文章编号：0253-374X(2021)10-1417-09

DOI：10.11908/j.issn.0253-374x.21021

基于逆向强化学习的装船时堆场翻箱智能决策

张艳伟，蔡梦蝶

（武汉理工大学 交通与物流工程学院，湖北 武汉 430063）

摘要：集装箱码头装船时堆场翻箱具有时序性与动态性，属于NP(non-deterministic polynomial)难问题。针对常见的顺岸式集装箱码头堆场，以最小化总翻箱次数为优化目标，考虑翻箱对装船连续性及效率的影响，基于马尔科夫决策过程构建装船时堆场翻箱模型，设计逆向强化学习算法。为验证算法的有效性，以随机决策为基准，将设计的逆向强化学习算法与码头常见规则决策、随机决策对比。结果表明，贝位堆存状态不佳时，常见的规则决策不一定优于随机决策；逆向强化学习算法可有效挖掘隐含专家经验，收敛至最小翻箱次数的概率更高，且不同堆存状态下均能更好地限制单次发箱的翻箱次数，可实现装船时堆场翻箱智能决策。

关键词：集装箱码头；堆场翻箱；智能决策；马尔科夫决策过程；逆向强化学习

中图分类号：U695.22　　**文献标志码**：A

An Inverse Reinforcement Learning Method for Container Relocation in Container Terminal Yard During Loading

ZHANG Yanwei, *CAI Mengdie*

(School of Transportation and Logistics Engineering, Wuhan University of Technology, Wuhan 430063, China)

Abstract: The container relocation during loading in the terminal yard has sequential and dynamic characteristics, and belongs to the non-deterministic polynomial hard problem. This paper takes the common container terminal yard, which is parallel to the shoreline, as the research object. Considering the relocation effect on the continuity and efficiency of shipment, the model based on Markov decision processes for the container relocation in the yard during loading was proposed, with the optimization objective to minimize the total relocation times, and the algorithm based on inverse reinforcement learning was designed. To verify the effectiveness of the algorithm, taking the random decision as criterion, the inverse reinforcement learning algorithm was compared with the common rule decision-making and the random decision-making . The results show that when the initial state of the bay is unsatisfactory, the common rule decision-making is not necessarily superior to random decision-making. The inverse reinforcement learning algorithm can effectively mine and apply the expert experience, and the probability of converging to the minimum relocation times is obviously better than that of the others. In addition, it can better control the relocation times of a single loading in different state of the bay, and realize the intelligent decision-making of container relocation during loading.

Key words: container terminal; yard relocation; intelligent decision-making; Markov decision processes; inverse reinforcement learning

集装箱码头堆场装船发箱和翻箱效率直接影响岸边装船作业和船舶靠泊时间，是提升集装箱港口竞争力的重要突破口之一。受船舶配载等不确定信息的影响，堆场装船前预翻箱整理往往不能完全满足发箱要求，装船时翻箱作业不可避免。由于装船时翻箱不产生价值[1]，单次发箱的翻箱次数过多会影响装船作业连贯性，优化装船时堆场翻箱方案，实现智能高效决策，是提高堆场装船发箱和翻箱效率、保障岸边装船作业的有效手段。集装箱码头堆场通常划分为不同箱区，每个箱区由多个贝位组成。一个贝位的长度为一个标准集装箱长，贝位内通常可堆垛6列4层及以上，即包含24个集装箱箱位及以上。由于多层直接堆垛、贝位内集装箱多且每个集装箱具有唯一属性、一个集装箱可能存在多次翻箱等，堆场翻箱问题解空间大，当堆存状态不佳时，甚

收稿日期：2021-01-14
基金项目：国家自然科学基金(60904067)
第一作者：张艳伟(1977—)，女，副教授，工学博士，主要研究方向为智慧港航、港口物流、智能决策与算法。
E-mail：zywtg@whut.edu.cn

论文拓展介绍

至无法在有效时间内得到可行方案,具有NP难属性[2]。因此,装船时翻箱问题(container relocation problem, CRP)一直是研究热点之一。关于CRP的研究,主要包括数学模型研究、精确求解方法研究和智能算法等方面。

Caserta等[3]提出二元线性规划模型CRP-Ⅱ,Expósito-Izquierdo等[4]指出CRP-Ⅱ会产生不可行解,难以保证解最优,将其改进为CRP-Ⅱ*。Zehendner等[5]也对CRP-Ⅱ进行了修正。Petering等[6]提出混合整数线性规划模型CRP-Ⅲ,整数决策变量比CRP-Ⅱ少。Galle等[7]改进二进制整数编码方式,提出二元整数规划模型CRP-Ⅰ。Jin [8]指出CRP-Ⅰ中后进先出约束存在缺陷,设计整数变量将其变为线性约束。常见的CRP求解分为精确求解[9-12]和智能算法求解[13-15]。Tanaka等[9]提出严格的下界用于分枝定界算法。Tricoire等[10]设计基于启发式规则的分枝定界算法。Forster等[11]设计启发式树搜索算法,得到基于翻箱移动序列的分枝方案。Tanaka等[12]提出消除树搜索不必要节点的方法。Ting等[13]设计波束搜索算法,并用启发式评估搜索节点。Bacci等[14]设计有界波束搜索算法,关注最有希望的节点以缩小搜索空间。Feillet等[15]基于局部搜索设计启发式算法,优化局部翻箱操作序列。

逆向强化学习(inverse reinforcement learning, IRL)是机器学习的重要分支,可挖掘示例数据中隐含信息,克服已有智能算法难以挖掘专家经验的局限,为实现装船时堆场翻箱智能决策奠定基础。Bengio等[16]对机器学习求解组合优化问题进行综述,主张结合机器学习与已有的组合优化算法,实现算法的改进与创新。强化学习(reinforcement learning, RL)在人为设计回报函数后,基于奖惩机制使收益最大化,可实现组合优化问题的序列决策[17]。为克服RL人为确定回报函数的局限,IRL将示例轨迹作为训练数据,实现回报函数的自动构建[18]。陈希亮等[19]综述深度IRL时,回顾了学徒学习、最大边际规划、结构化分类等经典IRL算法,指出IRL可实现专家示例数据高效利用。Lin等[20]将回报函数视为以特征期望为参数的函数,从专家示例中学习策略。Abbeel等[21]研究马尔科夫决策过程(Markov decision processes, MDP)时,假设回报函数是已知特征的线性组合,并根据专家正在优化该回报函数这一假设,设计学徒学习算法。杨放青等[22]运用学徒学习解决调度优化问题,基于MDP构建仿真模型并学习专家示范调度。

现有研究的CRP模型通常以最小化堆场总翻箱次数为优化目标,较少限制单次发箱对应的翻箱次数;算法难以有效挖掘隐含专家决策经验,堆存状态不佳时,优化效果有限。为此,论文从模型及算法两个方面进行研究,实现以下创新:

(1)以最小化总翻箱次数为目标,同时限制单次发箱对应的翻箱次数,将堆场翻箱过程描述为MDP,构建装船时堆场翻箱动态决策模型。

(2)设计基于IRL的装船时堆场翻箱智能决策算法,挖掘并应用翻箱方案中隐含的专家决策信息。

论文设计的模型和算法可以实现装船时堆场翻箱智能决策,能有效解决各种堆存状态下的装船时堆场翻箱决策问题。

1 装船时堆场翻箱问题描述

以常见的顺岸式集装箱码头为对象,研究装船时堆场翻箱问题。贝位内空箱位编号为0,集装箱发箱顺序用非0不重复编号表示,编号越小发箱越早,编号最小的集装箱为目标箱。各列最早发箱的集装箱不在最上层时,位于其上的集装箱为阻塞箱。如图1所示,以6列4层21个集装箱的贝位为例,空箱位为第4、5、6列的第4层,第3列第3层编号为1的集装箱为目标箱,1号集装箱的阻塞箱为3号集装箱。

装船时堆场发箱作业包括取箱操作和翻箱操作。取箱操作将位于列最上层的目标箱移至水平搬运设备,搬运至岸边装船;翻箱操作将目标箱上方的阻塞箱依次移至贝位内空箱位,直至目标箱位于最上层。装船时堆场翻箱落箱位在翻箱前为空箱位,需满足集装箱翻箱后不悬空。翻箱决策需避免后续不必要翻箱,减少总翻箱次数,同时保障单次发箱对应的翻箱次数均衡。图1中,对1号集装箱发箱时,阻塞箱3选择第4列第4层作为翻箱落箱位,可避免后续翻箱。完成1号集装箱发箱后,贝位堆存状态如图2所示,对2号集装箱发箱时,阻塞箱8的翻箱落箱位不选择第6列,可避免第6列中6号集装箱发箱时连续翻箱3次。

2 基于MDP的装船时堆场翻箱模型构建

2.1 模型假设

根据顺岸式集装箱码头生产实际做如下假设:

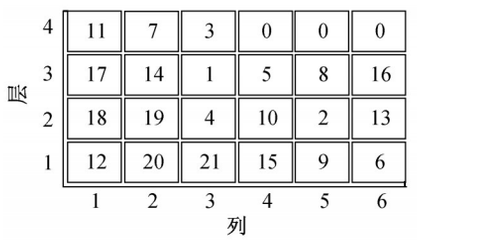

图1　堆场贝位初始堆存状态

Fig.1　Initial storage state of bay in storage yard

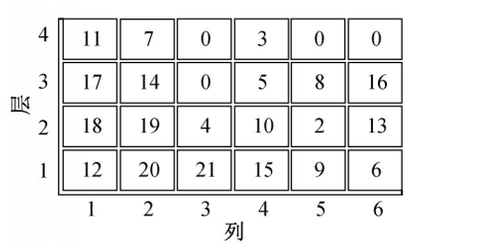

图2　集装箱1发箱后堆存状态

Fig.2　Storage status after delivering of container 1

(1)贝位内仅堆存同一尺寸类型集装箱,各拟装船集装箱发箱顺序已知且唯一。

(2)装船过程中,不允许其他集装箱进入拟装船集装箱所在贝位。

(3)翻箱操作发生在单一堆场贝位内,翻箱仅针对目标箱上方的阻塞箱。

2.2　符号说明

2.2.1　状态变量

I:贝位列标号集合。

H:贝位层标号集合。

K:贝位内拟装船集装箱集合。

k:当前目标箱发箱顺序,$k\in K$。

i:贝位列标号,$i\in I$。

j:贝位层标号,$j\in H$。

ij:作为下标时,表示第i列第j层的集装箱箱位。

n、n':当前目标箱的阻塞箱数量,$n=0$时无阻塞箱,$n\neq n'$。

s:贝位堆存状态,贝位同型矩阵,元素值为对应箱位集装箱发箱顺序,对应箱位为空时取0。

s_n^k:目标箱k有n个阻塞箱的贝位状态。

s':状态s的下一个堆存状态,其中,状态s不是最终状态。

S:贝位所有可能的堆存状态构成的有限集合,$s_n^k, s_{n-1}^k, s_{n'}^{k+1}, s, s'\in S$。

t:发生状态转移的时刻。初始堆存状态下,$t=0$,状态s不是最终状态时,转移到s时刻为t,转移到s'的时刻为$t+1$,定义状态与时刻的对应关系后,作为下标可与状态s通用。

T:状态转移时刻集合,$t\in T$。

N_t^k:时刻t目标箱k前一个集装箱完成发箱,目标箱k上方阻塞箱总数。

b_n^k:目标箱k上方n个阻塞箱中最上层的阻塞箱,无阻塞箱时,$b_n^k=0$。

a、a':堆场贝位内发箱作业动作,分为取箱和翻箱,目标箱有阻塞箱时为翻箱动作,目标箱无阻塞箱时为取箱动作,翻箱落箱位不同,作业动作不同,$a\neq a'$。

A:作业动作集合,翻箱时,元素为落箱位不同的翻箱动作,取箱时,元素为一个取箱动作,$a, a'\in A$。

$p_{ss'}^a$:堆存状态转移概率,为状态s下采取动作a后,转移到s'的概率。

P:堆存状态转移概率集合,$p_{ss'}^a\in P$。

γ:折扣因子,平衡当前奖励与未来奖励,$\gamma\in[0,1]$。

G_t:从时刻t到最终状态,获得的累积奖励。

π:发箱动作选择策略,为既定状态下执行所有可能动作的概率分布。

$q_\pi(s,a)$:状态动作值函数,表示从状态s出发,采取动作a后,再使用策略π的累积奖励。

$R(s)$:回报函数,计算状态转移时环境给出的奖励值,$R_t=R(s)$时,$R_{t+1}=R(s')$。

$v_\pi(s)$:价值函数,用于计算状态s下,采取策略π转移到最终状态的累积奖励期望。

M:一个正整数,用于限制单次发箱的翻箱次数,堆存状态良好时可取值为2,否则取值为初始贝位堆存状态下各列中最大阻塞箱数。

2.2.2　中间变量

d_{ij}:表示第i列第j层的集装箱箱位是否被占用,被占用时为1,否则为0。

x_{ij}:表示第i列第j层的集装箱箱位是否悬空,悬空时为1,否则为0。

l_k:表示是否对目标箱k最上层阻塞箱b_n^k以外的集装箱翻箱,是则为1,否则为0。

z_i:表示第i列的阻塞箱数量是否超过$M-1$,超过时为1,否则为0。

2.2.3　决策变量

$a_{ij}(s_n^k)$:0-1变量,在状态s_n^k下翻箱且选择第i列第j层的集装箱箱位时为1,否则为0(取箱时取

值为0)。

2.3 约束与目标函数

初始贝位已知时,取箱次数等于拟装船集装箱数量,翻箱次数越少,发箱-翻箱移动作业序列越短。每次作业动作后,贝位状态发生转移,记非最终状态s为s_n^k,最上层阻塞箱为b_n^k,转移情况为:若$n=0$,目标箱k阻塞箱为0,发箱动作后转移到s',目标箱更新为$k+1$,目标箱的阻塞箱数量更新为n',最上层阻塞箱更新为$b_{n'}^{k+1}$,s'为$s_{n'}^{k+1}$;若$n\neq 0$,对阻塞箱b_n^k翻箱,目标箱的阻塞箱数量减一,最上层阻塞箱序号更新,s'为s_{n-1}^k。初始贝位状态经过取箱、翻箱及落箱位决策,逐步更新到最终状态,翻箱方案记录每次翻箱操作动作。装船时堆场发箱过程时序性明显,描述为MDP,记为$\{S, A, P, R, \gamma\}$。以最小化总翻箱次数为优化目标,同时控制单次发箱对应的翻箱次数,构建基于MDP的翻箱模型,目标函数如式(1)所示。

$$\min \sum_{k\in K}\sum_{n}^{N_t^k} a_{ij}(s_n^k), t\in T \tag{1}$$

$$\sum_{k\in K}\sum_{n}^{N_t^k} a_{ij}(s_n^k) d_{ij}=0, \forall i\in I, \forall j\in H, t\in T \tag{2}$$

$$\sum_{k\in K}\sum_{n}^{N_t^k} a_{ij}(s_n^k) x_{ij}=0, \forall i\in I, \forall j\in H, t\in T \tag{3}$$

$$\sum_{k\in K}\sum_{n}^{N_t^k} a_{ij}(s_n^k)\, l_k=0, \forall i\in I, \forall j\in H, t\in T \tag{4}$$

$$\sum_{k\in K}\sum_{n}^{N_t^k} a_{ij}(s_n^k) z_i=0, \forall i\in I, \forall j\in H, t\in T \tag{5}$$

$$p_{ss'}^a=P[S_{t+1}=s'|S_t=s, A_t=a], t\in T \tag{6}$$

$$G_t=R_{t+1}+\gamma R_{t+2}+\gamma^2 R_{t+3}+\cdots=\sum_{n=0}^{\infty}\gamma^n R_{t+n+1}, \quad t\in T \tag{7}$$

$$\pi(a|s)=P[A_t=a, S_t=s], t\in T \tag{8}$$

$$q_\pi(s,a)=E_\pi[G_t|S_t=s, A_t=a]=E_\pi[R_t+\gamma\sum_{a\in A}p_{ss'}^a v_\pi(s')], t\in T \tag{9}$$

$$v_\pi(s)=E_\pi[R_{t+1}+\gamma v(S_{t+1})|S_t=s]=\sum_{a\in A}\pi(a|s)q_\pi(s,a), t\in T \tag{10}$$

式(2)表示每次翻箱时,落箱位决策仅针对当前状态中的空箱位;式(3)表示每次翻箱后,集装箱不悬空;式(4)表示每次翻箱仅针对当前目标箱最上层的阻塞箱;式(5)表示状态s经过翻箱转移到s'时,选择阻塞箱数量小于$M-1$的列,限制单次发箱的翻箱次数。

MDP具有动态性,状态转移与时刻t有关,可用时刻t表示状态。式(6)到式(10)为MDP相关理论。式(6)为状态s下采取动作a,转移到状态s'的概率,$p_{ss'}^a$取值与策略有关,随机选择翻箱落箱位时,取值为$|A|$的倒数。每次状态转移环境均会给出奖励值,若时刻t对应状态为s,式(7)表示从状态s转移到最终状态,得到的累积奖励值,距离s越远的状态受s影响越小,引入折扣因子平衡当前奖励与未来奖励。式(8)中,将状态s下动作a的概率分布记为$\pi(a|s)$。由价值函数和状态动作值函数的定义,可以得到式(9)。根据式(8)和式(9),推出价值函数与状态动作值函数关系如式(10)。

基于MDP的装船时堆场翻箱问题,旨在找到一种策略$\pi(s)$,在状态s下采取对应的动作a,使总翻箱操作回报累计值最大,即每次根据当前状态s选择翻箱动作时,选择价值函数最大的状态作为s',最大状态价值函数由最优状态动作值函数得到。用*表示最优,装船时堆场翻箱动态决策的目标函数可改写成如式(11)和式(12)所示。

$$v_*(s)=\max_a [R_t+\gamma\sum_{s'\in A}P_{ss'}^a v_*(s')], t\in T \tag{11}$$

$$q_*(s,a)=R_t+\gamma\sum_{s'\in A}p_{ss'}^a\max_{a'}q_*(s',a'), t\in T \tag{12}$$

3 基于IRL的求解算法设计

装船时堆场翻箱问题具有MDP特性,可用RL方法求解。为克服RL回报函数依赖人为确定的局限,设计基于IRL的装船时堆场翻箱智能决策算法,挖掘专家翻箱方案中隐含的决策经验,应用于装船时堆场翻箱落箱位选择,实现装船时堆场翻箱智能决策。

3.1 专家示例表示方法

RL中状态转移后,环境会给出奖励,训练数据是状态、动作和奖励值构成的系列。不同于RL,IRL的输入是专家示例,不需环境给出奖励,训练数据是状态-动作序列。装船时翻箱问题中,发箱动作矩阵可由相邻状态矩阵相减求出。以图1贝位状态为例,初始状态为矩阵$\boldsymbol{s}_0$,将3号集装箱翻箱至第4列第4层得到矩阵$\boldsymbol{s}_1$,完成1号集装箱取箱后得到矩阵$\boldsymbol{s}_2$。翻箱动作矩阵$\boldsymbol{a}_1$和取箱动作矩阵$\boldsymbol{a}_2$如下:

$$\boldsymbol{a}_1=\boldsymbol{s}_1-\boldsymbol{s}_0=\begin{bmatrix}0&0&-3&+3&0&0\\0&0&0&0&0&0\\0&0&0&0&0&0\\0&0&0&0&0&0\end{bmatrix}$$

$$\boldsymbol{a}_2=\boldsymbol{s}_2-\boldsymbol{s}_1=\begin{bmatrix}0&0&0&0&0&0\\0&0&-1&0&0&0\\0&0&0&0&0&0\\0&0&0&0&0&0\end{bmatrix}$$

在翻箱、取箱动作矩阵中，非0元素绝对值为被操作集装箱发箱顺序编号，元素位置对应于堆场贝位中位置。翻箱动作矩阵中，负值元素位置对应堆场贝位中翻箱动作起始位置，正值元素位置对应翻箱落箱位。取箱动作矩阵中无正值元素，负值元素位置对应当前目标箱所在集装箱箱位。发箱动作矩阵可由相邻状态矩阵计算得到，此时，专家示例可只用贝位状态序列表示。

3.2　基于IRL的回报函数确定

基于IRL构建回报函数，需先将状态映射为状态特征向量，并根据专家示例进行求解，相关理论如下：

$$R(s)=\boldsymbol{\omega}\phi(s) \tag{13}$$

$$\mu_\pi(s)=E\left[\sum_{n=0}^{\infty}\gamma^n\phi(s)|\pi\right] \tag{14}$$

$$v_\pi(s)=E\left[\sum_{n=0}^{\infty}\gamma^n\boldsymbol{\omega}\phi(s)|\pi\right]=\boldsymbol{\omega}\mu_\pi(s) \tag{15}$$

$$\left|E\left[\sum_{n=0}^{\infty}\gamma^nR_s|\pi^*\right]-E\left[\sum_{n=0}^{\infty}\gamma^nR_s|\bar{\pi}\right]\right|\leqslant\delta \tag{16}$$

其中，$\phi(s)$：$s\rightarrow\mathbf{R}^n$是基函数，将状态s映射到状态特征向量中。$\boldsymbol{\omega}\in\mathbf{R}^n$为参数向量，是根据专家示例构建回报函数的关键。$\pi^*$表示专家方案，$\bar{\pi}$表示所求方案，$\delta$为极小的正数。

IRL假设待求回报函数下专家示例最优，回报函数为状态特征向量和参数向量的内积，如式(13)所示，基于假设求解参数向量$\boldsymbol{\omega}$，即可构建回报函数。定义$\mu_\pi(s)$为特征期望，如式(14)所示。由式(13)和式(14)可将价值函数改写为式(15)。式(16)表示得到的方案$\bar{\pi}$在特征期望上接近专家方案π^*。

特征向量需简洁全面地刻画堆场贝位堆存状态，特征向量元素选取当前目标箱发箱顺序、目标箱的阻塞箱数量、有空箱位的列的数量、空列的数量。进行翻箱落箱位选择，需分析有空位的列的性质，为此，判断当前目标箱最上层阻塞箱发箱顺序是否晚于其他列的最早发箱顺序，并关注当前堆存状态下阻塞箱数量达到M的列数。每次进行翻箱动作选取之前，采取较为常见的蒙特卡洛模拟方法，计算下一个状态所有可能的特征期望。

3.3　IRL算法设计

回报函数影响RL的求解质量，人为确定具有很强的主观性，复杂问题甚至难以给出直观的回报函数。为此，以专家翻箱方案为训练数据，基于IRL还原回报函数，同时结合RL方法，设计基于IRL的装船时堆场翻箱智能决策算法，挖掘并应用专家翻箱方案中隐含决策经验，实现装船时堆场翻箱智能决策。

IRL分为最大边际和最大熵两大类，前者包括学徒学习、最大边际规划、结构化分类等，后者包括最大熵、相对熵和深度IRL等。学徒学习算法所需数据少，决策空间离散时可定量比较各策略，为此，设计学徒学习算法从专家翻箱方案中学习回报函数，结合RL实现问题求解：先基于学徒学习还原专家示例中的回报函数，用于RL进行策略迭代；对比当前策略与专家策略，基于最大边际求参数向量$\boldsymbol{\omega}$。循环以上两步，改进回报函数至能反应专家意图为止。装船时堆场翻箱智能决策算法流程如图3所示。

4　算例分析

4.1　对比策略设计

装船时堆场贝位按照既定发箱顺序发箱，若当前目标箱位于贝位内列的最上层，直接提箱装船；若当前目标箱不位于贝位内列的最上层，依次对其上方各集装箱进行翻箱操作，直至当前目标箱位于列的最上层，提箱装船。翻箱决策的本质是对当前拟翻箱的集装箱进行落箱箱位选取，最基础的翻箱决策不考虑总翻箱次数及单次发箱对应的翻箱次数，随机选择不悬空的空箱位作为翻出集装箱的落箱位。

由于随机决策无任何优化，在码头生产实际中，通常考虑翻出集装箱对后续装船的影响，依照一定的经验规则选取落箱箱位，尽量避免翻出集装箱再次阻塞贝位内其他集装箱，减少总翻箱次数。

基于规则的决策在集装箱码头生产实际中被广泛使用，但贝位堆存状态不佳时，其优化效果有限，甚至存在劣于随机决策的可能。

为验证本文算法与模型的有效性，将设计的IRL算法与码头常见规则决策、随机决策对比。

策略一：随机决策。随机选择不悬空的空箱位，作为翻出集装箱的落箱位。

策略二：规则决策。在有不悬空空箱位的列中，优先选取列内最早发箱顺序晚于被翻集装箱的列，作为翻出集装箱的落箱位；没有这样的列时，随机选择空列；没有空列时，随机选择不悬空的空箱位，作

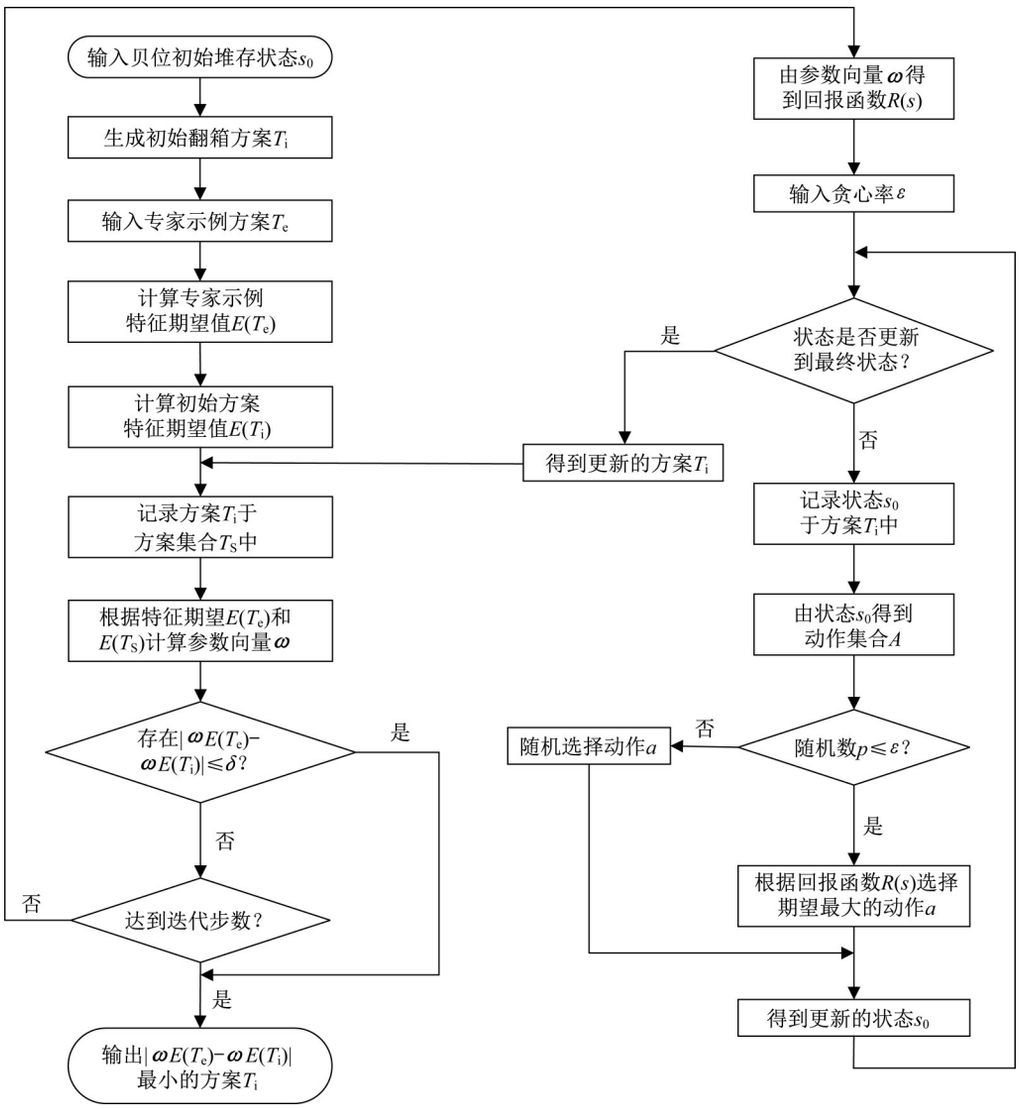

图3 基于IRL的装船时堆场翻箱智能决策算法流程

Fig.3 Flow chart of intelligent decision algorithm based on IRL for yard relocation during loading

为翻出集装箱的落箱位。

策略三:基于IRL的决策。挖掘并应用专家翻箱方案中隐含决策经验,设计基于IRL的装船时堆场翻箱智能决策算法,基于奖惩机制选择奖励期望最大的不悬空空箱位,作为翻出集装箱的落箱位。

4.2 算例设计

吞吐量和装卸效率是集装箱码头的重要统计指标,目前已形成完整的统计体系。相比之下,集装箱堆场翻箱率为翻箱次数与总操作次数的比值,受采集技术及堆存状态波动等影响,目前只有少数港口对翻箱率进行过初步统计,尚未形成统一有效的统计体系。根据集装箱码头生产实际,装船时堆场翻箱率通常为15%至30%不等,堆场堆存空间不足、集装箱堆存状态欠佳时,将有所增加。算例以常见的6列4层贝位为例,考虑翻箱空间需求,贝位内最多堆存21个集装箱,选取翻箱率20%、30%、40%进行算例设计较为合理,对应翻箱次数为4.2、6.3和8.4次。考虑到算例设计时翻箱方案和翻箱次数未知且需要设置初始堆存状态,为此,进行简化将贝位初始状态阻塞箱数量设置为4个、6个和8个。由于贝位初始状态阻塞箱总数是装船时堆场贝位翻箱次数的下界,该设置方法对应的翻箱率一定大于或约等于20%、30%和40%,比集装箱码头堆场实际堆存状态复杂,能够适应堆场翻箱决策需要。

定义贝位初始状态阻塞箱总数与待发集装箱总数比值为阻塞箱占比。如图4中3种贝位初始状态,阻塞箱个数分别为4个、6个和8个,阻塞箱占比依次为19.0%、28.5%和38.1%。随着贝位堆存状态变差,一般会出现连续多次翻箱的情况。M取值为3,从策略一、策略二中选择总翻箱次数少,连续翻箱3

次情形少的方案作为专家示例。电脑处理器为Intel(R)Core(TM)i7-8750H CPU@2.20GHz，运行内存8 GB，基于Python实现3种状态下随机决策、规则决策和基于IRL的决策各100次。

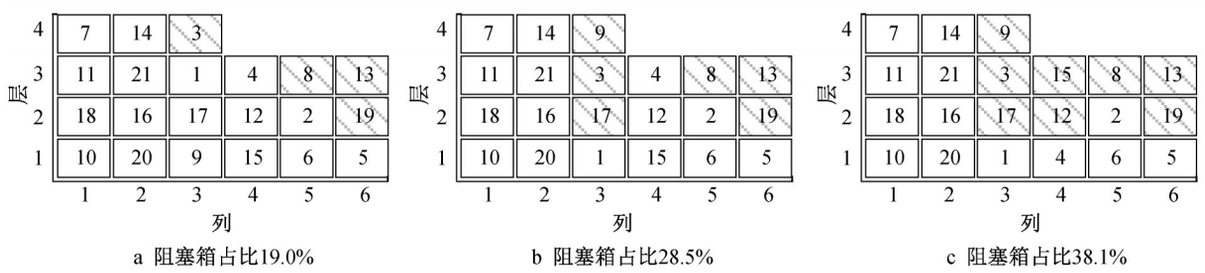

图4　3种堆场贝位初始堆存状态

Fig.4　Initial storage state of bay in three scenarios

4.3　结果分析

3种状态下策略一、策略二和策略三对应的100个方案的翻箱次数箱形图如图5所示。从图5可以看出，基于IRL的决策整体上求解质量较高。相对于随机决策和规则决策，基于IRL决策的总翻箱次数下限和中位数小，贝位阻塞箱占比较大时，翻箱次数集中分布在较小翻箱次数处。表明基于IRL的翻箱决策优势明显，可实现专家经验的有效利用。

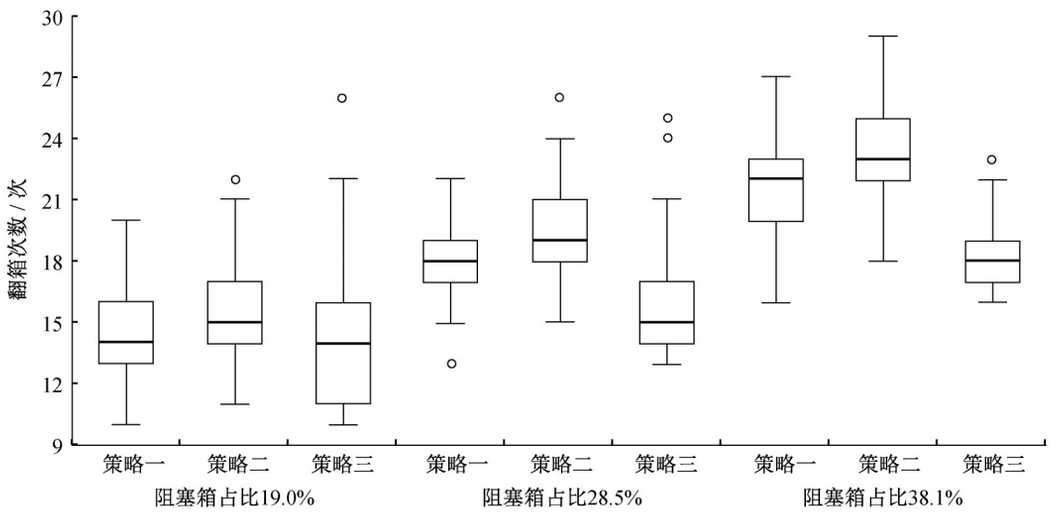

图5　3种策略的翻箱次数箱形图

Fig.5　Box plot of relocation times of three policies

为避免岸桥等待集装箱，装船时贝位翻箱需限制单次发箱的翻箱次数，优秀的翻箱策略更易形成总翻箱次数较少的方案，同时可限制方案中单次发箱的翻箱次数。以最小总翻箱次数及其出现概率、连续多次翻箱的概率为重要指标，3种策略的翻箱方案结果如表1所示。

由表1可知：

(1)基于IRL的决策可产生总翻箱次数最少的方案，且该类方案出现的次数明显高于其他2种策略。说明其他2种策略随机性大，基于IRL的决策对应的方案收敛至最小翻箱次数的概率显著增大。

(2)阻塞箱占比增加时，规则决策适应性不强，阻塞箱占比28.5%时，出现3回及以上连翻3次翻箱的方案数远高于其他2种策略。说明规则设计与贝位初始状态有关联，提炼规则用于装船时堆场翻箱决策的难度较大。

(3)基于IRL的决策在阻塞箱占比较大时，连续翻箱3次为3回及以上的方案数显著少于其他2种策略。说明贝位堆存状态不佳时，基于IRL的决策在限制单次发箱的翻箱次数上优势更明显。

(4)堆场贝位堆存较满且阻塞箱占比约20.0%及以上时，很难避免多次翻箱，翻箱率远高于阻塞箱占比。

Kim等[23]提出预期翻箱次数(expected number of additional relocations, ENAR)概念，设计启发式算法OH。徐亚等[24]赋予有空位的列不同的优先级，

表1 3种策略下翻箱方案结果

Tab.1 Results of three kinds of decision-making schemes

阻塞箱占比/%	19.0	19.0	19.0	28.5	28.5	28.5	38.1	38.1	38.1
策略	策略一	策略二	策略三	策略一	策略二	策略三	策略一	策略二	策略三
最小总翻箱次数	10	11	10	13	15	13	16	18	16
最小总翻箱次数出现的次数	1	1	11	1	2	19	1	1	10
最小总翻箱次数对应的翻箱率/%	47.6	52.4	47.6	61.9	71.4	61.9	76.2	85.7	76.2
不出现连翻3次翻箱的方案数	28	37	40	0	0	0	0	0	0
出现1回连翻3次翻箱的方案数	44	55	42	14	5	10	0	0	0
出现2回连翻3次的方案数	23	5	15	52	34	66	10	16	56
出现3回及以上连翻3次翻箱的方案数	5	3	3	34	61	24	90	84	44

基于ENAR提出另一种启发式算法IH。游鑫梦等[25]在堆场箱区及贝位分配的基础上,针对多种堆存情形提出翻箱落箱位选取规则,设计启发式算法与OH算法及IH算法对比,分析不同贝位规模下算法的性能。

为验证IRL的有效性,选取常见的6列4层贝位规模,与OH算法[23]、IH算法[24]和启发式算法[25]对比,算法求解方案的平均翻箱次数和平均运行时间如表2所示。由表2可知,IRL可以有效地控制总翻箱次数。由于IRL算法在每次状态转移时均需计算所有下一个状态的价值函数,总循环迭代次数多,求解时间相对较长,但耗时在可接受范围内。

表2 不同算法的平均翻箱量和平均运行时间

Tab.2 Average relocation times and average running time of different algorithms

算法性能评价指标	OH算法	IH算法	启发式算法	IRL算法
平均翻箱量/次	18.04	14.85	14.40	14.39
平均运行时间/s	0.08	0.06	0.05	12.48

5 结语

装船时集装箱堆场发箱作业时序性明显,以最小化总翻箱次数为目标,同时限制连续多次翻箱,基于MDP构建混合整数模型。为克服RL回报函数难以确定的局限,设计IRL算法学习专家示例,挖掘专家经验应用于翻箱决策中。以随机决策为基准,将基于IRL的决策与码头常见规则决策、随机决策对比。基于IRL决策时,总翻箱次数收敛于最小翻箱次数的概率更大;堆场贝位状态较差时,优化方案总翻箱次数集中分布于较小翻箱次数处,单次发箱时翻箱多次的情况得到有效限制。与已有智能算法进行对比,发现IRL可以有效控制总翻箱次数,受算法循环迭代次数多的影响,耗时相对较长,但在可接受范围。基于IRL制定翻箱方案可解决常见智能算法难以提炼并应用专家经验的局面,实现专家决策中隐含经验在翻箱决策中的智能应用。

作者贡献声明:

张艳伟:提出研究方向,设计论文框架并指导模型构建、论文撰写。

蔡梦蝶:构建模型,进行编程求解及论文撰写。

参考文献:

[1] 宓为建, 张晓华, 秦婴, 等. 集装箱码头装船贝内发箱顺序决策[J]. 中国工程机械学报, 2016, 14(4): 369.
MI Weijian, ZHANG Xiaohua, QIN Zhao, *et al*. Decisions on container retrievingorders forcontainer terminals [J]. Chinese Journal of Construction Machinery, 2016, 14(4): 369.

[2] 郑斯斯, 王爱虎. 路径优化算法求解集装箱码头堆场翻箱问题[J]. 工业工程与管理, 2017, 22(3): 31.
ZHENG Sisi, WANG Aihu. Paths optimum algorithm for the container relocation problem in the container terminals [J]. Industrial Engineering and Management, 2017, 22(3): 31.

[3] CASERTA M, SCHWARZE S, VOß S. A mathematical formulation and complexity considerations for the blocks relocation problem [J]. European Journal of Operational Research, 2012, 219(1): 96.

[4] EXPÓSITO-IZQUIERDO C, MELIÁN-BATISTA B, MORENO-VEGA J M. An exact approach for the blocks relocation problem [J]. Expert Systems with Applications, 2015, 42(17/18): 6408.

[5] ZEHENDNER E, CASERTA M, FEILLET D, *et al*. An improved mathematical formulation for the blocks relocation problem[J]. European Journal of Operational Research, 2015, 245(2): 415.

[6] PETERING M E H, HUSSEIN M I. A new mixed integer program and extended look-ahead heuristic algorithm for the block relocation problem [J]. European Journal of Operational Research, 2013, 231(1): 120.

[7] GALLE V, BARNHART C, JAILLET P. A new binary formulation of the restricted container relocation problem based on a binary encoding of configurations[J]. European Journal of Operational Research, 2018, 267(2): 467.

[8] JIN B. On the integer programming formulation for the relaxed restricted container relocation problem[J]. European Journal of Operational Research, 2020, 281(2): 475.

[9] TANAKA S, TAKII K. A faster branch-and-bound algorithm for the block relocation problem [J]. IEEE Transactions on Automation Science and Engineering, 2016, 13(1): 181.

[10] TRICOIRE F, SCAGNETTI J, BEHAM A. New insights on the block relocation problem [J]. Computers and Operations Research, 2018, 89: 127.

[11] FORSTER F, BORTFELDT A. A tree search procedure for the container relocation problem[J]. Computers and Operations Research, 2012, 39(2): 299.

[12] TANAKA S, MIZUNO F. An exact algorithm for the unrestricted block relocation problem [J]. Computers and Operations Research, 2018, 95: 12.

[13] TING C J, WU K C. Optimizing container relocation operations at container yards with beam search [J]. Transportation Research Part E: Logistics and Transportation Review, 2017, 103: 17.

[14] BACCI T, MATTIA S, VENTURA P. The bounded beam search algorithm for the block relocation problem [J]. Computers and Operations Research, 2019, 103: 252.

[15] FEILLET D, PARRAGH S N, TRICOIRE F. A local-search based heuristic for the unrestricted block relocation problem[J]. Computers and Operations Research, 2019, 108: 44.

[16] BENGIO Y, LODI A, PROUVOST A. Machine learning for combinatorial optimization: a methodological tour d'horizon [J]. European Journal of Operational Research, 2021, 290(2): 405.

[17] RUVOLO P, FASEL I, MOVELLAN J. Optimization on a budget: a reinforcement learning approach[C]//22nd Annual Conference on Neural Information Processing Systems, NIPS 2008.Vancouver:Curran Associates Inc, 2009: 1385 - 1392.

[18] ZENG Y, XU K, YIN Q, *et al*. Inverse reinforcement learning based human behavior modeling for goal recognition in dynamic local network interdiction [J]. Indian Journal of Medical Research, 2016, 120(3): 151.

[19] 陈希亮,曹雷,何明,等.深度逆向强化学习研究综述[J].计算机工程与应用, 2018, 54(5):24.
CHEN Xiliang, CAO Lei, HE Ming , *et al*. Overview of deep inverse reinforcement learning[J]. Computer Engineering and Applications, 2018, 54(5): 24.

[20] LIN J L, HWANG K S, SHI H, *et al*. An ensemble method for inverse reinforcement learning [J]. Information Sciences, 2020, 512: 518.

[21] ABBEEL P, NG A Y. Apprenticeship learning via inverse reinforcement learning [C]//Proceedings, Twenty-First International Conference on Machine Learning, ICML 2004. Banff:Association for Computing Machinery, 2004: 1 - 8.

[22] 杨放青,王超,姜滨,等.舰载机出动回收调度策略生成方法[J].北京理工大学学报, 2018, 38(10):1030.
YANG Fangqing, WANG Chao, JIANG Bin, *et al*. A method of policy automated generation for carrier aircraft sortie and recovery scheduling[J]. Transactions of Beijing Institute of Technology, 2018, 38(10): 1030.

[23] KIM K H, HONG G P. A heuristic rule for relocating blocks [J]. Computers and Operations Research, 2006, 33(4): 940.

[24] 徐亚,陈秋双,龙磊,等.集装箱倒箱问题的启发式算法研究[J].系统仿真学报, 2008, 20(14):3666.
XU Ya, CHEN Qiushuang, LONG Lei, *et al*. Heuristics for container relocation problem[J]. Journal of System Simulation, 2008, 20(14):3666.

[25] 游鑫梦,梁承姬,张悦.进口集装箱堆存和翻箱策略两阶段规划模型[J].上海海事大学学报, 2020, 41(4): 1.
YOU Xinmeng, LIANG Chengji, ZHANG Yue. Two-stage programming model for storage and re-handling strategies of import containers[J]. Journal of Shanghai Maritime University, 2020, 41(4): 1.