计算机科学
COMPUTER SCIENCE

基于后状态强化学习的最优订单接受决策

钱静，吴克宇，陈超，胡星辰

引用本文

钱静，吴克宇，陈超，胡星辰.基于后状态强化学习的最优订单接受决策[J]．计算机科学, 2022, 49(11A): 210800261-9.

QIAN Jing, WU Ke-yu, CHEN Chao, HU Xing-chen. Optimal Order Acceptance Decision Based on After-state Reinforcement Learning [J]. Computer Science, 2022, 49(11A): 210800261-9.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210800261

基于后状态强化学习的最优订单接受决策

钱 静 吴克宇 陈 超 胡星辰
国防科技大学系统工程学院 长沙 410073
(2516591697@qq.com)

摘 要 随着客户多样化需求不断提升,根据客户对订单的不同需求来组织生产的订单生产型(Make-To-Order,MTO)模式在企业生产活动中越来越重要。根据企业有限的生产能力和订单状态来确定是否接受到达的订单,对企业提高利润至关重要。在传统的订单接受问题基础上,提出了更完备的MTO企业订单接受问题的模型:在延期交货成本、拒绝成本、生产成本传统模型要素的基础上,进一步考虑了订单的库存成本、多种顾客优先级因素,并将最优订单接受决策问题建模为马尔可夫决策过程(Markov Decision Process,MDP)。此外,由于经典的MDP求解方法依赖于对高维状态价值函数的求解和估计,其计算复杂性较高,为了降低复杂性,证明了经典的MDP问题中基于状态价值函数的最优策略可以等价地用基于后状态的价值函数进行定义和构造,将多维控制问题转化为一维控制问题。同时,为了解决连续状态空间问题,结合神经网络对后状态价值函数进行参数化标表征,解决了状态空间较大的问题。最后,通过仿真验证了所提出的订单接受策略模型和算法的适用性和优越性。

关键词:订单接受;强化学习;马尔可夫决策过程;神经网络;后状态
中图法分类号 TP399

Optimal Order Acceptance Decision Based on After-state Reinforcement Learning

QIAN Jing,WU Ke-yu,CHEN Chao and HU Xing-chen
College of Systems Engineering,National University of Defense Technology,Changsha 410073,China

Abstract As the diversification of customer demand increases, the make-to-order (MTO) model, i. e., adapting production scheme according to customers' orders, has attracted increasingly more attention from industry. How to determine whether to accept incoming orders according to the limited production capacity and order status of the enterprise, which is crucial for the enterprise to improve profits. On the basis of the traditional order acceptance problems, this paper proposes a more complete model. Besides the traditional model elements (including delayed delivery cost, rejection cost, and production cost), we further consider the order inventory cost, customer priority and others. Moreover, we model the optimal order acceptance problem as a Markov decision process (MDP). In addition, because the classic MDP method relies on solving and estimating high-dimensional state value function, its computation complexity is high. Therefore, in order to reduce the complexity, this paper proves that the optimal strategy based on the state value function in the classical MDP problem can be defined and constructed by the value function based on the after-state equivalent, thus transforming the multi-dimensional control problem into a one-dimensional control problem. At the same time, in order to solve the continuous state space, this paper combines neural network to parameterize the after-state value function, and solves the problem of large state space. Finally, simulation experiments verify the applicability and superiority of the proposed order acceptance strategy model and algorithm.

Keywords Order acceptance, Reinforcement learning, Markov decision process, Neural network, After-state

1 引言

为了更容易地观察和接触终端客户[1],最大限度地满足顾客的个性化需求,越来越多的企业开始采用面向订单生产(MTO)模式。在通常情况下,影响MTO企业订单接受决策的因素有很多,这就需要企业综合考虑各种因素来制定相应的订单接受策略,但考虑的因素越多,模型及相应的求解算法就会越复杂。因此,如何既在建模时将变量因素考虑全面,又能在算法求解时降低复杂度,以保证企业利润最大化,是本文旨在解决的问题。

从已有的研究看,有关订单接受问题的决策方法已经取得了丰富成果。例如,较早开始研究订单接受问题的学者之一是Miller[1],他利用队列法来求解订单接受问题。Abedi等[2]提出了一种基于资源可用性的混合整数线性规划方法来确定最优订货数量。Zhang等[3]以企业利润最大化为目标,综合考虑生产计划和订单接受决策,建立了一个整数规划模型来帮助企业进行订单接受决策。Gao等[4]考虑到需求时序关联的MTO企业的订单选择与调度问题,建立了产能有限

基金项目:国家自然科学基金青年科学基金项目(62001495);湖南省自然科学基金青年科学基金项目(2020JJ5675)
This work was supported by the National Natural Science Foundation of China(62001495) and Natural Science Foundation of Hunan Province, China(2020JJ5675).
通信作者:吴克宇(keyuwu@nudt.edu.cn)

的 MTO 企业长期最优化的订单接受与调度的综合模型。Fan 等[5]以利润最大化为目标，运用基于 EMSR-a 和 EMSR-b 两种启发式方法来进行订单接受决策，并验证了其有效性。Wang 等[6]将订单接受和调度问题转化为混合整数规划模型，以实现订单总利润最大化。

基于以上来看，这些研究假定所有的订单是静态到达，即订单的信息是可以提前获取的，然而这种假设是难以满足实际需要的，因为在现实生活中，订单到达是随机动态的，不具有确定性。对此，为了更加符合实际，基于随机动态环境下的订单接受问题受到了学术界的关注和研究。Tarik 等[7]分析了在订单数量不确定的场景下，将需求灵活性引入到生产计划问题中，并提出了一种两阶段混合整数线性规划启发法的求解方法。Fan 等[8]针对随机动态变化的订单接受系统，提出了基于动态规划的方法来解决订单接受问题。Li 等[9]研究了以最大化企业总收益减去延期惩罚为目标的联合订单接受与调度问题，通过数值实验表明，基于动态规划的拉格朗日松弛算法在计算效率方面是最有效的。随着智能技术的发展，一些先进的智能启发式算法被引入企业订单接受问题的求解中，如利用遗传算法[10]、精确算法[11]、禁忌搜索算法[12]等来解决 MTO 企业订单接受问题。Wang 等[13]利用改进 NEH(Nawaz-Enscore-Ham)算法、离散和声搜索算法和变邻域搜索的混合算法对订单接受模型进行求解。Rahman[14]提出了一种混合遗传算法和粒子群优化算法来解决实时订单接受和调度决策问题。

但是基于动态环境下的订单接受问题中有很多不确定因素(如订单属性、订单到达时间、订单生产时间等)，当问题规模变大时，基于动态环境下的模型求解困难；同时，面对复杂情境时，难以给出精确的状态转移概率模型。而强化学习不需要建立系统的状态转移概率，通过与环境的交互，对值函数进行逼近，从积累的经验中学习最优的控制策略，从而解决规模较大的随机动态决策问题。对此，不少研究者将强化学习应用到订单接受决策问题中。Li 等[15]运用强化学习算法解决订单价格和交货期的设定、订单的接受或拒绝以及潜在的需求增长和产能限制的问题。Arredondo 等[16]提出了一种基于平均奖励强化学习的动态订单接受策略。Hao 等[17]从收益管理的思想出发，考虑订单价格、订单提前期等因素，运用强化学习方法对订单接受策略进行研究。Wang 等[18]考虑了订单生产成本、延迟惩罚成本、拒绝成本及顾客等级等因素，构建半马尔可夫决策过程模型，运用强化学习算法对订单接受进行决策。

现有关于强化学习订单接受决策的研究尽管已取得了丰富的成果，但是仍存在着问题模型考虑因素不全面和状态输入维度高而导致求解复杂度高等不足。例如，在模型考虑的因素方面，文献[16-18]中均没有考虑库存成本因素；文献[15]虽然考虑到了库存成本因素，但却忽视了顾客优先级；文献[15-18]均没有将库存成本和顾客优先级这两个因素同时考虑到模型中。在状态输入维度方面，文献[15-18]在定义状态空间时，均是多维度的，其中文献[15]将状态定义为新到达订单类型、新到达订单数量、n 种订单类型分别已接受的数量及剩余可用的生产能力，文献[16]将状态定义为订单的属性(订单数量、价格、最迟交货期)和当前生产能力，文献[17]将状态定义为订单类型、价格和提前期，文献[18]将状态定义为订单属性和生产订单所需要的时间。

对此，为了解决现有研究中的模型考虑因素不全面和求解过程复杂度高问题，本文将考虑 MDP 模型，进一步地引入库存成本和多种顾客优先级订单接受决策要素。特别地，提出了一种基于后状态(after-state)[19]和神经网络相结合的低复杂度的求解算法来解决 MTO 企业订单接受问题。具体地，本文的贡献如下：

(1)针对随机动态环境下考虑 MTO 企业订单接受问题，首先在考虑企业的生产成本、延迟惩罚成本、拒绝成本因素的基础上，又考虑了提前期之前完成的订单的库存成本和多种顾客优先级因素，构建了 MDP 订单接受模型。

(2)通过后状态方法对传统的 MDP 中的最优策略的求解进行了转换，证明了经典 MDP 问题中基于状态价值函数的最优策略可以等价地用基于后状态的价值函数进行定义和构造，将多维控制问题转化为一维控制问题，从而大大简化了求解过程。

(3)传统的 SRASA，SMART 等算法属于表格型强化学习方法[19]，该类方法仅能处理离散状态空间下的最优决策问题。为了解决连续状态空间下订单接受策略学习问题，本文利用神经网络对后状态价值函数进行参数化表征，并设计相应的训练算法，实现了对后状态价值函数的估计和订单接受策略的快速求解。

2 问题描述与假设

本文假设某产能有限的 MTO 企业通过单一生产线进行生产。假设市场上存在多种类型的顾客订单，订单相关的信息包括顾客优先级 μ、价格 pr、产品数量 q、提前期 lt 及最迟交货期 dt 等。单位时间生产的产品数量为 b，单位时间延期惩罚成本为 u，单位时间库存成本为 h，单位产品生产成本为 c。单位时间内顾客订单到达数量服从参数为 λ 的泊松分布。每个订单的价格、产品需求数量、订单提前期和最迟可接受交货期均服从均匀分布。对此，对于独立且同分布的订单信息 μ, pr, q, lt, dt，可用概率密度函数分别表示为 $f_M(x)$，$f_{PR}(x)$，$f_Q(x)$，$f_{LT}(x)$，$f_{DT}(x)$。

当有订单到达时，如果该订单不能在最迟交货期限完成，企业直接拒绝订单；否则企业需要根据自身生产能力等因素判断是否接受订单。如果拒绝订单，则产生拒绝成本 $\mu * F$，并且顾客优先级越高，拒绝成本越高。

如果接受订单，则获得该订单的利润，即 $I=pr*q$；同时消耗生产成本，即 $C=c*q$。MTO 企业对已经接受的订单按照先来先服务的原则进行生产，如果订单没有在顾客要求的提前期之前交货，即 $lt<t+q/b<dt$(其中 t 表示已接受的订单仍需要的生产时间)时，企业需要支付一定的延期交货成本 Y，即 $Y=\mu*u*((t+q/b)-lt)$，并且顾客优先级越高延期惩罚成本越高。顾客对提前期之前生产出来的产品不提前取货，即当 $t+q/b\leqslant lt$ 时，产品被暂存在 MTO 企业仓库中，产生库存成本 N，即 $N=h*(lt-(t+q/b))$，每个订单不进行拆分生产，订单生产完成后一次性发给顾客，并且企业一旦接受订单，顾客不能对订单进行更改或取消。

本文要解决的问题是，当顾客订单随机达到时，MTO 企业在考虑当前的生产能力以及延期交货成本 Y、拒绝成本 $\mu*F$、生产成本 C、库存成本 N 及多种顾客优先级因素 μ 的

基础上，决策是否接受当前到达的订单，以保证企业长期平均利润最大化。具体过程如图 1 所示。

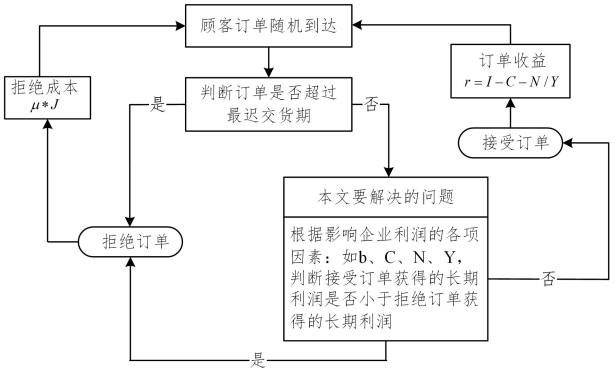


图 1　问题描述流程图

Fig. 1　Problem description flowchart

3　基于 MDP 理论的订单接受决策建模

当有顾客订单随机动态到达时，企业需要立刻做出是否接受订单的决策，可见 MTO 企业订单接受决策问题是一类随机序贯决策问题。MTO 企业决策者在决策接受或者拒绝订单后，系统状态发生改变，即决策状态转移至下一时刻新订单到达，从此决策阶段以后的发展过程仅与该决策阶段有关，而与该决策阶段以前所经历的状态无关，即其具有无后效性。因此，可依据 MDP 理论，将该问题抽象成 MDP 模型。MDP 模型定义为四元组(S,A,f,R)，其中，S，A，f 和 R 分别表示系统状态空间、系统动作空间、状态转移函数、奖励函数：

(1)系统状态：假设系统中有无限种订单类型，系统状态可由向量 S 表示：$S=(\mu,p,q,lt,dt,t)$。t 表示已接受的订单仍需要的生产完成时间，基于有限产能的 MTO 企业，t 有最大上限值。

(2)系统动作集合：在决策时刻 m，当有顾客订单到达时，MTO 企业需要做出接受或拒绝订单的决策。模型中的动作集合可由向量 $A=(a_0,a_1)$ 表示，其中 a_0 表示拒绝订单，a_1 表示接受订单。

(3)状态转移模型：在给定初始状态 s 和已经采取动作 a 的情况下，下一个状态的概率密度函数用 $f(\cdot\mid(s,a))$ 表示。由于订单信息 μ,pr,q,lt,dt 均是独立且同分布的，因此订单信息$\mu_{m+1},pr_{m+1},q_{m+1},lt_{m+1},dt_{m+1}$关于$(s_m,a_m)$是独立的。但是$t_{m+1}$受$(s_m,a_m)$的影响，这是因为不同的$(q_m,t_m,a_m)$会导致不同的订单生产时间，$t_{m+1}$同样也受订单到达时间间隔的影响，即$t_{m+1}$可以表示为：

$$t_{m+1}=[(t_m+a_m*\frac{q_m}{b}-AT)]^+ \tag{1}$$

其中，$[x]^+\triangleq\max(x,0)$；AT 表示两个订单之间到达的时间间隔，即单位时间内订单到达的数量服从参数为 λ 的泊松分布。

由于订单信息$\mu_{m+1},pr_{m+1},q_{m+1},lt_{m+1},dt_{m+1}$关于$(s_m,a_m)$是独立的，所以在给定当前状态 s 和动作 a 时，下一决策时刻状态 s'的条件概率密度可以表示为：

$$f(s'\mid s,a)=f_M(\mu')*f_{PR}(pr')*f_Q(q')*f_{LT}(lt')*f_{DT}(dt')*f_T(t'\mid s,a)$$

其中，$f_T(t'\mid s,a)$表示在当前状态 s 下，采取动作 a 后，下一时刻生产已接受的订单仍需要的生产时间，其具体形式可以由式(1)和相关随机变量定义。

(4)奖励函数：在决策时刻 m，MTO 企业在做出是否接受订单决策后，获得的立即回报函数为：

$$r(s_m,a_m)\triangleq\begin{cases}I-C-Y \text{ or } I-C-N, & a_m=1\\ -\mu*F, & a_m=0\end{cases}$$

(5)最优策略：在 MTO 企业订单接受问题中，目标是寻找一个最优的订单接受策略π^*，从而使企业长期利润最大化。每个策略 π 是从系统状态 s 到动作 a 的函数，其决定了企业如何根据当前状态信息选择是否接受订单。对于任意策略 π，定义其价值函数为其长期平均利润，即：

$$V^\pi=E[\sum_{m=0}^{+\infty}\gamma^m r(s_m,\pi(s_m))] \tag{2}$$

其中，$0<\gamma\leqslant 1$ 表示未来奖励折扣(其保证式(2)中定义的求和项是有意义)。而我们关心的是所有策略中的最优策略 π^*，其定义为：

$$\pi^*=\arg\sup_{\pi\in\Pi}\{V^\pi\}$$

其中，Π 表示所有策略集合。

在 MDP 理论中，最优策略π^*可以用状态价值函数V^*进行构造：

$$\pi^*(s)=\arg\max_a\{r(s,a)+\gamma E[V^*(s')\mid s,a]\} \tag{3}$$

其中，期望 $E[\cdot]$是定义在给定当前状态 s 和动作 a 的下一个随机的状态 s'。同时，V^* 是 Bellman 方程的一个解，V^* 可以用式(4)来计算得到，即：

$$V(s)=\max_a\{r(s,a)+\gamma E[V(s')\mid s,a]\} \tag{4}$$

可见，经典的 MDP 理论提供了一种基于状态价值函数 V^* 的最优策略求解方法。但是，由于本文中需要解决的问题状态是多维的，即 $S=(\mu,p,q,lt,dt,t)$，同时式(3)需要随机下一个状态的条件期望，这就导致了计算复杂度高的问题。对此，本文提出了基于后状态价值函数的最优策略构造方法。

4　基于后状态的 MDP 模型转化

后状态是两个连续状态之间的中间变量，可以用来简化某些 MDP 的最优控制。后状态的概念是强化学习中经常用于棋类游戏的学习任务的一种技巧[19]。

例如，智能体(Agent)在运用强化学习算法下围棋的时候，Agent 可以确定地控制自己的走法，但对手的行动是随机的。如图 2 所示，黑棋代表我方(Agent)，白棋代表对方，在棋局状态s_1下，轮到我方行动，Agent 根据当前棋盘上特定的棋子位置来决定自己的行动，接着在对方采取行动之后，棋局状态会由s_1转移至状态s_2，这与经典的 MDP 模型中的状态是相同的。而 Agent 每一步棋的后状态被定义为这一步行动之后但在对手移动之前所在棋盘的状态，如图 2 所示，在棋局状态s_1下，轮到我方行动，Agent 在状态s_1采取动作a_1后，对方还未采取行动，棋局状态会变为图 2 中s_1与s_2中间的状态，该状态即为后状态。Agent 如果能够了解所有不同后状态的获胜机会，那么就可以使用这些已知概率来实现最优行为，即简单地选择获胜机会最大的后状态进行相应行动。

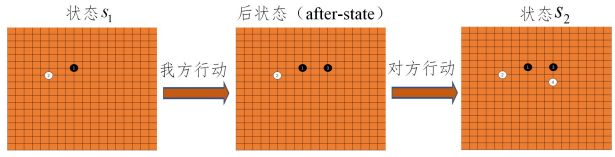


图 2　后状态解释图

Fig. 2　After- state interpretation diagram

本文采用类似的后状态方法对订单接受问题进行变换。在我们考虑的订单接受问题中，后状态变量p_m定义为在m决策时刻选择动作a_m后生产已接受订单仍需要的生产时间。因此，在给定当前状态s_m和动作a_m后，后状态可以表示为：

$$p_m=\sigma(s_m,a_m)=\frac{q_m}{b}*a_m+t_m \tag{5}$$

不难看出，给定p_m，下一个决策时刻$m+1$订单仍需要的生产时间t_{m+1}可以表示为：

$$t_{m+1}=\left[\left(t_m+a_m*\frac{q_m}{b}\right)-AT\right]^+=[p_m-AT]^+ \tag{6}$$

其中，AT为两个订单之间到达的时间间隔。所以，给定当前后状态p，下一个决策时刻状态$S'=(\mu',p',q',lt',dt',t')$的条件概率密度可以表示为：

$$f(s'|p)\triangleq f_M(\mu')*f_{PR}(pr')*f_Q(q')*f_{LT}(lt')*f_{DT}(dt')*f_T(t'|p) \tag{7}$$

其中，条件概率密度函数$f_T(\cdot|p)$由式(6)和相关的随机变量所定义。

因此，在式(3)和式(4)中的条件期望$E[V(s')|s,a]$，可以改写成用式(7)表示的条件期望$E[V(s')|\sigma(s,a)]$，其中$\sigma(s,a)$表示后状态，见式(5)，所以π^*可以被重新定义如下。首先定义后状态值函数为：

$$J^*(p)=\gamma E[V^*(s')|p] \tag{8}$$

将式(8)加入式(3)，则最优策略π^*可以使用后状态价值函数J^*构造如下：

$$\pi^*(s)=\arg\max_a\{r(s,a)+J^*(\sigma(s,a))\} \tag{9}$$

进一步地，将式(8)代入式(4)：

$$V^*(s)=\max_a\{r(s,a)+J^*(\sigma(s,a))\}$$

所以$\gamma E[V^*(s')|p]=\gamma E[\max_{a'}\{r(s',a')+J^*(\sigma(s',a'))\}|p]$。事实上由式(8)可知，$\gamma E[V^*(s')|p]$即为$J^*(p)$，所以可以得出：

$$J^*(p)=\gamma E[\max_{a'}\{r(s',a')+J^*(\sigma(s',a'))\}|p]$$

最后，根据文献[19]，可通过值迭代算法对J^*进行求解，即J_0为任意初始化函数，则有：

$$J_{k+1}(p)=\gamma E[\max_{a'}\{r(s',a')+J_k(\sigma(s',a'))\}|p] \tag{10}$$

当$k\to\infty$时，J_k收敛到J^*。

由式(3)知，计算最优策略时需要使用期望$E[V^*(s')|s,a]$。对于文本的状态空间，即$S=(\mu,p,q,lt,dt,t)$，如果考虑使用期望$E[V^*(s')|s,a]$来计算最优策略，则需要付出很大的计算代价。而式(9)中的最优策略不需要考虑期望，直接用J^*来替代$E[V^*(s')|s,a]$，不仅解决了上述问题，还将高维状态空间降低为一维状态空间，大大降低了求解复杂度。

5 基于神经网络的最优订单接受决策

在此前已经证明了π^*可以用J^*来构造，而J^*又可以通过式(10)值迭代的方式来求解。但是式(10)在实施过程却存在两种困难：首先，如果$f_M(\cdot)$，$f_{PR}(\cdot)$，$f_Q(\cdot)$，$f_{LT}(\cdot)$，$f_{DT}(\cdot)$和$f_T(\cdot|\sigma(s,a))$是不可获取的，将无法计算$E[\cdot|p]$；其次，由于后状态是连续而非离散的，每次迭代式(10)都必须在无穷多个p值上计算，因此不能用传统的表格型算法对J^*进行计算。对此，我们采取一种通过学习参数向量来实现J^*的近似，对后状态进行泛化，学习的过程是利用数据样本。换句话来说，算法的设计包括：

(1)参数化：函数$\hat{J}(p|\theta)$可以由参数向量θ来近似表示；

(2)参数学习：给定初始参数向量θ，θ通过对数据样本的学习而变为θ^*，进而可以使用$\hat{J}(p|\theta^*)$来近似J^*，即最优的策略可以表示为：

$$\hat{\pi}(s|\theta^*)=\arg\max_a\{r(s,a)+\hat{J}(\sigma(s,a)|\theta^*)\} \tag{11}$$

对比式(11)和式(9)可知，如果$\hat{J}(p|\theta^*)$近似于$J^*(p)$，则$\hat{\pi}(s|\theta^*)$接近最优策略π^*。

5.1 神经网络近似

万能近似定理(Universal Approximation Theorem)[20]表明三层人工神经网络(Artificial Neural Network，ANN)能够将一个连续函数近似到任意精度，因此对于本文中要求解的$J^*(p)$，ANN[21]是一个很好的选择。所以$\hat{J}(p|\theta)$可以用神经网络表示为：

$$\hat{J}(p|\theta)=\left(\sum_{i=1}^{N}u_i\Phi_H(w_ip+\alpha_i)\right)+\beta \tag{12}$$

其中，参数向量可以表示为：

$$\theta=[w_1,\cdots,w_N,\alpha_1,\cdots,\alpha_N,u_1,\cdots,u_N,\beta]$$

$$\Phi_H(x)=1/(1+e^{-x})$$

式(12)中的函数$\hat{J}(p|\theta)$如图3所示，它实际上是一个三层的单输入单输出神经网络。具体来说，只有一个单节点的输入层，其输出表示后状态p的值，还有一个包含N个节点的隐藏层，第i个节点的输入是加权后态值w_i*p和隐含层偏置α_i的和。每个隐藏层节点的输入与输出关系由$\Phi_H(\cdot)$函数表示，其中$\Phi_H(\cdot)$被称为激活函数[21]。最后，输出层有一个节点，其输出代表最终逼近的函数值$\hat{J}(p|\theta)$，它的输入是隐藏层加权输出和输出层偏置β的总和。

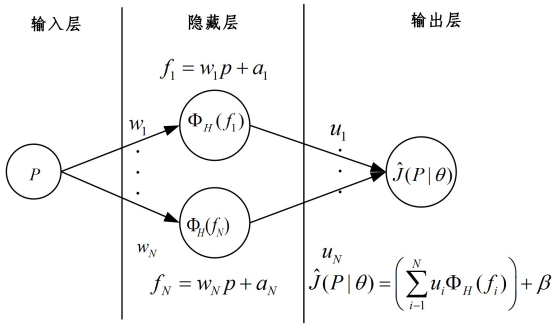


图3 三层神经网络架构

Fig. 3 Three-layer neural network architecture

5.2 值迭代训练神经网络

为了达到最优控制，通过值迭代的方法来训练神经网络中的参数是非常有用的[22]，见图4。具体的训练过程解释如下。

(1)训练数据的获取。本文需要一批训练样本$\Gamma=(p_m,s_m=[\mu_m,pr_m,q_m,lt_m,dt_m,t_m])_{m=0}^{M-1}$，对于每个$m$，采样获得$p_m,\mu_m,pr_m,q_m,lt_m,dt_m$，其中$p_m$服从均匀分布，$\mu_m\sim f_M(\cdot)$，$pr_m\sim f_{PR}(\cdot)$，$q_m\sim f_Q(\cdot)$，$lt_m\sim f_{LT}(\cdot)$，$dt_m\sim f_{DT}(\cdot)$。进一步地，根据$p_m$生成$t_m$，即$t_m=(p_m-AT)^+$，其中$AT$为两个订单之间到达的时间间隔。

(2)迭代拟合(见算法 1)。如式(10)所示,在第 k 次迭代时,设当前 ANN 参数向量为θ_k,用式(12)定义的函数$\hat{J}(p|\theta_k)$,即给定当前值函数$J_k(p)=\hat{J}(p|\theta_k)$,更新后的值函数为:

$$J_{k+1}(p)=\gamma E[\max_{a'}\{r(s',a')+J_k(\sigma(s',a')|\theta_k)\}|p] \tag{13}$$

算法 1　AFVINN:近似$J^*(p)$

输入:p_m,s_m

输出:学习到的$\hat{J}(p|\theta_K)$

1. 初始化参数 θ_0
2. for k from 0 to K−1 do
3.　for m from 0 to M−1 do
4. 从 Γ 数据集中拿取第 m 个数(p_m,s_m)
5. 根据式(14)从θ_k和s_m中计算o_m
6. 收集$Y_k(m)=(p_m,o_m)$
7.　end for
8. 利用Y_k和θ_k,更新参数θ_{k+1}(见算法 2)
9. end for
10. 根据式(12),用更新的参数 $\theta=\theta_K$得到$\hat{J}(p|\theta_K)$

我们希望利用更新的参数来获得一个新的函数$\hat{J}(p|\theta_{k+1})$来逼近$J_{k+1}(p)$。所以,从 Γ 和θ_k中,我们构造一批训练数据:

$$Y_k=\{(p_m,o_m)\}_m^{M-1}$$

其中,o_m定义如下:

$$o_m=\gamma\max_a\{r(s_m,a)+\hat{J}(\sigma(s_m,a)|\theta_k)\} \tag{14}$$

基于训练数据Y_k,参数更新可以表示为:

$$\theta_{k+1}=\arg\min_\theta\{L(\theta|Y_k)\} \tag{15}$$

其中,$L(\theta|Y_k)$为训练误差。

$$L(\theta|Y_k)=\frac{1}{2M}\sum_{m=0}^{M-1}(\hat{J}(p_m|\theta)-o_m)^2 \tag{16}$$

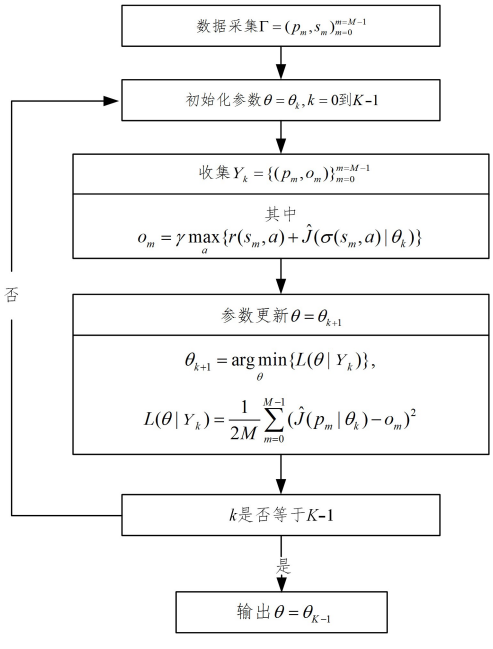


图 4　AFVINN 算法流程图

Fig. 4　Flow chart of AFVINN algorithm

(3)训练参数(见算法 2)。应用梯度下降求解 AFVINN 参数θ_{k+1},使得$\hat{J}(p|\theta_{k+1})$在Y_k上误差最小,见式(15)。梯度下降是通过迭代搜索参数空间:梯度迭代中初始参数$\theta^{(0)}$设为θ_k,则迭代过程中参数更新为:

$$\theta^{(z+1)}=\theta^{(z)}-\alpha*\nabla L(\theta^{(z)}) \tag{17}$$

其中,α 为更新步长参数,$\nabla L(\theta^{(z)})$是定义在式(16)中的 L 在$\theta^{(z)}$的梯度。因此,给定足够的迭代次数 Z,我们用$\theta_{k+1}=\theta^{(Z)}$作为式(15)的近似解。最后$\nabla L(\theta)$表示为:

$$\nabla L(\theta)=\frac{\partial L}{\partial\theta}=\frac{1}{M}\sum_{m=0}^{M-1}(\hat{J}(p_m|\theta)-o_m) \tag{18}$$

算法 2　Inner loop of AFVINN:参数 θ 更新

输入:Y_k,初始化θ_k

输出:训练后的参数θ_{k+1}

1. $\theta^{(0)}=\theta_k$
2.　for z from 0 to Z−1
3. 根据式(18)用Y_k计算$\nabla L(\theta^{(z)})$
4. 根据式(17)用$\theta^{(z)}$和 $L(\theta^{(z)})$获得新的参数$\theta^{(z+1)}$
5.　end for
6. $\theta_{k+1}=\theta^{(Z)}$

综上,收集训练数据 Γ 后,执行算法 1,通过值迭代神经网络学习参数θ_K,让$\theta^*=\theta_K$,然后通过式(9)进一步构造策略$\hat{\pi}(\cdot|\theta_K)$。在 M 和 K 比较大的情况下,$\hat{\pi}(s|\theta_K)$可以接近$\pi^*(s)$。对此,根据学习到的策略,我们将其应用到订单接受的最优决策中,具体过程见图 5。最优的订单接受策略π^*遵循以下结构:

$$\pi^*(s)=\begin{cases}1, & \text{if } dt\geqslant t+q/b \text{ and } r_1+J^*(p_1)\geqslant\\ & r_0+J^*(p_0)\\ 0, & \text{else}\end{cases} \tag{19}$$

其中,$\pi^*(s)=\pi^*([\mu,pr,q,lt,dt,t])$,$r_1=I-C-Y/N$,$r_0=-\mu*F$,$J^*(p_1)=J^*(t+q/b)$,$J^*(p_0)=J^*(t)$。

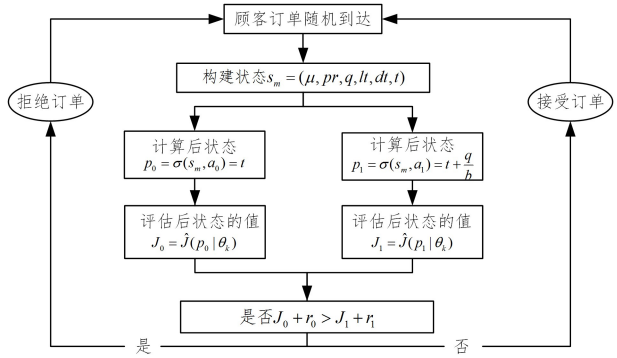


图 5　最优订单接受决策流程图

Fig. 5　Flow chart of optimal order acceptance decision

6　数值仿真实验

仿真中所需要的相关订单信息根据文献[15,17-18,23-25]的数据生成方法,按照以下规则生成:订单价格 pr 服从均匀分布$U(e_1,l_1)$,订单数量 q 服从均匀分布$U(e_2,l_2)$,单位时间订单的到达数量服从参数为 λ 的泊松分布;订单提前期和最迟可接受交货期分别服从均匀分布$U(e_3,l_3)$,$U(e_4,l_4)$。

这里我们选取 $pr\sim U[30,50]$,$q\sim U[300,500]$,$lt\sim U[15,20]$,$dt\sim U[20,60]$,$\lambda=0.2$。企业的单位生产能力、

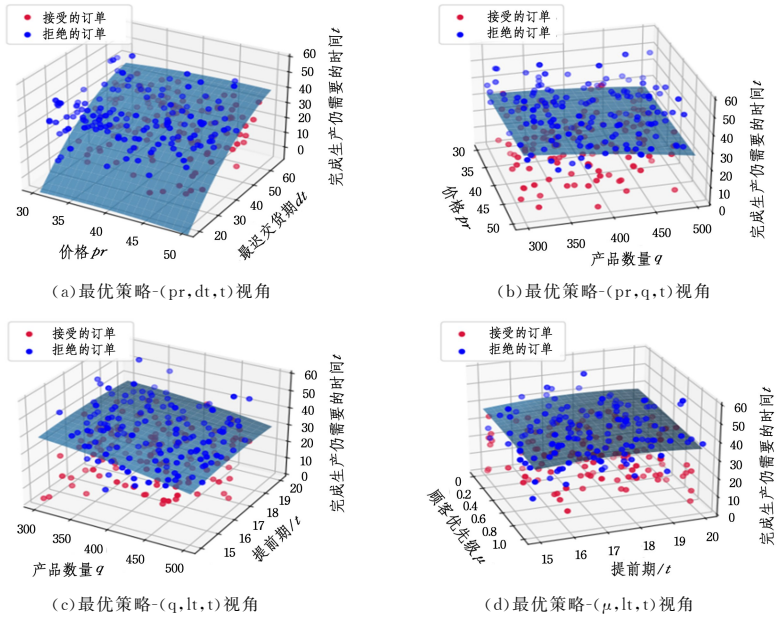
单位生产成本、拒绝成本依据文献[18]分别取得 $b=20$，$c=15$，$F=200$。单位时间延期惩罚成本 $u=200$，顾客等级服从均匀分布 $\mu\sim U(0,1]$，单位时间库存成本 $h=50$。

通过仿真实验，对本文所提出基于 AFVINN 算法的企业订单接受策略的有效性进行分析。

本仿真实验由 3 部分组成。在第一部分中，根据学习到的参数θ_K，让$\theta^*=\theta_K$，通过式(9)计算得出最优策略$\pi^*(s)$。在第二部分中，首先将 AFVINN 算法与传统的 Q-learning 算法进行学习效率的对比，对比策略是通过样本利用效率进行评估的；其次将提出的 AFVINN 算法与 FCFS 方法、greedy 方法及 Q-learning 算法对企业长期平均利润和订单接受率进行对比和分析。其中，FCFS 方法是指当订单到达时，若企业有能力在最迟交货期内完成生产，则直接接受该订单；greedy 方法是指当订单到达时，若企业有能力在最迟交货期内完成生产，并且该订单的顾客优先级较大，则接受该订单。订单的接受率是指接受订单的数量除以总到达订单的数量。在第三部分中，首先将本文的算法与文献[18]中提出的不考虑库存成本因素的模型(SMART 算法)进行对比，来观察两种模型对 MTO 企业平均利润的影响；其次通过改变与顾客优先级有关的延期惩罚成本和拒绝成本因素，来分析 AFVINN 算法对 MTO 企业平均利润及订单接受率的影响。

6.1 最优订单接受策略

图 6 为不同变量下的最优订单接受策略图，我们用红色圆点表示接受的订单，蓝色圆点表示拒绝的订单，蓝色的面为订单接受决策的超平面。



(a)最优策略-(pr,dt,t)视角　(b)最优策略-(pr,q,t)视角

(c)最优策略-(q,lt,t)视角　(d)最优策略-(μ,lt,t)视角

图 6 最优订单接受策略图(电子版为彩图)

Fig. 6 Diagram of optimal order acceptance strategy

由图 6 可知：1)当顾客优先级和订单的价格越高、产品数量越少、提前期和最迟交货期较长及已接受订单仍需要的生产时间越短时，企业越容易接受订单；2)当顾客优先级和订单的价格越低、产品数量越多、提前期和最迟交货期较短及已接受订单仍需要的生产时间越长时，企业越容易拒绝订单。这与式(19)构造的最优订单接受策略相符合，解释如下。

当最迟交货期较短、产品数量越多和已接受订单仍需要的生产时间越长，即 $dt<t+q/b$ 时，企业会因为自身生产能力不足而直接拒绝订单。如果 $dt\geqslant t+q/b$，企业会从长远的角度出发，综合考虑影响利润的因素，来判断是否接受订单。由于本文中定义的后状态 $J(p)$是非增的，即后状态价值函数随着 p 的增加而减少，所以企业在对订单进行接受决策时，不仅要考虑即时收益，也要考虑到长期收益。当最迟交货期较长、产品数量较少和已接受订单仍需要的生产时间较短，即 dt 较大。$t+q/b$ 较小时，如果顾客优先级和订单价格较高，提前期较长，则接受该订单所产生的利润 $pr*q$ 会较大，延迟交货期$[(t+q/b)-lt]^+$会较小或为零，相比拒绝订单带来的负收益$r_0=-\mu*F$，接受该订单所获得的利润r_1会很大，因此$r_1+J^*(p_1)\geqslant r_0+J^*(p_0)$，企业会选择接受订单。同理，当产品数量较多和已接受订单仍需要的生产时间较长时，如果顾客优先级和订单价格较低，提前期较短，接受该订单所产生的利润 $pr*q$ 会较小，延迟交货期$[t+q/b)-lt]^+$会较大，则接受该订单所获得的利润r_1会比较小甚至为负，$r_1+J^*(p_1)\leqslant r_0+J^*(p_0)$，则企业会选择拒绝订单。

可见针对动态环境下的订单接受问题，在使用 MDP 对订单接受决策问题进行建模时，需要综合考虑影响利润的各种因素，但状态空间会随着考虑因素的增加而增大，这在提高决策结果准确性的同时，也导致了由于状态空间维度高而计算复杂度高的问题。因此，本文提出了一维后状态来解决高维状态空间的问题，大大降低了求解的复杂性。

6.2 算法比较

已有文献[15,17-18]关于强化学习 MTO 企业订单接受策略中，均是使用传统多维状态空间进行建模和求解。对此，本文将提出 AFVINN 算法与传统的 Q-learning 算法进行学习效率对比，对比策略是每次迭代消耗 100 个数据样本，根据消耗的数据样本数量来评估学习效率。

由图 7 可知：1)经过足够的数据样本，AFVINN 算法和传统的 Q-learning 算法都可以收敛，但 Q-learning 算法的

平均利润略低于 AFVINN 算法，这是因为本文的状态是连续的，状态空间较大，在运用 Q-learning 算法时，需要将其进行离散化处理，因此平均利润略低于 AFVINN 算法；2）AFVINN 算法的学习效率远远高于传统 Q-learning 算法的学习效率，前者约是后者的 200 倍。由此可知，本文提出的 AFVINN 算法能够将高维控制问题转化为一维控制问题，即传统的六维状态输入 $s=(\mu,pr,q,lt,dt,t)$ 转为一维后状态 p，大大简化了求解过程，在数据样本利用效率方面表现出较强的优势；同时，利用神经网络对状态空间进行泛化，可以处理较大的规模订单接受决策问题。

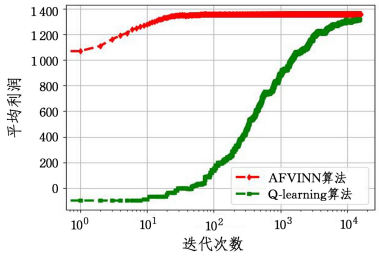


图 7　样本学习率

Fig. 7　Sample learning rate

表 1　基本情境

Table 1　Bbasic situation

算法	平均利用	订单接受率
AFVINN	1 361.661 5	0.205 5
Q-learning	1 309.724 8	0.209 5
FCFS	1 203.513 9	0.211 2
greedy	908.429 2	0.203 4

由表 1 表明：1）AFVINN 算法在最大化 MTO 企业长期平均利润方面均优于其他 3 种方法；2）在订单接受率较低时，AFVINN 算法仍可以保持较高的平均利润。由此可知，AFVINN 算法在订单接受策略上，能够以更高的概率接受具有更高利润的订单，以达到最大化企业长期平均利润的目的。

生产能力对于 MTO 企业的盈利是非常关键的。通过改变 MTO 企业单位生产能力[13,16]，其他的参数与基本情境相同，来观察 AFVINN 算法与 FCFS 方法、greedy 方法以及 Q-learning 算法在 MTO 企业订单接受策略方面的变化。

由图 8 可知：1）4 种方法的长期平均利润均随企业单位生产能力的提升而增加，长期平均利润与企业单位生产能力成正比例关系；2）不论企业的单位生产能力降低还是提高，AFVINN 算法都可以始终保持较高的利润水平。由此可知，AFVINN 算法能够合理地利用企业有限的资源，从而为企业创造更高的利润，在资源有限的情况下有更好的适应性。

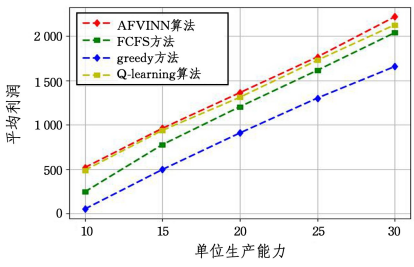


图 8　不同单位生产能力的平均利润

Fig. 8　Average profit of different unit production capacity

订单到达率同样是 MTO 企业订单接受决策中的一个重要因素。通过改变订单的到达率[14,18]，其他参数与基本情境相同，来观察 AFVINN 算法与 FCFS 方法、greedy 方法以及 Q-learning 算法在 MTO 企业订单接受策略方面的变化。

由图 9 可知：1）当单位时间内顾客订单到达的数量逐渐增加时，4 种方法的长期平均利润不仅没有增加，反而呈现下降趋势；2）面对不同单位时间内顾客订单到达的数量，AFVINN 算法始终保持较高的利润水平。当 λ 提高时，单位时间内订单到达的数量增加，即两个订单之间到达的时间间隔减少时，这会使得 MTO 企业安排已接受的订单时间减少，所以，在最迟交货期限内，已接受订单的完成概率将会减少，企业不得不拒绝掉更多的订单，承担更多的拒绝成本。因此，当单位时间内顾客订单到达数量增加时，企业应考虑提升自身的生产能力，比如增加生产设备、增加劳动力等因素，降低因拒绝成本增加而导致平均利润降低的局面。

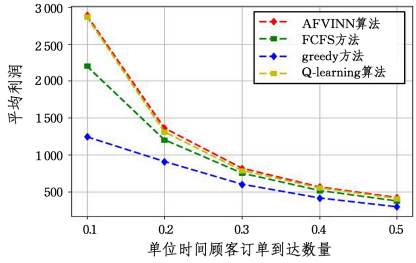


图 9　不同订单到达率的平均利润

Fig. 9　Average profit of different order arrival rates

6.3　模型比较

已有文献[15-16]在运用强化学习算法对订单接受问题进行建模求解时，均没有考虑库存成本这一因素。本节将考虑库存成本的模型（AFVINN 算法）和不考虑库存本的模型（SMART 算法）在订单接受策略中加以对比。SMART 算法在订单接受问题中先不考虑库存成本这一因素，但在计算平均利润时将库存成本考虑进去与本文的 AFVINN 算法进行对比。

由图 10 可知：1）在其他因素不变时，考虑库存成本这一因素下的企业平均利润始终高于不考虑库存成本这一因素下的企业平均利润；2）在其他因素不变时，当库存成本不断增加时，考虑库存成本这一因素的企业平均利润下降趋势比不考虑库存成本这一因素的企业平均利润下降趋势慢。因此，在 MTO 企业订单接受问题进行建模求解过程中，将库存成本这一因素加以考虑，企业可以根据不同的库存成本做出不同的订单接受决策，以保证企业长期平均利润最大化；而且现实生活中，库存成本的存在常常会影响企业利润，占用企业资金，影响企业资金的运转，所以在订单接受过程中不能忽视库存成本这一因素。

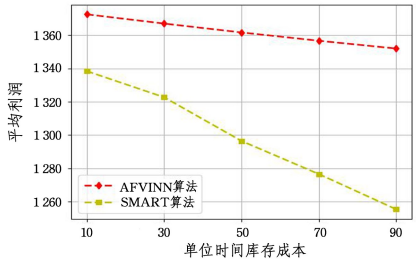


图 10　不同库存成本下的平均利润

Fig. 10　Average profit at differentinventory costs

已有文献大多是仅考虑订单特征，假设顾客是同等重要的[2-3,7-8,15,17]，虽然文献[16]运用强化学习思想进行建模和求解时涉及到了顾客优先级因素，但其仅仅将顾客优先级划分为3个等级，并且设定顾客订单类型仅为36种。在现实生活中，顾客订单类型多种多样，相应的顾客优先级也会存在多种。因此，本文将顾客优先级设置为 $\mu\in(0,1]$ 上的等概率分布，其中延期惩罚成本和拒绝成本与顾客优先级有直接的关系。在本节实验中，以基本情境为基准，首先在拒绝成本不变的情境下，改变单位延期惩罚成本；其次在单位延期惩罚成本不变的情境下，拒改变绝成本。

由表2可知：1)顾客优先级的存在会影响企业长期平均利润；2)基AFVINN算法下的顾客等级大于或等于0.5的订单接受率，随延期惩罚成本的增加而降低；而顾客等级小于0.5的订单接受率随延期惩罚成本的增加而上升；3)当拒绝成本增加，即拒绝成本对MTO企业利润影响越来越大时，AFVINN算法在进行订单接受决策时，顾客等级大于等于0.5的订单接受率呈上升趋势，而顾客等级小于0.5的订单接受率呈下降趋势。

表2 改变单位延期惩罚成本和拒绝成本

Table 2 Change unit delay penalty cost and rejection cost

参数	订单收益	$0<\mu\leqslant0.5$ 的订单接受率	$0.5<\mu\leqslant1$ 的订单接受率
$u=100$ $J=200$	1473.7378	0.1067	0.1093
$u=150$ $J=200$	1419.3591	0.1072	0.1018
$u=200$ $J=200$	1361.6615	0.1098	0.0957
$u=250$ $J=200$	1268.2084	0.1106	0.0920
$u=300$ $J=200$	1207.1880	0.1112	0.0891
$u=200$ $J=100$	1399.0198	0.1116	0.0918
$u=200$ $J=150$	1386.6070	0.1107	0.0937
$u=200$ $J=200$	1361.6615	0.1098	0.0957
$u=200$ $J=250$	1350.6669	0.1092	0.0986
$u=200$ $J=300$	1315.9328	0.1086	0.1005

这是因为当延期惩罚成本较大时，在接受顾客优先级较高的订单时，若企业没有在规定的期限完成生产，则需要付出较高的成本，所以高优先级顾客订单的接受率随延期惩罚成本的增加有所下降；当拒绝成本较高时，企业拒绝高优先级顾客的订单需要承担较高的费用，所以高优先级顾客订单的接受率随拒绝成本的增加有所上升。因此，当延期惩罚成本较大时，企业可以增加接受顾客优先级比较低的订单，而适当减少顾客优先级高的订单；当拒绝成本较大时，企业可以增加接受高优先级顾客的订单。所以在面对不同的延期惩罚成本和拒绝成本时，AFVINN算法能够及时调整订单接受策略，尽可能降低拒绝成本对MTO企业平均利润的影响，以使企业长期平均利润最大化。

结束语 本文在传统MTO企业订单接受决策问题考虑的因素基础上，增加了订单库存成本及多种顾客优先级因素，构建了马尔可夫决策过程订单接受模型，并运用AFVINN算法进行求解。该算法不仅能够将MTO企业订单接受问题中的多维状态空间转化为一维状态空间，简化了求解过程；而且通过与神经网络的结合，能够对状态空间进行泛化，可以处理较大规模的订单接受决策问题。

仿真实验表明，在MTO企业订单接受问题中，顾客优先级和库存成本因素对企业订单接受策略和利润是重要的；与基于传统Q-learning算法相比，基于AFVINN算法能够将高维控制问题转化为一维控制问题，提高样本的学习效率，简化求解过程；基于AFVINN算法在最大化企业长期平均利润方面优于Q-learning算法、FCFS方法以及greedy方法，该算法有较高的订单接受选择能力，而且对环境变化具有较好的适应能力，能够很好地权衡订单利润与各项成本因素，为MTO企业带来更高的利润。

在进一步的研究中，可将订单接受与订单调度问题进行联合建模，从而降低库存成本、延迟惩罚成本及拒绝成本对企业利润的影响，使得基于后状态强化学习算法在MTO企业订单接受问题的现实应用中更具有可行性和可靠性。

参 考 文 献

[1] MILLER B L. A Queueing Reward System with Several Customer Classes[J]. Management Science, 1969, 16(3): 234-245.

[2] ABEDI A, ZHU W H. An advanced order acceptance model for hybrid production strategy[J]. Journal of Manufacturing System, 2020, 55: 82-93.

[3] ZHANGX, MA S H. Order acceptance with limited capacity and finite output buffers in MTO environment[J]. Industrial Engineering and Management, 2008, 13(2): 34-38.

[4] GAO H L, DAN B, YAN J. Integrated order selection and scheduling decisions in the MTO environment considering the timeseries associations[J]. Journal of Management Engineering, 2017, 31(3): 108-116.

[5] FAN L F, CHEN X. Order Acceptance Policy based on EMSR-Method[J]. Management Review, 2010, 22(4): 109-113.

[6] WANG Z, QI Y Q, CUI H R, et al. A hybrid algorithmfor order acceptance and scheduling problem in make-to-stock/make-to-order industries[J]. Computers & Industrial Engineering, 2019, 127: 841-852.

[7] TARIK A, KOBE G, KUNAL K, et al. Production planning with order acceptance and demand uncertainty[J]. Computers and Operations Rsearch, 2018, 91: 145-159.

[8] FAN L F, CHEN X. Order pricing and acceptance policy in make-to-order firm based on revenue management[J]. System Engineer, 2011, 29(2): 87-93.

[9] LI X, VENTURA J A. Exact algorithms for a joint order acceptance and scheduling problem[J]. International Journal of Production Economics, 2020, 223: 107516.

[10] ROM W O, SLOTNICK S A. Order acceptance using genetic algorithms[J]. Computers & Operations Research, 2008, 36(6): 1758-1767.

[11] NOBIBON F T, LEUS R. Exact algorithms for a generalization of the order acceptance and scheduling problem in a single-ma-

chine environment[J]. Computers & Operations Research, 2010,38(1):367-378.

[12] CESARET B,OGUZ C,SALMAN F S. A tabu search algorithm for order acceptance and scheduling[J]. Computers and Operations Research,2010,39(6):1197-1205.

[13] WANG L,XU Z Y,ZHAO Y,et al. Model and algorit-hm for order acceptance on multi-node production environment with limited buffer[J]. Chinese Journal of Management Science, 2015,23(12):135-141.

[14] RAHMAN H F,JANARDHANAN M N,NIELSEN I E. Real-time order acceptance and scheduling problems in a flow shop environment using hybrid GA-PSO algorithm[J]. IEEE Access, 2019,7:112742-112755.

[15] lLI X P,WANG J,SAWHNEY R. Reinforcement learning for joint pricing,lead-time and scheduling decisions in make-to-order systems[J]. European Journal of Operational Research, 2012,221(1):99-109.

[16] ARREDONDO F,MARTINEZ E. Learning and adaptation of a policy for dynamic order acceptance in make-to-order manufacturing[J]. Computers and Industrial Engineering,2009,58(1): 70-83.

[17] HAO J,YU J J,ZHOU W H. Order acceptance policy in make-to-order manufacturing based on average-reward reinforcement learning[J]. Journal of Computer Applications, 2013, 33(4): 976-979.

[18] WANG X H,WANG N N,FAN Z P. Reinforcement learning based order acceptance policy in make-to-order enterprises[J]. System Engineering-Theory & Practice, 2014, 34(12): 3121-3129.

[19] SUTTON R S,BARTO A G. Reinforcement learning:An introduction[M]. Cambridge:Cambridge University,2011.

[20] LEWICKI G,MARINO G. Approximation by superpositions of a sigmoidal function[J]. Journal for Analysis and Its Applications,2003,22(2):463-470.

[21] MITCHELL T. Machine Learning[M]. New York:McGraw-Hill,1997.

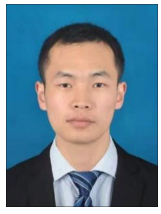
[22] RIEDMILLER M. Neural fitted Q iteration-first experiences with a data efficient neural reinforcement learning method [C]// Machine Learning: European Conference on Machine Learning (ECML) 2005. Porto:Portugal,2005:317-328.

[23] HERBOTS J,HERROELEN W,LEUS R. Dynamic order acceptance and capacity planning on a single bottleneck resource [J]. Naval Research Logistics,2007,54(8):874-889.

[24] HING M M,HARTEN A V,SCHUUR P. Reinforcement learning versus heuristics for order acceptance on a single resource [J]. Journal of Heuristics,2007,13(2):167-187.

[25] CHARNSIRISAKSKUL K,GRIFFIN P M,KESKINOCAK P. Order selection and scheduling with leadtime flexibility[J]. IIE Transactions,2004,36(7):697-707.

QIAN Jing, born in 1998, postgraduate. Her main research interests include reinforcement learning and computer intelligent decision making technology.



WU Ke-yu, born in 1990, assistant professor. His main research interests include reinforcement learning, deep learning and their applications in networked systems.