots, K; \quad (10)$$

$$u_{ik}(n+1) = u_{ik}(n) + \gamma_u c_n w_k(n)\varphi_k(1-\varphi_k)\frac{x_i - u_{ik}(n)}{\sigma_{ik}^2(n)},\ i = 1, 2, \cdots, M+1,\ k = 1, 2, \cdots, K; \quad (11)$$

$$\sigma_{ik}(n+1) = \sigma_{ik}(n) + \gamma_\sigma c_n w_k(n)\varphi_k(1-\varphi_k)\frac{(x_i - u_{ik}(n))^2}{\sigma_{ik}^3(n)},\ i = 1, 2, \cdots, M+1,\ k = 1, 2, \cdots, K. \quad (12)$$

这里有一个特殊情况,即系统当前时刻状态为所有缓存库为空, $s = (N_1, N_2, \cdots, N_M)$,前视距离为$d_s = \infty$时,无穷大作为网络输入会引起网络不稳定,于是对该状态行动对的Q值进行单独存储与学习更新. 其更新公式(γ为学习步长)如下:

$$Q_\alpha^*(s = (N_1, N_2, \cdots, N_M), \mathrm{d}_s) := Q_\alpha^*(s = (N_1, N_2, \cdots, N_M), d_s) + \gamma c_n. \quad (13)$$

下面给出RBF-Q学习算法的详细步骤.

Step 1: 初始化网络权值w_i、中心u_{ik}、宽度σ_{ik}、学习步数N以及学习率γ_w、γ_u、γ_σ.

Step 2: 在T_n时刻,观察得到当前系统状态s_n,若为一般状态,转入Step 3;若为特殊状态,转入Step 4.

Step 3: 在当前状态下,通过式(5)计算出贪婪行动d_s^{greedy},根据当前ε-greedy策略,以$1-\varepsilon$的概率选择贪婪行动d_s^{greedy},以ε的概率选择随机的探索行动d^*,系统转移到下一实际状态s_{n+1},转入Step 5.

Step 4: 直接执行相应的行动,系统转移到下一实际状态s_{n+1},转入Step 5.

Step 5: 记录系统逗留时间ω_n及服务时间τ_n,通过式(7)计算出即时差分c_n.

Step 6: 若s_n为所有缓存库全空状态,则根据式(13)更新该状态行动对Q值;否则,根据式(10)~(12)更新网络参数权值w_i、中心u_{ik}、宽度σ_{ik}.

Step 7: $n := n+1$,若$n > N$,则结束学习;否则,转入Step 2.

3 实验结果

为了充分比较Q学习与RBF-Q学习在多品种CSPS系统的优化性能,本文分别取品种数$M=2$、$M=3$和$M=4$三种情况进行对比. 其系统状态数分别为$n_s=(N_1+1)\times(N_2+1)-1=19$,$n_s=(N_1+1)\times(N_2+1)\times(N_3+1)-12=88$,$n_s=(N_1+1)\times(N_2+1)\times(N_3+1)\times(N_4+1)-122=488$. 缓存库容量对应为后文系统参数的设置情况.

仿真实验中,站点的前视距离作为行动,本身为连续变量. 而在用传统Q学习进行优化时,需先对行动进行离散化,而离散粒度选择没有任何的先验知识. 因此为了选择一个合适的离散粒度,需先对各品种数情况下分别进行不同离散粒度的仿真实验.

本文以品种数$M=3$时为例,分别取离散粒度Δ为0.1、0.07、0.05、0.02四种情况进行比较. 在仿真时,涉及到的关于CSPS系统的参数有缓存库的容量N_m、各品种的工件到达率λ_m及服务率u_m,需要保证这些参数相同. 设站点加工m品种工件的时间服从服务率为u_m的L阶的Erlang分布,缓存库容量N_m的设置符合实验室生产线的实际情况和最优缓存库容量配比[10]. $M=3$时,设置的参数如表1所示.

表1 $M=3$时的相关参数设置

参数	$l_{\max}$	u_1	u_2	u_3	L	λ_1	λ_2	λ_3	N_1	N_2	N_3
值	1	4	5	8	4	0.2	0.3	0.5	3	4	4

图3是在表1的参数下的仿真结果,纵轴表示当前Q值下贪婪策略的平均性能代价的评估值,评估时进行10次独立实验,每次独立实验时系统仿真运行1×10^4步,然后取统计平均值. 设定$\Delta=0.1$时总学习步数$N=5\times10^5$,$\Delta=0.07$和$\Delta=0.05$时总学习步数$N=1\times10^6$,$\Delta=0.02$时总学习步数$N=2.4\times10^6$. 综合最后的优化结果与收敛速度来看,$M=3$时,$\Delta=0.05$是较优的选择,在学习步数$n=4\times10^5$左右后,平均性能代价基本稳定在一个范围内,此时$\eta^{v^*}=-5.785\pm0.007$.

表2给出了品种数$M=2$时的相关参数. 同样,选择了综合比较后一个较优离散粒度$\Delta=0.1$,对应的Q学习算法优化曲线如图4所示. 设定总学习步数$N=2.8\times10^5$,在学习步数$n=8\times10^4$左右后,平均性能代价基本稳定在$\eta^{v^*}=-4.593\pm0.008$.

表2 $M=2$时的相关参数设置

参数	$l_{\max}$	u_1	u_2	L	λ_1	λ_2	N_1	N_2
值	1	4	5	4	0.4	0.6	3	4

表3给出了品种数$M=4$时的相关参数. 同样,选择了综合比较后一个较优离散粒度$\Delta=0.05$,对应的Q学习算法优化曲线如图5所示. 设定总学习步数$N=2.8\times10^6$,在学习步数$n=8\times10^5$左右后,平均性能代价基本稳定在$\eta^{v^*}=-5.904\pm0.007$.

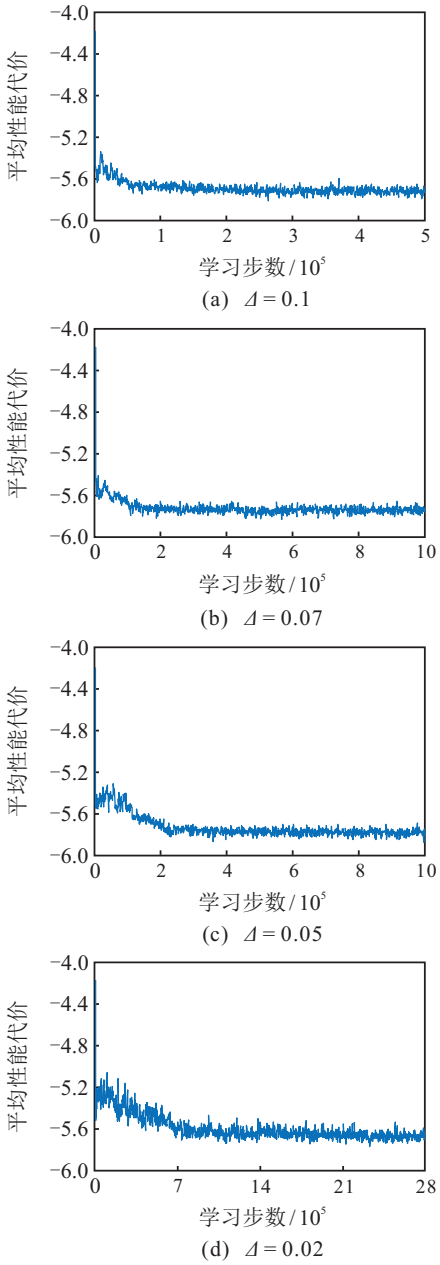

图3 $M=3$时,不同离散粒度Q学习平均性能代价

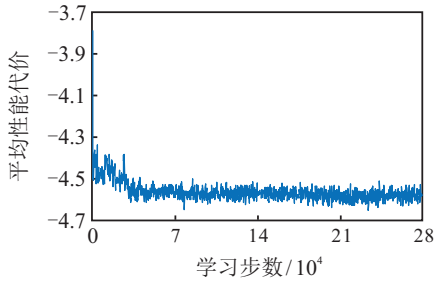

图4 $M=2$时,Q学习平均性能代价

表3 $M=4$时的相关参数设置

参数	$l_{\max}$	u_1	u_2	u_3	u_4	L	λ_1	λ_2	λ_3	λ_4	N_1	N_2	N_3	N_3
值	1	6	6.5	7	7.5	4	0.1	0.2	0.3	0.4	3	4	4	5

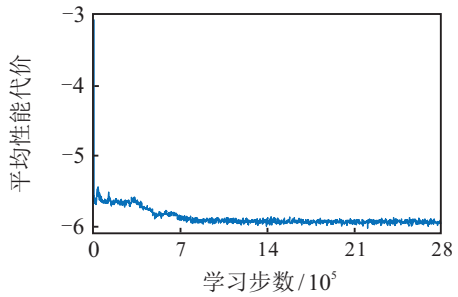

图5　$M=4$时,Q学习平均性能代价

以上是3种不同品种数情况下,经过多次不同离散粒度仿真实验后,选择一个较优离散粒度的Q学习的优化曲线. 对应于上面表格给出的系统参数,图6给出了相应不同品种数情况下,RBF-Q学习算法的平均性能代价优化曲线. 在$M=2$和$M=3$时,设定总学习步数$N=2\times10^4$,即算法每200步进行一次当前贪婪策略的评估. 在$M=4$时,设定总学习步数$N=5\times10^4$,即算法每500步进行一次当前贪婪策略的评估. 可以看出:$M=2$时,在学习步数$n=6\times10^3$后,RBF-Q学习算法平均性能代价稳定在$\eta^{v*}=-4.623\pm0.008$;$M=3$时,在学习步数$n=6\times10^3$后,平均性能代价基本稳定在$\eta^{v^*}=-5.805\pm0.007$;$M=4$时,在学习步数$n=1.5\times10^4$后,平均性能代价基本稳定在$\eta^{v^*}=-5.911\pm0.007$. RBF-$Q$学习优化曲线整体下降平稳,波动较小,收敛速度较快.

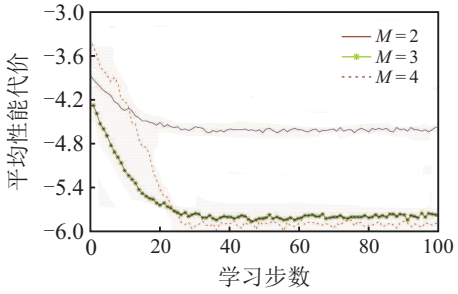

图6　不同品种数下RBF-Q学习平均性能代价

图7给出了两种算法在不同品种数情况下收敛时的学习时间(不包括评估的时间)和系统状态规模. 可以看出,随着工件品种数的增加,系统状态数呈指数增长的趋势,状态规模复杂. 相应地,Q学习算法的优化曲线收敛的学习时间有显著增加的趋势,这与Q学习需要用表格存储全部状态离散行动和状态行动对的Q值信息有关. 而RBF-Q学习算法优化曲线收敛的学习时间仅略微增加,并且收敛时间远远小于Q学习,这很好地体现了RBF网络的信息泛化能力强、学习速度快的特点.

图8给出了两种优化算法在不同工件品种数情况下空间复杂度的对比情况. 从图8可以看出,随着品种数的增加,RBF-Q学习算法比Q学习算法节省

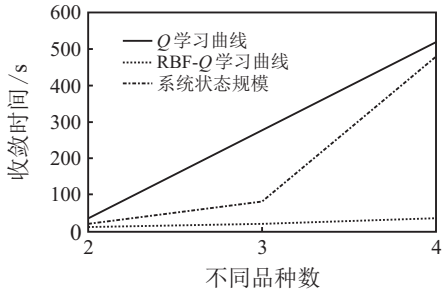

图7　不同品种数下算法的收敛时间与系统状态规模

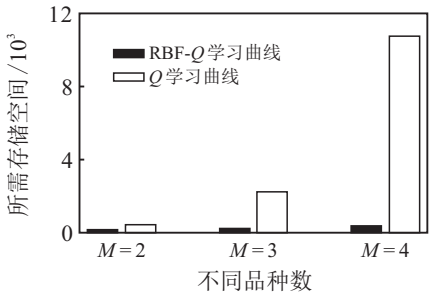

图8　不同品种数下算法的空间复杂度对比

存储资源的优势变明显. 例如在$M=4$时,Q学习算法需要存储每个状态行动对的Q值信息,所需的存储空间为$n_s\times n_d=488\times\left(\frac{1}{0.05}+2\right)$,即10 736个;而RBF-$Q$学习算法需要存储网络的参数信息,包括权值、中心和宽度信息,在$M=4$时,隐节点数$K=12$,可以计算出所需存储空间为132个,大大节省了计算机存储资源.

为了便于对比,表4给出了详细的结果对比. 表4中存储规模一栏,Q学习列出的是需要存储的状态行动对Q值信息的表格大小(离散粒度取仿真实验中较优的),RBF-Q学习列出的是网络隐节点数K;生产效率一栏定义为系统在T时间段内所有品种工件加工数量与到达工件总数的比值.

表4　结果对比

M	算法	收敛步数	时间/s	平均性能代价	存储规模	生产效率/%
2	Q	8×10^4	27.46	-4.593 ± 0.008	19×12	70.41
	RBF-Q	6×10^3	6.02	-4.625 ± 0.008	7	71.65
3	Q	8×10^5	258.43	-5.785 ± 0.007	88×22	75.17
	RBF-Q	6×10^3	8.35	-5.805 ± 0.007	9	75.88
4	Q	8×10^5	530.17	-5.904 ± 0.007	488×22	76.21
	RBF-Q	1.5×10^4	22.79	-5.911 ± 0.007	12	76.93

由表4可以看到,随着品种数增加,生产效率增高. 这与在工件总到达率相同情况下,系统可存最大容量变大,流失工件会相应减少有关. 同时,在相同品种数下,RBF-Q学习的生产效率略高于Q学习. 从优化结果的平均性能代价可以看出,Q学习算法和RBF-Q学习算法对多品种CSPS系统都能有一个很好的性能优化,提高了系统运行效率. RBF-Q学习比

Q学习平均性能代价要好0.1 %～0.65 %,这是因为Q学习算法需要对行动进行离散化处理,而这种处理会产生离散化误差,并且学习过程中没有遍历过的状态行动对的Q值不会发生变化;而RBF网络能够处理连续化变量作为输入,有较强的信息泛化能力,因而获得较好的优化效果,这体现了连续变量控制的优势.

4 结 论

针对多品种CSPS系统状态规模庞大,且Agent前视距离为连续变量的特点,本文将RBF-Q学习算法应用到多品种CSPS系统前视距离优化控制中,充分发挥了RBF网络较好的泛化能力和学习速度快的优势. 该算法不需要对系统的前视距离行动进行离散化,从而避免了离散粒度选择好坏对系统优化效果影响的问题,也不需要用表格存储规模庞大的状态行动对Q值信息,避免了占用过多的内存资源. 仿真结果表明,该算法对多品种CSPS系统有良好的优化效果,并提高了学习速度. 本文算法应用于多品种CSPS系统的品种数还是较少,当品种数目更多时,或者考虑多站点等更为复杂的情况,直接采用本文方法可能会有困难. 因此,考虑将深度神经网络与强化学习相结合将是一个有意义的研究方向.

参考文献(References)

[1] 周济. 智能制造——“中国制造2025”的主攻方向[J]. 中国机械工程, 2015, 26(17): 2273-2284.
(Zhou J. Intelligent — Main direction of “made in china 2025”[J]. China Mechanical Engineering, 2015, 26(17): 2273-2284.)

[2] 傅建中. 智能制造装备的发展现状与趋势[J]. 机电工程, 2014, 31(8): 959-962.
(Fu J Z. Development status and trend of intelligent manufacturing equipment[J]. J of Mechanical & Electrical Engineering, 2014, 31(8): 959-962.)

[3] Matsui M. A generalized model of convey-serviced production station (CSPS)[J]. J of Japan Industrial Management Association, 1993, 44(1): 25-32.

[4] Matsui M. CSPS model: Look-ahead controls and physics[J]. Int J of Production Research, 2005, 43(10): 2001-2025.

[5] Tang H, Arai T. Look-ahead control of conveyor-serviced production station by usingăpotential-based online policyăiteration[J]. Int J of Control, 2009, 82(10): 1917-1928.

[6] Yamada T, Satomi K, Matsui M. Strategic selection of assembly systems under viable demands[J]. Assembly Automation, 2006, 26(4): 335-342.

[7] Matsui M. Manufacturing and service enterprise with risks[M]. New York: Springer Science+Business Media, 2009: 1.

[8] Shen W J, Duenyas I, Kapuscinski R. Optimal pricing, production, and inventory for new product diffusion under supply constraints[J]. Manufacturing & Service Operations Management, 2014, 16(1): 1523-1614.

[9] Zhou Y, Chen C. New settlement projects for multiple type, single part, small batch production and seldom type, large batch production — Review of IMTS 2006(II)[J]. Manufacturing Technology & Machine Tool, 2007, 29(5): 28-36.

[10] Zhou Y M, Tang H, Zhou L, et al. Optimal control model of single conveyor-serviced production station with multi-type products[C]. The 34th Chinese Control Conf. Hangzhou, 2015: 2667-2672.

[11] 唐昊, 李博川, 王彬, 等. 两类品种工件混流的多站点CSPS系统优化控制[J]. 控制与决策, 2017, 32(9): 1614-1620.
(Tang H, Li B C, Wang B, et al.Optimal control of multiple CSPS system with two-type product mixed flow[J]. Control and Decision, 2017, 32(9): 1614-1620.)

[12] Tang Y, He H, Ni Z, et al. Fuzzy-based goal representation adaptive dynamic programming[J]. IEEE Trans on Fuzzy Systems, 2016, 24(5): 1159-1175.

[13] Er M J, Deng C. Online tuning of fuzzy inference systems using dynamic fuzzy Q-learning[J]. IEEE Trans on Systems, Man, & Cybernetics Part B: Cybernetics, A Publication of the IEEE Systems Man & Cybernetics Society, 2004, 34(3): 1478-1489.

[14] 周雷, 孔凤, 唐昊, 等. 小脑模型关节控制器网络在传送带给料生产加工站学习优化控制中的应用[J]. 控制理论与应用, 2011, 28(11): 1665-1670.
(Zhou L, Kong F, Tang H, et al. The application of the cerebellar model joint controller network in the learning optimization control of the transmission and processing station[J]. Control Theory & Applications, 2011, 28(11): 1665-1670.)

[15] Xu M L, Xu W B. Fuzzy Q-learning in continuous state and action space[J]. The J of China Universities of Post and Telecommunications, 2010, 17(4): 100-109.

[16] Broomhead D S, Lowe D. Multivariable functional interpolation and adaptive networks[J]. Complex Systems, 1988, 2(3): 321-355.

[17] Snyman J A. Practical mathematical optimization: An introduction to basic optimization theory and classical and new gradient-based algorithms[M]. New York: Springer, 2005: 249.

作者简介

唐昊(1972−), 男, 教授, 博士生导师, 从事离散事件动态系统、强化学习等研究, E-mail: htang@hfut.edu.cn;

杨羊(1993−), 男, 硕士生, 从事离散事件动态系统、强化学习的研究, E-mail: yyang@mail.hfut.edu.cn;

戴飞(1989−), 男, 博士生, 从事离散事件动态系统、强化学习、自动化生产线的研究, Email: df8992@163.com;

谭琦(1985−), 男, 讲师, 博士, 从事多目标优化、生产优化与控制等研究, E-mail: tanqi@hfut.edu.cn.

(责任编辑: 齐 霁)