DOI:10.19951/j.cnki.1672-9331.20221011001 文章编号:1672-9331(2023)02-0027-08
引用格式:周一帆,郭凯,李帮诚.基于多智能体强化学习的多部件系统维修优化[J].长沙理工大学学报(自然科学版),2023,20(2):27-34.
Citation: ZHOU Yifan, GUO Kai, LI Bangcheng. Maintenance optimization of multi-component system based on multi-agent reinforcement learning[J].J Changsha Univ Sci Tech (Nat Sci),2023,20(2):27-34.

基于多智能体强化学习的多部件系统维修优化

周一帆,郭凯,李帮诚
(东南大学 机械工程学院,江苏 南京 211189)

摘 要:【目的】研究多智能体强化学习算法用于多部件生产系统维修优化的有效性,及维修优化领域知识用于强化学习的可行性。**【方法】**将生产系统的维修决策建模为马尔可夫决策过程(Markov decision process,MDP),并采用一种基于奖励塑造的分布式Q学习(shaped reward distributed Q-learning,SR-DQL)算法对其进行求解。通过对智能体的设计和奖励塑造,把维修优化的领域知识应用于强化学习中。**【结果】**使用包含5个生产单元和4个缓冲库存的生产系统对本文所提出的SR-DQL算法进行验证。相较于Q学习算法,SR-DQL算法能够提升6%的平均收益。此外,由该算法计算得到的平均收益也比由分布式Q学习算法和深度强化学习算法计算得到的大。**【结论】**多智能体强化学习能有效处理大规模生产系统的维修优化问题,添加奖励塑造可以提升算法性能,并得到更优的维修策略。

关键词:多部件生产系统;奖励塑造;分布式Q学习;多智能体强化学习;深度强化学习
中图分类号:TH17 **文献标志码:**A

0 引言

多部件系统的维修优化是可靠性领域的研究热点。维修理论普遍认为这些部件间存在多种相关性,因而难以确定合理的维修策略。研究多部件系统维修优化可以帮助企业节省成本并提高系统的可靠性。

部分研究通过给定策略结构简化维修优化问题,其中最为典型的是设定一组预防性维修门限值。LIU等[1]将煤炭运输系统建模为串并联系统,使用蚁群算法确定了不同部件的预防性维修阈值。YUAN等[2]研究了一个由3个生产单元和2个缓冲库存组成的制造系统,并比较了两门限策略和四门限策略的结果。RASMEKOMEN等[3]研究了一个工业冷箱系统,并使用模拟退火算法优化了各部件的预防性维修门限值及状态检查周期。马维宁等[4]考虑部件间的随机相关性,以最大化系统可用度为目标,采用人工蜂群算法确定了多部件系统的视情维修策略,该策略由各部件的预防性维修门限值、机会维修门限值及检测间隔期组成。虽然以上研究获得了较为合理的维修策略,但其所采用的维修策略结构的优越性,通常没有得到严格证明。

为避免维修策略结构的限制,部分研究者采用马尔可夫决策过程(Markov decision process,MDP)进行维修策略优化,得到不同系统状态下的最优维修动作。当维修优化问题的状态转移模型已知且状态和动作空间较小时,可以通过基于模型的方法(如值迭代、策略迭代等)求解MDP模型。KARAMATSOUKIS等[5]针对由2个生产单元和1个缓冲库存组成的制造系统,建立了离散时间MDP模型并使用策略迭代得到了最优维修策略。此外,作者还研究了最优策略的结构,证明当缓冲库存水平及一个生产单元的退化量固定时,另一个生产单元存在一个最优预防性维修门限值。OLDE KEIZER等[6]采用动态规划算法优化了考虑经济相关性的K-out-of-N系统的维修策略,并研

收稿日期:2022-10-11;**修回日期:**2022-12-18;**接受日期:**2022-12-28
基金项目:国家自然科学基金资助项目(72071044)
通信作者:周一帆(1981—)(ORCID:0000-0002-2898-0632),男,教授,主要从事可靠性、维修优化方面的研究。
E-mail:yifan.zhou@seu.edu.cn

究了最优策略的性质。UIT HET BROEK等[7]考虑维修活动和生产活动的联系,对系统的视情维修和生产策略进行了联合优化。作者将联合优化问题建模为MDP模型,并使用值迭代算法进行求解。XU等[8]研究了K-out-of-N:G系统的视情维修策略,将问题建模为离散时间MDP模型。作者将连续的状态空间离散化,并使用值迭代算法进行求解。ZHENG等[9]研究了两部件系统视情维修和备件库存的联合优化问题,采用离散时间MDP进行建模,并使用值迭代算法进行求解。WANG等[10]同样研究了两部件系统的视情维修和备件库存联合优化。作者将问题建模为半马尔可夫决策过程。NAJAFI等[11]针对考虑经济相关性的两部件系统,建立了半马尔可夫决策过程模型。作者使用比例风险模型描述了退化量和系统失效的关系。上述基于模型的方法(值迭代、策略迭代等)可以求出理论上的最优策略,但其不适用于状态转移模型未知或状态空间和动作空间过大的情况。近年来,也有学者采用近似的基于模型的方法求解MDP。文献[12]使用聚合方法对系统状态进行缩减,并使用值迭代算法求得了近似最优维修策略。文献[13]使用分解MDP克服了维修优化中的"维度诅咒"。然而此类方法受限于问题特性,不具有通用性。

本文研究了一个包含缓冲库存的多单元串联制造系统,将其维修优化建模为MDP模型,并提出一种基于奖励塑造的分布式Q学习算法(shaped reward distributed Q-learning,SR-DQL)对其进行求解。其中,多智能体强化学习可有效处理系统状态空间和动作空间呈指数增长的问题,具有良好的可扩展性。奖励塑造将系统总的奖励与各生产单元奖励结合后传递给每个智能体,从而得到全局最优策略。在智能体设计和奖励塑造过程中,本文充分考虑了维修优化的领域知识以提高强化学习(reinforcement learning,RL)算法的性能。

1 生产系统模型

带缓冲库存的串联生产系统如图1所示。其中,方形代表生产单元,圆形代表缓冲库存。缓冲库存用于暂存上游生产单元的产品,并提供给下游生产单元。缓冲库存的加入也使得生产系统不会因为单个生产单元维修或发生故障而停产。这类系统广泛存在于实际生产中,例如轮胎生产系统以及瓷砖制造系统等[14]。

图1 带缓冲库存的制造系统

Fig. 1 Manufacturing system with buffer stock

图1中$M_n(n \in 1,2,\cdots,N)$为生产单元,$B_n(n \in 1,2,\cdots,N-1)$为缓冲库存。将$B_n$在$t$时刻的库存量表示为$K_n(t)=0,1,\cdots,N_{n,B}$,其中$N_{n,B}$为缓冲库存$B_n$的库存容量。本文假设各生产单元的退化都服从离散时间、离散状态的马尔可夫过程。生产单元M_n的退化量可以表示为离散状态$\{1,2,\cdots,D_n\}$,其中,1表示全新状态,D_n表示故障状态。如果生产单元M_n的上游缓冲库存为0,则M_n处于饥饿状态;如果M_n的下游缓冲库存达到上限,则M_n被阻塞。特别地,M_1不会饥饿,M_N不会被阻塞。

将生产单元M_n的生产率表示为v_n。当生产单元M_n处于故障、饥饿、阻塞或维修中时,$v_n=0$;其余时刻,$v_n=v_{n,\mathrm{norm}}$。假设B_n在当前时刻的库存量为K_n,则其在下一时刻的库存量K_n'为:

$$K_n' = K_n + v_n - v_{n+1} \tag{1}$$

本文假设M_n的退化过程受其生产率v_n的影响,在M_n正常工作时,其状态转移矩阵为$\boldsymbol{P}_{n,\mathrm{norm}}$;在$M_n$处于饥饿或阻塞状态时,其状态转移矩阵为$\boldsymbol{P}_{n,\mathrm{idle}}$。

本文仅考虑预防性维修和事后维修两种维修活动。这两种维修活动都可以使生产单元恢复到全新状态。假设维修时长服从几何分布,在一个单位时间内,生产单元M_n预防性维修及事后维修的完成概率分别为$p_{n,\mathrm{PM}}$、$p_{n,\mathrm{CM}}$,且$p_{n,\mathrm{CM}}<p_{n,\mathrm{PM}}$。对应的预防性维修及事后维修的成本分别为$C_{n,\mathrm{PM}}$、$C_{n,\mathrm{CM}}$,且$C_{n,\mathrm{PM}}<C_{n,\mathrm{CM}}$。单位时间内$M_n$的运行成本为$C_{n,\mathrm{OP}}$。缓冲库存$B_n$在单位时间内持有一个部件的成本为$C_{n,\mathrm{H}}$。系统每生产一个部件会产生收益$R_\mathrm{r}$。维修优化的目标是使单位时间内收益的平均期望最大化。

2 MDP建模

MDP模型可以由四元组$<\psi,\zeta,\xi,\varsigma>$表示,其中$\psi$表示系统状态集合,$\zeta$表示系统动作集合,$\xi$表示系统转移概率模型,$\varsigma$表示系统即时奖励。决策

者根据当前的系统状态 $\boldsymbol{X}\in\psi$，执行动作 $\boldsymbol{A}\in\zeta$，获得即时奖励 $R(\boldsymbol{X},\boldsymbol{A})\in\varsigma$，并根据状态转移概率 $P(\boldsymbol{X}'|\boldsymbol{X},\boldsymbol{A})\in\xi$ 转变到下一个状态 $\boldsymbol{X}'$。

2.1　系统状态

本文为生产单元新增了一个 D_n+1 状态，用于表示该生产单元处于预防性维修状态。因此，可将 M_n 的状态表示为 $S_n\in\{1,2,\cdots,D_n,D_n+1\}$，进而可将系统状态表示为 $\boldsymbol{X}=[S_1,S_2,\cdots,S_N,K_1,K_2,\cdots,K_{N-1}]$。

2.2　系统动作

生产单元 M_n 的维修动作为 $A_n\in\{\mathrm{DN},\mathrm{PM},\mathrm{CM}\}$。其中，DN、PM及CM分别代表“不维修”、“预防性维修”及“事后维修”。将系统的维修动作表示为 $\boldsymbol{A}=[A_1,A_2,\cdots,A_N]$。

2.3　系统奖励

将奖励函数定义为当系统状态为 $\boldsymbol{X}$ 且动作为 $\boldsymbol{A}$ 时，系统在一个单位时间内的收益，即：

$$R(\boldsymbol{X},\boldsymbol{A})=R_{\mathrm{r}}(\boldsymbol{X},\boldsymbol{A})-C_{\mathrm{P}}(\boldsymbol{X},\boldsymbol{A})-C_{\mathrm{M}}(\boldsymbol{X},\boldsymbol{A})-C_{\mathrm{B}}(\boldsymbol{X},\boldsymbol{A}) \tag{2}$$

式中：$R_{\mathrm{r}}(\boldsymbol{X},\boldsymbol{A})$ 为系统生产产品的收益，表示为：

$$R_{\mathrm{r}}(\boldsymbol{X},\boldsymbol{A})=v_{N,\mathrm{norm}}I(A_N=\mathrm{DN}\ \text{且}\ v_N=v_{N,\mathrm{norm}}) \tag{3}$$

$C_{\mathrm{P}}(\boldsymbol{X},\boldsymbol{A})$ 为各生产单元运行成本总和，表示为：

$$C_{\mathrm{P}}(\boldsymbol{X},\boldsymbol{A})=\sum_{n=1}^{N}C_{n,\mathrm{OP}}I(A_n=\mathrm{DN}) \tag{4}$$

$C_{\mathrm{M}}(\boldsymbol{X},\boldsymbol{A})$ 为各生产单元的维修成本总和，表示为：

$$C_{\mathrm{M}}(\boldsymbol{X},\boldsymbol{A})=\sum_{n=1}^{N}(C_{n,\mathrm{PM}}I(A_n=\mathrm{PM})+C_{n,\mathrm{CM}}I(A_n=\mathrm{CM})) \tag{5}$$

其中，函数 $I(\omega)$ 的含义如下：

$$I(\omega)=\begin{cases}1,\omega\ \text{为真}\\0,\omega\ \text{为假}\end{cases} \tag{6}$$

$C_{\mathrm{B}}(\boldsymbol{X},\boldsymbol{A})$ 为缓冲库存的总持有成本，表示为：

$$C_{\mathrm{B}}(X,A)=\sum_{n=1}^{N-1}C_{n,\mathrm{H}}K_n \tag{7}$$

2.4　转移概率模型

本文假设各生产单元的退化相互独立。因此，系统转移概率可简化为：

$$P(\boldsymbol{X}'|\boldsymbol{X},\boldsymbol{A})=\prod_{n=1}^{N}P(S_n'|S_n,A_n)\cdot\prod_{n=1}^{N-1}I(K_n'=K_n+v_n-v_{n+1}) \tag{8}$$

当 $A_n=\mathrm{DN}$ 时，根据 M_n 是否处于可运行状态，其状态转移概率为：

$$P(S_n'=s_n'|S_n=s_n)=(P_{n,\mathrm{norm}})_{s_n,s_n'} \tag{9}$$

或：

$$P(S_n'=s_n'|S_n=s_n)=(P_{n,\mathrm{idle}})_{s_n,s_n'} \tag{10}$$

当 $A_n=\mathrm{PM}$ 时，M_n 的状态转移概率为：

$$P(S_n'=1|S_n=s_n)=p_{n,\mathrm{PM}} \tag{11}$$

$$P(S_n'=D_n+1|S_n=s_n)=1-p_{n,\mathrm{PM}} \tag{12}$$

当 $A_n=\mathrm{CM}$ 时，M_n 的状态转移概率为：

$$P(S_n'=1|S_n=s_n)=p_{n,\mathrm{CM}} \tag{13}$$

$$P(S_n'=D_n|S_n=s_n)=1-p_{n,\mathrm{CM}} \tag{14}$$

虽然该MDP的状态转移模型已知，但由于其状态空间较大，因此，本文没有采用基于模型的算法对其进行求解。

3　维修优化方法

3.1　基于门限值的维修策略

为了和SR-DQL做比较，本文通过遗传算法（genetic algorithm，GA）确定各生产单元的预防性维修门限值。基于门限值的维修策略可表示为 $\boldsymbol{\mu}=[\mu_1,\mu_2,\cdots,\mu_N]$，其中，$\mu_n\in\{2,3,\cdots,D_n\}$ 是 M_n 的预防性维修门限值。当 $\mu_n=D_n$ 时，M_n 不进行预防性维修。由于系统的复杂性，稳态分析较为困难。因此，在用GA进行寻优时，本文采用仿真方法计算不同门限值组合下的系统平均收益。

3.2　Q学习

强化学习可以在状态转移概率模型未知的情况下，直接通过智能体与环境的交互来学习最优策略。Q学习（Q-learning，QL）是常用的表格式强化学习算法，其结果可以用策略函数 $\pi(\cdot)$ 表示。其中，$\pi(\boldsymbol{X})$ 表示系统状态为 $\boldsymbol{X}$ 时应采用的维修动作。QL的目的是找到一个最优策略 π^*，使其对应的总折扣收益最大，策略 π 对应的总折扣收益为：

$$R_\pi=\sum_{t=0}^{\infty}\gamma_t R\big(\boldsymbol{X}(t),\pi(\boldsymbol{X}(t))\big) \tag{15}$$

式中：$\boldsymbol{X}(t)$ 为 t 时刻的系统状态；γ 为收益折扣率。

和QL紧密相关的一个概念是状态动作值函数，在策略为 π 时的状态动作值函数 $Q_\pi(\boldsymbol{X},\boldsymbol{A})$ 表示当前状态动作对 $(\boldsymbol{X},\boldsymbol{A})$ 对应的总折扣收益，可以用递归方程表示为：

$$Q_\pi(\boldsymbol{X},\boldsymbol{A})=R(\boldsymbol{X},\boldsymbol{A})+\gamma E_{X'}(Q_\pi(\boldsymbol{X}',\pi(\boldsymbol{X}'))) \tag{16}$$

QL使用时间差分方法更新状态动作值函数$Q(\boldsymbol{X},\boldsymbol{A})$,其更新公式为:

$$Q(\boldsymbol{X},\boldsymbol{A})=(1-\eta)Q(\boldsymbol{X},\boldsymbol{A})+\eta(R+\max_{\boldsymbol{A}'}Q(\boldsymbol{X}',\boldsymbol{A}')) \tag{17}$$

式中:η为智能体学习率。

在算法收敛后,可以将$Q(\boldsymbol{X},\boldsymbol{A})$近似理解为最优策略对应的状态动作值函数。QL的一般框架如图2所示。首先,智能体观察到系统状态$\boldsymbol{X}$,然后根据ε-greedy策略选择该状态下的动作,使系统转变到新的状态$\boldsymbol{X}'$,并获得单步奖励R。

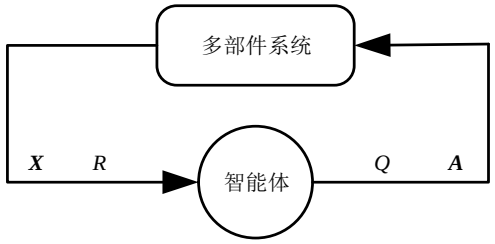

图2 QL框架

Fig. 2 QL farmwork

3.3 深度Q网络

当系统状态空间和动作空间的维度变大时,QL中用于表示其状态动作值函数的表格将呈指数增大并导致"维度灾难"。为克服这一困难,深度Q网络(deep Q-network, DQN)算法将QL中的表格用评论家网络Q代替。图3所示为DQN算法中的评论家网络。

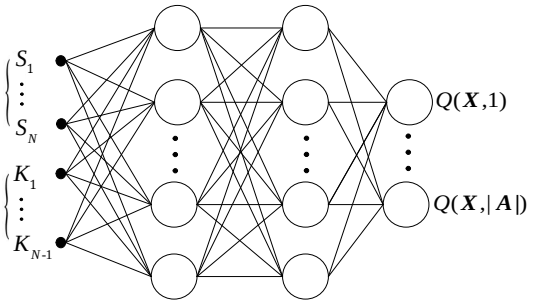

图3 评论家网络

Fig. 3 Critic network

在使用DQN算法时,评论家网络的输入为各生产单元的退化状态和各缓冲库存的库存量,评论家网络的输出为不同系统级维修动作$\boldsymbol{A}$对应的Q值。DQN算法和QL算法的流程类似,但神经网络近似带来的误差造成了学习过程的不稳定。为此,在DQN算法中引入了经验回放技术,将每一次智能体与环境交互的结果存放在经验缓冲区中,在训练网络时随机在经验缓冲区中选择小批次经验数据来更新网络,从而有效降低数据间的相关性。同时,在DQN算法中引入目标评论家网络Q_t来增加训练的稳定性,通过最小化损失函数来更新评论家网络:

$$L=\frac{1}{m}\sum_{j=1}^{m}(R_j+\max_{\boldsymbol{A}}Q_t(\boldsymbol{X}_j',\boldsymbol{A})-Q(\boldsymbol{X}_j,\boldsymbol{A}_j))^2 \tag{18}$$

式中:m为随机选择的经验数量;$\{\boldsymbol{X}_j,\boldsymbol{A}_j,\boldsymbol{R}_j,\boldsymbol{X}_j'\}$为选择到的第$j$个经验。

可以使用软更新或硬更新两种方法进行目标评论家网络Q_t的更新。本文使用硬更新方法对目标网络进行更新,即每隔一定训练步数将评论家网络Q的参数复制到目标评论家网络Q_t。

3.4 SR-DQL

除DQN算法外,多智能体强化学习算法也可以有效求解大规模MDP模型。常用的算法为分布式Q学习算法(distributed Q-learning, DQL),其一般框架如图4所示。

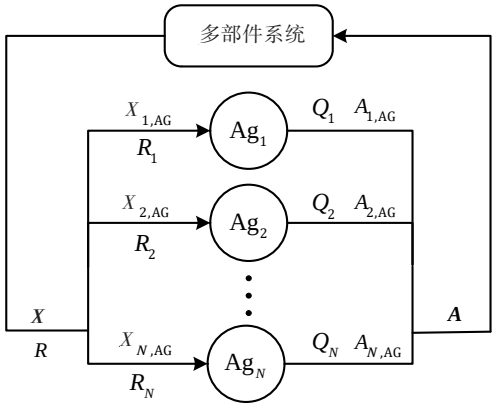

图4 DQL框架

Fig. 4 DQL farmwork

根据本文所研究的维修优化问题特性,每个智能体能观察到一个生产单元的退化状态以及在其上下游缓冲的库存水平,并根据观察到的信息为该单元选择维修动作。因此,整个系统有N个智能体,表示为:$\mathrm{Ag}_1,\mathrm{Ag}_2,\cdots,\mathrm{Ag}_N$。智能体$\mathrm{Ag}_n$观察到的系统状态向量为$\boldsymbol{X}_{n,\mathrm{AG}}=[S_n,K_{n-1},K_n]$,动作为$A_{n,\mathrm{AG}}=A_n$。DQL中每个智能体的状态空间和动作空间相对于整个系统来说显著减小。因此,可以采用表格的形式存储各智能体的状态动作值函数$Q_1,Q_2,\cdots,Q_N$。

在原始DQL算法中,各智能体的训练过程即各值函数的更新过程是相互独立的。在更新各智

能体的值函数时,所有智能体使用的都是生产系统的总收益,即所有智能体共享一个单步奖励。在实际应用中,这种共享收益的方式会使各智能体的训练过程相互干扰,造成收敛速度慢和收敛结果不理想。受啤酒游戏中奖励塑造[15]的启发,本文在DQL的基础上添加了奖励塑造,即SR-DQL。在更新函数Q_n时,对Ag_n的奖励变为:

$$R'_n = R_n + \beta(R - R_n) \tag{19}$$

式中:β为正则化系数;R为生产系统的总收益;R_n为生产单元M_n对应的成本,计算如下:

$$R_n = -C_{n,\mathrm{OP}} I(A_n = \mathrm{DN}) - C_{n,\mathrm{PM}} I(A_n = \mathrm{PM}) - C_{n,\mathrm{CM}} I(A_n = \mathrm{CM}) \tag{20}$$

式中:$C_{n,\mathrm{PM}}$、$C_{n,\mathrm{CM}}$、$C_{n,\mathrm{OP}}$分别为M_n在一个单位时间内的预防性维修成本、事后维修成本和运行成本。新的奖励函数同时兼顾了Ag_n决策对生产单元M_n成本的影响和系统总收益,从而降低了智能体之间训练过程的耦合性,提高了训练效率。与文献[15]不同的是,啤酒游戏只有在每个回合结束后才能得到系统总收益,因此,其更新时使用的收益都是每个回合的总收益,本文中每一步的单步收益均已知,因此,对奖励塑造以单步收益进行构造。各智能体状态动作值函数的更新与QL类似,具体形式如下:

$$Q_n(\boldsymbol{X}_n, A_n) = (1 - \eta) Q_n(\boldsymbol{X}_n, A_n) + \eta\left(R'_n + \max_{A'_n} Q_n(\boldsymbol{X}'_n, A'_n)\right) \tag{21}$$

4　数值试验

本文以5个生产单元和4个缓冲库存组成的串联生产系统为例,验证了所提维修优化算法的性能。此外,分析了所得的最优维修策略。

每个生产单元共有3种退化状态,其中状态1为完好状态,状态3为故障状态。各生产单元处于生产情况下的状态转移概率见表1。

表1　生产单元生产时的转移矩阵

Table 1　Transfer matrix of nominal production rate

n	$\left(P_{n,\mathrm{norm}}\right)_{1,1}$	$\left(P_{n,\mathrm{norm}}\right)_{1,2}$	$\left(P_{n,\mathrm{norm}}\right)_{1,3}$	$\left(P_{n,\mathrm{norm}}\right)_{2,2}$	$\left(P_{n,\mathrm{norm}}\right)_{2,3}$
1	0.70	0.20	0.10	0.80	0.20
2	0.80	0.10	0.10	0.75	0.25
3	0.75	0.15	0.10	0.75	0.25
4	0.70	0.15	0.15	0.80	0.20
5	0.80	0.10	0.10	0.80	0.20

如果生产单元处于饥饿或阻塞状态,其状态转移概率见表2。表3列出了生产单元的其他参数。各缓冲库存的容量为4。产品在各缓冲库存中的持有成本分别为0.25、0.20、0.20、0.15。系统的生产收益为18。在本算例中,成本单位为万元。

表2　饥饿或阻塞时的转移矩阵

Table 2　Transfer matrix when starved or blocked

n	$\left(P_{n,\mathrm{idle}}\right)_{1,1}$	$\left(P_{n,\mathrm{idle}}\right)_{1,2}$	$\left(P_{n,\mathrm{idle}}\right)_{1,3}$	$\left(P_{n,\mathrm{idle}}\right)_{2,2}$	$\left(P_{n,\mathrm{idle}}\right)_{2,3}$
1	0.90	0.050	0.050	0.90	0.10
2	0.85	0.100	0.050	0.90	0.10
3	0.95	0.030	0.020	0.85	0.15
4	0.95	0.025	0.025	0.90	0.10
5	0.90	0.050	0.050	0.95	0.05

表3　生产单元参数

Table 3　Parameters of production units

生产单元	v_{norm}	C_{OP}	C_{PM}	C_{CM}	p_{PM}	p_{CM}
M_1	3	0.10	3	11	0.90	0.60
M_2	2	0.05	3	11	0.85	0.55
M_3	2	0.10	4	12	0.95	0.50
M_4	2	0.08	3	11	0.90	0.55
M_5	1	0.10	4	12	0.85	0.60

在使用SR-DQL优化维修策略时,各智能体学习率η由0.011按指数下降至0.001,收益折扣率γ为0.99,奖励塑造的正则化系数β为2。除未使用奖励塑造外,DQL其余超参数均与SR-DQL相同,QL的超参数与DQL中某个智能体的相同。在DQN中评论家网络有9个输入节点,包括5个机器的退化状态以及4个缓冲库存的库存量,输出节点个数为2^5=32,对应5个生产单元选择维修或不维修的所有可能组合。除输入输出层外,网络中还有两个包含256个节点的全连接层,各节点使用Relu激活函数。神经网络的学习率为0.005,DQN探索率开始为0.9,在5 000步后线性下降为0.1,经验缓冲区的容量为10^5,每次随机选取64个经验数据进行训练。为确保算法的收敛性,RL算法都将学习10^7个单位时间的仿真数据。

每种优化算法均重复运行5次以评价其稳定性,结果见表4。表4中第2、3列展示了最优策略对应平均收益的均值和标准差。其中,SR-DQL得到的平均收益最高,DQL的次之。QL虽能处理该维修优化问题,但因系统状态多,效果不佳,所得收益未超过GA找到的基于门限值策略的收益。

DQN能解决状态空间爆炸问题,但由于神经网络近似带来的误差,其结果比GA的差。在不同算法中,DQL算法几次试验所得结果的标准差最小,本文所提出的SR-DQL的标准差次之,说明多智能体强化学习方法在多部件系统维修优化问题中相较于传统方法有更好的稳定性。从算法效率上看,表格式RL算法远好于GA和DQN,均能在9 min内处理完10^7个单位时间的仿真数据。而GA由于对每个解都需要分别进行仿真,最后耗时在2 h以上。DQN处理10^7个检修周期的仿真数据则需要花费24 h,说明更新评论家网络的效率比更新表格式Q函数的效率低得多。综上所述,本文所提出的方法在解的质量、算法稳定性及效率等方面均优于传统的基于门限值的方法。

表4 不同算法的计算结果

Table 4 Results of different algorithms

方法	均值/万元	标准差	时间/min
SR-DQL	4.153 7	0.021 5	8.37
DQL	4.135 1	0.011 7	8.11
GA	4.073 3	0.031 1	134.00
DQN	3.939 7	0.063 7	1 440.00
QL	3.902 1	0.028 6	5.36

图5显示了每种RL算法的5次重复试验中结果最好的一次的收敛过程。为减弱曲线的波动,图5中的点对应的数值是每隔1 000步计算得到的均值。从图5可以看出,SR-DQL和DQL算法收敛所需的训练数据最少,且奖励塑造也确实提高了SR-DQL的收敛速度。虽然QL训练花费时间较少,但其收敛所需的数据却是最多的;虽然DQN训练效率较低,但其所需训练数据远少于QL。

表5展示了由SR-DQL得到的生产单元M_4的维修策略。在SR-DQL中,Ag_4根据M_4的退化状态及其上下游缓冲库存B_3和B_4的库存量进行维修决策。由表5可知,当M_4的状态为2时,所采用的维修动作随B_3和B_4的库存量而变化。当B_4库存不足且B_3仍然有库存时,M_4维持运行状态,在其他情况下则进行预防性维修。GA找到的M_4的预防性维修门限值为2,也就是说只要M_4的退化状态为2,都要进行预防性维修。因此,由SR-DQL得到的维修策略比基于预防维修门限值的策略更灵活,能考虑更多的因素。

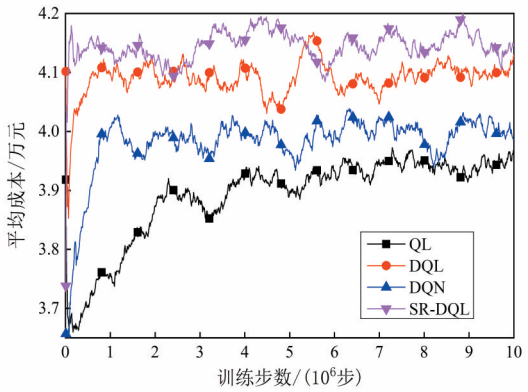

图5 不同算法收敛过程

Fig. 5 Convergence process of different algorithms

表5 M_4的维修策略

Table 5 Maintenance strategies for M_4

M_4	B_3	B_4	维修动作
2	2	0	DN
2	2	1	DN
2	2	2	PM
2	2	3	PM
2	2	4	PM
2	0	0	PM
2	0	1	PM
2	0	2	PM
2	0	3	PM
2	0	4	PM

5 结论

针对复杂多部件系统最优维修优化策略难以确定、单智能体强化学习方法无法处理较大规模状态空间和动作空间的问题,本文提出了一种多智能体强化学习算法,并引入奖励塑造进一步提升算法性能。

1)将多部件系统建立为MDP模型,分别确定单智能体强化学习和多智能体强化学习情况下合理的状态、动作及奖励函数。

2)提出了一种用于多部件生产系统维修优化的SR-DQL算法,引入奖励塑造使每个智能体可以将全局奖励与自身奖励相结合,提升了DQL算法的性能。

3)为验证算法的有效性,比较了不同算法解的质量、算法稳定性以及算法效率。分析了不同算法所得结果的维修策略,验证了本文所提出的

SR-DQL算法的有效性。

综上所述,多智能体强化学习算法能解决大规模系统状态空间和动作空间呈指数增长的问题,克服单智能体强化学习收敛困难的缺点。对于其他的多部件系统,多智能体强化学习算法也是一种很有潜力的方法。同时,智能体的参数设置和奖励塑造都需要维修优化的领域知识。因此,如何将维修优化的领域知识应用于强化学习是未来一个重要的研究方向。

[参考文献]

[1] LIU Y, HUANG H Z. Optimization of multi-state elements replacement policy for multi-state systems [C]//the Proceedings of Annual Reliability and Maintainability Symposium. San Jose: IEEE, 2010: 1-7. DOI: 10.1109/RAMS.2010.5448061.

[2] YUAN C, ZHOU Y F, MA L. A condition-based maintenance policy for a serial flow line with multiple intermediate buffers [C]//the Proceedings of 2019 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (QR2MSE). Zhangjiajie: IEEE, 2019: 279-284. DOI: 10.1109/QR2MSE46217.2019.9021164.

[3] RASMEKOMEN N, PARLIKAD A K. Condition-based maintenance of multi-component systems with degradation state-rate interactions [J]. Reliability Engineering & System Safety, 2016, 148: 1-10. DOI: 10.1016/j.ress.2015.11.010.

[4] 马维宁, 胡起伟, 杨志远. 考虑退化相关的装备多部件系统维修决策优化模型[J]. 系统工程与电子技术, 2022, 44(4): 1424-1432. DOI: 10.12305/j.issn.1001-506X.2022.04.42.
MA Weining, HU Qiwei, YANG Zhiyuan. Maintenance decision model of equipment multi-component systems with degradation dependence [J]. Systems Engineering and Electronics, 2022, 44(4): 1424-1432. DOI: 10.12305/j.issn.1001-506X.2022.04.42.

[5] KARAMATSOUKIS C C, KYRIAKIDIS E G. Optimal maintenance of two stochastically deteriorating machines with an intermediate buffer [J]. European Journal of Operational Research, 2010, 207(1): 297-308. DOI: 10.1016/j.ejor.2010.04.022.

[6] OLDE KEIZER M C A, TEUNTER R H, VELDMAN J. Clustering condition-based maintenance for systems with redundancy and economic dependencies [J]. European Journal of Operational Research, 2016, 251(2): 531-540. DOI: 10.1016/j.ejor.2015.11.008.

[7] UIT HET BROEK M A J, TEUNTER R H, DE JONGE B, et al. Joint condition-based maintenance and condition-based production optimization[J]. Reliability Engineering & System Safety, 2021, 214: 107743. DOI: 10.1016/j.ress.2021.107743.

[8] XU J, LIANG Z L, LI Y F, et al. Generalized condition-based maintenance optimization for multi-component systems considering stochastic dependency and imperfect maintenance[J]. Reliability Engineering & System Safety, 2021, 211: 107592. DOI: 10.1016/j.ress.2021.107592.

[9] ZHENG M M, YE H Q, WANG D, et al. Joint optimization of condition-based maintenance and spare parts orders for multi-unit systems with dual sourcing [J]. Reliability Engineering & System Safety, 2021, 210: 107512. DOI: 10.1016/j.ress.2021.107512.

[10] WANG J J, QIU Q G, WANG H H. Joint optimization of condition-based and age-based replacement policy and inventory policy for a two-unit series system [J]. Reliability Engineering & System Safety, 2021, 205: 107251. DOI: 10.1016/j.ress.2020.107251.

[11] NAJAFI S, ZHENG R, LEE C G. An optimal opportunistic maintenance policy for a two-unit series system with general repair using proportional hazards models [J]. Reliability Engineering & System Safety, 2021, 215: 107830. DOI: 10.1016/j.ress.2021.107830.

[12] KANG Y Y, JU F. Flexible preventative maintenance for serial production lines with multi-stage degrading machines and finite buffers [J]. IISE Transactions, 2019, 51(7): 777-791. DOI: 10.1080/24725854.2018.1562283.

[13] ZHOU Y F, GUO Y M, LIN T R, et al. Maintenance optimisation of a series production system with intermediate buffers using a multi-agent FMDP [J]. Reliability Engineering & System Safety, 2018, 180: 39-48. DOI: 10.1016/j.ress.2018.07.008.

[14] CHANG P C, LIN Y K, CHIANG Y M. System reliability estimation and sensitivity analysis for multi-state manufacturing network with joint buffers: a simulation approach [J]. Reliability Engineering & System Safety, 2019, 188: 103-109. DOI: 10.1016/j.ress.2019.03.024.

[15] OROOJLOOYJADID A, NAZARI M, SNYDER L V, et al. A deep Q-network for the beer game: deep reinforcement learning for inventory optimization [J]. Manufacturing & Service Operations Management, 2022, 24(1): 285-304. DOI: 10.1287/msom.2020.0939.

Maintenance optimization of multi-component system based on multi-agent reinforcement learning

ZHOU Yifan, GUO Kai, LI Bangcheng
(School of Mechanical Engineering, Southeast University, Nanjing 211189, China)

Abstract: [Purposes] This paper investigates the effectiveness of multi-agent reinforcement learning algorithms for maintenance optimization of multi-component production system. The feasibility of applying domain knowledge of maintenance optimization in reinforcement learning is also studied. [Methods] The maintenance decision making process of the production system was modeled as a Markov decision process (MDP), which was solved by a shaped reward distributed Q-learning (SR-DQL) algorithm. The domain knowledge of maintenance optimization was introduced into reinforcement learning by designing parameters of agents and reward shaping. [Findings] The proposed methods were validated using a production system with five production units and four inventory buffers. The proposed SR-DQL algorithm had a 6% ehancement of average revenuse comparing with the commonly used Q-learning. SR-DQL also outperformed distributed Q-learning and deep reinforcement learning algorithms. [Conclusions] The SR-DQL algorithm can effectively deal with the maintenance optimization problem of large-scale production systems, and reward shaping can improve the performance of the reinforcement learning algorithm.

Key words: multi-component production system; reward shaping; distributed Q-learning; multi-agent reinforcement learning; deep reinforcement learning

---

Manuscript received: 2022-10-11; **revised**: 2022-12-18; **accepted**: 2022-12-28
Foundation item: Project (72071044) supported by the National Natural Science Foundation of China
Corresponding author: ZHOU Yifan (1981—) (ORCID: 0000-0002-2898-0632), male, professor, research interest: reliability and maintenance optimization. E-mail: yifan.zhou@seu.edu.cn

(责任编辑:石月珍;校对:李脉;英文编辑:彭卓寅)