控制与决策
Control and Decision

大量需求点下基于深度Q学习的受损路网抢修队调度

张国富, 常加远, 苏兆品, 沈宇锋

引用本文:
张国富,常加远,苏兆品,沈宇锋. 大量需求点下基于深度Q学习的受损路网抢修队调度[J]. 控制与决策, 2022, 37(12): 3267–3277.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2021.0121

大量需求点下基于Q学习的受损路网抢修队调度

张国富[1,2,3,4†]，常加远[1]，苏兆品[1,2,3,4]，沈宇锋[1]

(1. 合肥工业大学 计算机与信息学院，合肥 230601；2. 合肥工业大学 大数据知识工程教育部重点实验室，合肥 230601；3. 合肥工业大学 智能互联系统安徽省实验室，合肥 230009；4. 合肥工业大学 工业安全应急技术安徽省重点实验室，合肥 230601)

摘 要: 受损路网抢修是重特大自然灾害发生后开展应急处置和救援的一个基本前提,主要研究如何对道路抢修队进行合理的调度以快速恢复路网畅通、保障救援队伍和应急物资从出救点及时输送到各需求点. 鉴于已有研究在面向大量需求点时往往很难给出有效的调度策略,首先基于路网模型和马尔科夫决策过程分析抢修队修复受损路网的关键因素,并设计一种双反馈回报函数;然后基于深度Q学习求解抢修队的最优调度策略;最后通过对比实验结果表明,在大量需求点环境下,所提出方法具有较好的稳定性和可靠性,兼顾受损路网的修复效率和运输效率,能够以更少的修复代价令所有需求点可达,为灾后复杂应急场景下的受损路网抢修提供有益的尝试.

关键词: 应急处置和救援；路网抢修；大量需求点；抢修队调度；双反馈回报函数；深度Q学习

中图分类号: TP181 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2021.0121

引用格式: 张国富, 常加远, 苏兆品, 等. 大量需求点下基于深度Q学习的受损路网抢修队调度 [J]. 控制与决策, 2022, 37(12): 3267-3277.

Repair crew scheduling for damaged road network with enormous demand points using deep Q-learning

ZHANG Guo-fu[1,2,3,4†], *CHANG Jia-yuan*[1], *SU Zhao-pin*[1,2,3,4], *SHEN Yu-feng*[1]

(1. School of Computer Science and Information Engineering，Hefei University of Technology，Hefei 230601，China；2. Key Laboratory of Knowledge Engineering with Big Data of Ministry of Education，Hefei University of Technology，Hefei 230601，China；3. Intelligent Interconnected Systems Laboratory of Anhui Province，Hefei University of Technology，Hefei 230009，China；4. Anhui Province Key Laboratory of Industry Safety and Emergency Technology，Hefei University of Technology，Hefei 230601，China)

Abstract: Repairing the damaged road network, which mainly focuses on how to reasonably schedule the repair crew to quickly unblock the road network and ensure that rescue teams and emergency resources in the source node can be delivered to different demand nodes in time, is a basic premise for emergency disposal and rescue after the occurrence of extraordinarily serious natural disasters. However, it is difficult for the existing methods to find a feasible scheduling strategy under enormous demand nodes. Therefor the key factors of repairing the damaged road network are first analyzed according to the road network model and the Markov decision-making process, based on which a double-feedback reward function is designed. Then, the deep Q-learning is utilized to solve the optimal scheduling strategy of the repair crew. Finally, comprehensive experimental studies show that for the damaged road network with enormous demand points, the proposed method has high stability and reliability, can achieve a good balance between the repair efficiency and the transportation efficiency, and can make all the demand points achievable with less repair cost, which may provide a useful attempt to repair the damaged road network in complex emergency scenarios of post-disaster.

Keywords: emergency disposal and rescue；road network repairs；enormous demand nodes；repair crew scheduling；double-feedback reward function；deep Q-learning

收稿日期: 2021-01-21；**录用日期:** 2021-09-10.

基金项目: 安徽省重点研究与开发计划项目 (202004d07020011, 202104d07020001)；中国工程院战略咨询重点项目 (2020-XZ-3)；教育部人文社会科学研究青年基金项目 (19YJC870021)；广东省类脑智能计算重点实验室开放课题项目 (GBL202117)；中央高校基本科研业务费专项资金项目 (PA2020GDKC0015, PA2021GDSK0073, PA2021GDSK0074).

责任编委: 阳春华.

†通讯作者. E-mail: zgf@hfut.edu.cn.

0 引 言

以“大智移云”为特征的新一代信息技术的快速发展,给应急管理的信息化、智能化和科学化提供了新的机遇和挑战. 为此,《“十四五”国家应急体系规划》明确指出,要利用大数据、人工智能、机器学习等新一代信息技术提高我国重特大自然灾害风险感知、监测预警、协同救援和应急处置的能力,为推进应急管理体系和能力现代化建设提供科技支撑,实现大国智慧应急. 其中,对受损路网进行合理的修复以快速恢复路网畅通、打通救援生命线、保障救援队伍和应急物资能够及时输送到各需求点是重特大自然灾害发生后开展应急处置和救援的一个基本前提. 旨在利用智能决策理论和信息技术制定道路抢修队的调度策略,确定按照什么修复顺序、修复哪些受损路段可以最小化路网的修复时间、最大化路网的运输效率,即以最小的修复代价尽可能确保出救点到每个需求点的运输路径最短,这对迅速实施应急救援、最大限度降低灾害损失具有重要的现实意义.

在早期的研究中,道路抢修被构建成一个面向多种应急物资的整数时空网络流模型. 基于此,文献[1-2]设计了一种启发式算法将该模型划分成若干子问题,每个子问题按顺序分别采用CPLEX进行求解,其优化的目的是最小化修复路段带来的短期运营成本. Yan等[3]引入蚁群优化(ant colony optimization, ACO)来最小化整个路网的修复时间. 陈钢铁等[4]同时考虑整个路网的修复时间和应急物资运输时间的最小化. 上述这些时序决策模型虽然可以模拟道路的抢修过程,但建模和求解都非常复杂和困难.

近期的研究大都将实际的受损路网进行简化,去掉中间不必要的过渡节点,并以出救点与需求点之间以及各需求点之间的最短路径作为边构造无向图. 基于此,Liberatore等[5]提出一种考虑修复成本、运输时间、安全性和可靠性的层次模型,并使用分层法按照目标的偏好顺序进行优化. Tuzun等[6]提出一种启发式方法以最大化整个路网的连通性和最小化修复时间. Akbari等[7]基于整数规划和启发式的混合算法最小化路网修复时间或最大化路网修复后的总效用. Iloglu等[8]采用拉格朗日松弛与次梯度算法最小化时间约束下出救点到各需求点之间的累积加权距离. Moreno等[9]基于分支定界和Benders分解最小化各需求点不可达的累计加权时间. 在其后续研究中,Moreno等[10]从多抢修队角度构建了3种考虑调度决策和道路抢修队同步的混合整数规划模型. 不过,上述工作为了简化问题的求解,大都将道路抢修队的调度和路线分开进行考虑,这种分层序列优化方法对优化的各目标具有典型的偏好,限制了算法对解空间的探索,在一定程度上影响了解的质量. 为此,Maya等[11]基于动态规划和贪婪策略最大化路网中需求点的可达性. Kim等[12]采用ACO最小化需求点的总损失和路网修复时间. Shin等[13]基于ACO最小化路网修复后从出救点到各需求点的最大运输时间. 苏兆品等[14]将道路抢修队看作一个智能体,基于马尔科夫决策过程模拟抢修队的修复活动,设计了一种基于Q学习(basic Q-learning, BQL)的道路抢修队调度算法,通过道路抢修队自身的不断学习积累同时给出最优调度策略和路线,但是其算法在整个受损路段集合中选择动作,当路段受损率较大时容易选取到不可达的受损路段,从而导致算法失效. 为此,张国富等[15]在苏兆品等[14]的工作基础上提出一种改进的Q学习策略(improved Q-learning, IQL),通过对道路抢修队的动作集进行裁减,规定道路抢修队在决策时只能从当前可达的未修复受损路段集合中选择下一个动作,进一步提升了算法的稳定性.

需要指出的是,虽然上述已有工作在其构建的应用场景中具有一定的合理性,但仍存在以下问题: 1) 在衡量道路抢修队的修复性能时,没有很好地分析受损路网修复问题的关键特征量,大都着眼于路网中的受损边,所设计的评价指标、目标函数和约束条件等异常复杂,导致算法计算和求解十分困难. 2) 已有工作均只考虑少量需求点的情形,无法适应地震、洪水等重特大自然灾害的应急场景. 一个严峻的事实是,我国幅员辽阔,地理气候条件异常复杂,地震、洪水灾害频繁发生,是世界上地震、洪水灾害损失最严重的少数国家之一. 例如,2008年的5·12汶川大地震,其重灾区涉及了6个市州、88个县市区、1 204个乡镇. 已有方法在处理这些大量需求点的应急场景时往往计算和存储负担巨大,导致算法求解困难甚至失效,算法的稳定性和可靠性难以满足实际应急需求.

基于上述背景,本文在总结和分析已有工作的基础上,针对大量需求点下的受损路网抢修队调度问题,首先基于路网模型和马尔科夫决策过程分析道路抢修队修复受损路网的关键性能指标,根据这些指标设计更加简单有效的双反馈回报函数;然后基于深度Q学习(deep Q-learning, DQL)[16]求解抢修队的最优调度策略;最后通过对比实验验证所提出方法的有效性.

1　路网模型

考虑如图1所示的路网模型$G = \{V, E\}$. 其中: $V = \{0, 1, \ldots, n\}$为G中所有节点序号集合,包括一个出救点“0”和$n \in N$个需求点; $E = \tilde{E} \bigcup \bar{E}$, $\tilde{E} = \{1, 2, \ldots, m\}$为$G$中$m \in N$条受损路段的集合, $\bar{E} = \{1, 2, \ldots, p\}$为$G$中$p \in N$条可通行路段集合.

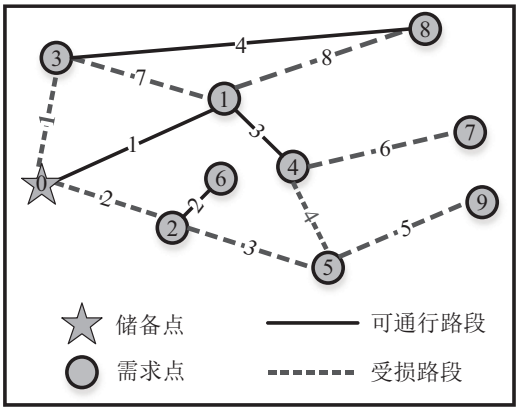


图1　受损路网

任意需求点$i \in \{1, 2, \ldots, n\}$有一个权重$I_i \in (0, 1]$,以区分各个需求点的受灾严重程度. 此外,任意$j \in \{1, 2, \ldots, m\}$和任意$k \in \{1, 2, \ldots, p\}$都有一个路段长度,分别用$a_j \in R$和$b_k \in R$表示. 道路抢修队从出救点“0”出发,行进速度为$v \in R$,修复一条受损路段$j \in \{1, 2, \ldots, m\}$的时间开销为$t_j \in R$.

在受损路网修复中,道路抢修队需要决定修复哪些受损路段以及按照什么修复顺序可以最快地使所有需求点可达,同时每个需求点到出救点的距离要尽可能短. 假设抢修队整个修复活动为有序集$H \subseteq E$,即抢修队所经历的路段集合包括经历的可通行路段和受损路段.对于每一个H,其对应的时间消耗为

$$f_1(H) = \sum_{i=1}^{n} (T_i \cdot I_i), \tag{1}$$

其中T_i为需求节点i被打通时的累积时间开销. 不妨设需求节点i被打通时抢修队经过的路段有序集为$H_i \subseteq H$,则有

$$T_i = \sum_{j \in H_i \cap \tilde{E}} \left(t_j + \frac{a_j}{v}\right) + \sum_{k \in H_i \cap \bar{E}} \frac{b_k}{v}. \tag{2}$$

H对应的出救路径长度为

$$f_2(H) = \sum_{i=1}^{n} (D_i \cdot I_i), \tag{3}$$

其中D_i为路网修复后出救点“0”到需求节点i的最短路径长度,可由最短路径算法计算得到.

基于上述考虑,优化H的目标函数[14-15]为

$$\min f(H) = \omega_1 \cdot f_1(H) + \omega_2 \cdot f_2(H). \tag{4}$$

其中$\omega_1, \omega_2 \in (0, 1)$为权重,满足$\omega_1 + \omega_2 = 1$,体现了$f_1$和$f_2$在整体应急响应目标中的偏好. 与已有工作不同的是,本文对所有节点和路段均以相应的序号和集合加以标识,在不改变目标函数的基础上,进一步简化模型表示,为方便后续决策模型和求解算法的设计打下基础.

此外,有如下结果.

命题1　可能的H数为$\sum_{x=1}^{m} \frac{m!}{(m-x)!}$.

证明　对决策起决定作用的是选择了哪些受损路段,以及这些受损路段的排列顺序. 因此,对于有序集H,相当于从受损路段集合$\tilde{E}$的m个不同元素中任取x个元素,并按照一定顺序排列起来,可能的排列数为$\tilde{E}_m^x = \frac{m!}{(m-x)!}$.又$1 \leqslant x \leqslant m$,故可能的$H$数为$\sum_{x=1}^{m} \frac{m!}{(m-x)!}$. □

命题2　$x \geqslant n - p$.

证明　修复受损路网是为了令所有需求点可达,即所有节点都能与储备点“0”连通. 在图1所示的受损路网中,根据生成树原理可知,任一生成树(包含所有$n+1$个节点的连通子图)的边数为n,因此,若想所有需求点连通,则需要修复的受损路段数x至少为$n - p$,此时该生成树包含所有p个可通行路段(如果存在的话),因此有$x \geqslant n - p$. □

2　决策模型

受损路网修复是典型的序贯决策问题,而马尔科夫决策过程是序贯决策的一种经典表达形式,通常包括状态空间S、动作空间A和回报函数R三个基本要素. 将抢修队看作一个智能体,其状态空间为$S \leftarrow V$,即受损路网中所有节点的集合,动作空间为$A \leftarrow \tilde{E}$,即受损路网中当前所有受损路段的集合.

为了优化目标函数(4),文献[14-15]均基于(4)直接设计抢修队的回报函数,导致回报函数计算异常复杂. 特别地,上述工作均基于已经连通的需求节点集合、可达需求节点到出救点的最短路径列表和已经修复的受损路段集合构建三维状态空间,当面对大量需求点时需要耗费大量的计算开销和存储开销. 需要指出的是,在强化学习中,通常只需根据拟实现的目标给予智能体合理的奖励刺激,智能体即会按照奖励的趋势自主学习,并朝着目标积极探索,没有必要完全按照目标函数设计奖励值. 为此,在简化路网模型的基础上,进一步考虑如何简化回报函数的设计.

受损路网修复过程中最核心的问题是能否快速修复一条受损路段,以及修复后能否打通更多受灾严重的需求节点. 因此,当抢修队执行一个动作(即选择

一条受损路段j进行修复),应考虑如下几个因素:

1) 抢修队修复受损路段j,新打通的需求节点数应该越多越好,用最经济的修复代价达到最大的修复效果,即

$$N = |\vec{V}|, \tag{5}$$

其中$\vec{V} \subseteq V$为新打通的需求节点集合.

2) 抢修队修复受损路段j,新打通的需求节点的受灾严重程度应该越高越好,因为受灾越严重的需求节点越应尽早得到响应,即

$$\vec{I} = \sum_{i \in \vec{V}} I_i. \tag{6}$$

3) 抢修队修复受损路段j,所耗费的修复时间t_j应该越小越好.

4) 抢修队修复受损路段j后,剩余未连通的需求节点中,受灾程度最大的需求节点i^*应该在后面的决策中得到体现,即

$$i^* = \arg\max_{i \in \ddot{V}} I_i, \tag{7}$$

其中$\ddot{V} \subseteq V$为还未连通的需求节点集合.

5) 抢修队修复受损路段j后,剩余未修复的受损路段中,修复时间最短的受损路段j^*也应该在后面的决策中加以考虑,即

$$j^* = \arg\min_{j \in \ddot{E}} t_j, \tag{8}$$

其中$\ddot{E} \subseteq \tilde{E}$为剩下的受损路段集合.

对这5个因素同等考虑,将抢修队修复受损路段j的回报函数设计为

$$R \leftarrow \begin{cases} -I_{i^*} - \dfrac{1}{t_{j^*}}, \ N = 0; \\ \dfrac{\vec{I}}{N} + \dfrac{1}{t_j}, \ N \geqslant 1. \end{cases} \tag{9}$$

具体而言,抢修队修复受损路段j后,如果没有新的需求节点被打通(即$N = 0$),则该动作并不能给整个路网的修复带来显著效果,此时抢修队将得到一定的惩罚(即负反馈),这个惩罚由当前路网中未连通且受灾程度最大的需求节点i^*和修复时间最短的受损路段j^*共同给出. i^*的受损程度I_{i^*}越严重,j^*的修复时间t_{j^*}越短,所受到的惩罚也越大. 相反的,如果有新的需求节点被打通(即$N \geqslant 1$),则该动作给整个路网的修复带来了明显的效果,此时抢修队应得到一定的奖励(即正反馈),且新打通的需求节点数N越多总受损程度$\vec{I}$越严重,修复时间t_j越短所获得的奖励值越大,从而刺激抢修队的决策.

需要指出的是,为了解决不同数量级带来的偏差,采用修复时间倒数归一化修复时间带来的回报. 当多个需求点被同时打通时,简单的受灾程度累加不能很好地区分受灾程度不同带来的影响,总是希望新打通的需求点受损程度都比较严重才好,这样可以优先响应重灾区. 因此,为了平滑多个新打通需求点受灾程度不同对策略的影响,同时解决不同数量级带来的偏差,利用$\vec{I}/N$归一化受灾程度总和以及新打通需求点数带来的回报,同时区分不同受灾程度的影响,以实现受损程度均比较严重则回报相对较大、受灾程度差异较大则回报相对较小. 这样处理的好处是,无论N取何值,得到的回报增减幅度都在一个数量级,不会出现明显的偏向.

3 基于DQL的最优调度策略求解

本文引入DQL[16]求解抢修队的最优调度策略. DQL的一个基本实现方式是利用深度Q网络(deep Q-network, DQN)[16]通过引入神经网络结构有效解决状态空间过大的问题. 当前,DQN有3种主流的改进版本:DDQN(double deep Q-networks)[17]、PER (prioritized experience replay)[18]和Dueling DQN[19]. 为了更加清晰地表明DQL算法如何求解抢修队最优调度策略,下面将在DQN的基础上详细介绍DDQN、PER和Dueling DQN中的一些关键细节,并给出DQL算法的整体框架和具体步骤.

3.1 DQN

DQN[16]的基本思路来源于Q学习,不同的是,其Q值不是直接通过状态值和动作计算,而是通过一个Q网络(神经网络)计算,并使用经验实现Q值的更新,具体流程如算法1所示

算法1 DQN算法.

输入: 训练周期数T,状态集S,动作集A,Q网络参数θ,批量梯度下降的样本数x.

1. 随机初始化参数θ,基于θ初始化所有的S和A对应的Q值,清空经验回放池C;

2. for $t := 1$ to T

1) 随机选择一个初始状态s;

2) 将s输入到Q网络中得到s下所有动作对应的当前Q值,利用ε贪婪策略在当前Q值中选择一个动作a;

3) 执行动作a得到新状态s',根据回报函数计算奖励r,并将此次状态转移$\{s, a, r, s'\}$存储到C中;

4) 从C中抽取x个状态转移样本$\{s_j, a_j, r_j, s'_j\}, j = 1, 2, \ldots, x$,基于式(10)计算每个样本的目标$Q$值$y_j$;

5) 基于式(11)通过神经网络的梯度反向传播更新Q网络参数θ;

6) 如果s'不是终止状态,则令$s \leftarrow s'$,转2);

end for

为了训练Q网络参数θ,从经验回放池C中抽取$x \in N$个状态转移样本$\{s_j, a_j, r_j, s'_j\}, j = 1, 2, \ldots, x$,其中每个样本表示状态$s_j$执行动作$a_j$得到新状态$s'_j$,并获得奖励$r_j$. 如果$s'_j$是终止状态,则状态转移$\{s_j, a_j, r_j, s'_j\}$对应的$Q$值$y_j = r_j$. 如果$s'_j$不是终止状态,则将$s'_j$输入到$Q$网络中,得到$s'_j$下所有动作对应的$Q$值,然后在$Q$网络中找出最大$Q$值对应的动作$a'$,再利用$a'$计算$y_j$,即

$$y_j \leftarrow \begin{cases} r_j,\ s'_j\text{是终止状态;} \\ r_j + \gamma \arg\max\limits_{a'} Q(s'_j, a'; \theta),\ \text{otherwise.} \end{cases} \tag{10}$$

其中:γ为衰减因子,θ为Q网络参数. 得到x个样本的目标Q值后,使用均方差损失函数

$$L(\theta) \leftarrow \frac{1}{x}\sum_{j=1}^{x}(y_j - Q(s_j, a_j; \theta))^2, \tag{11}$$

通过神经网络的梯度反向传播更新Q网络的参数θ.

3.2 DDQN

DDQN是在DQN的基础上通过构造两个结构相同但参数不同的神经网络(主Q网络和目标$\vec{Q}$网络)来解耦目标Q值动作的选择和计算,以此消除DQN中对动作Q值的过度估计问题[17].

为了计算每个状态转移$\{s_j, a_j, r_j, s'_j\}$的目标Q值y_j,首先将s'_j同时输入主Q网络和目标$\vec{Q}$网络中,分别得到s'_j下所有动作对应的Q值;然后在主Q网络中找出最大Q值对应的动作a',利用a'在目标$\vec{Q}$网络中找到其对应的Q值用于计算y_j,即

$$y_j \leftarrow r_j + \gamma\vec{Q}(s'_j, \arg\max_{a'} Q(s'_j, a'; \theta); \vec{\theta}), \tag{12}$$

其中$\vec{\theta}$为目标$\vec{Q}$网络参数. 随后主Q网络参数θ的更新与DQN一致. 此外,每隔一定的训练周期后,需要更新目标Q网络参数实现递进学习$\vec{\theta} \leftarrow \theta$.

3.3 PER

在DDQN中,从经验回放池里抽取状态转移样本时,所有样本均具有相同的采样率,忽视不同样本的重要性是不对的. 为此,在PER中,每个样本$\{s_j, a_j, r_j, s'_j\}$都有一个时间差分误差,即目标$\vec{Q}$网络计算的目标Q值与主Q网络计算的当前Q值之间的差距[18],有

$$\delta_j \leftarrow y_j - Q(s_j, a_j; \theta). \tag{13}$$

则$\{s_j, a_j, r_j, s'_j\}$的采样概率为

$$P_j \leftarrow \frac{{p_j}^{\alpha}}{\sum\limits_{j'} {p_{j'}}^{\alpha}}. \tag{14}$$

其中:α表示使用多少优先性;

$$p_j \leftarrow |\delta_j| + \rho \tag{15}$$

表示样本的优先级,影响它被采样的概率;ρ用以防止未采样的状态误差为0. PER通常使用SumTree二叉树结构设计经验回放池. 所有状态转移样本只保存在最下面的叶子节点上面,一个叶子节点一个样本,叶子节点除了保存状态转移数据$\{s_j, a_j, r_j, s'_j\}$,还要保存该样本的优先级p_j. 内部节点不保存状态转移数据,只保存自己孩子节点的优先级之和. 通过这种存储结构容易实现式(13)中优先级较高的样本更容易被采样. 除了抽样方式的改进,PER还考虑了优先回放引入的误差,并采用重要性权重进行修正,有

$$w_j \leftarrow (|C|P_j)^{-\beta}. \tag{16}$$

其中:$|C|$为经验池中的样本总数,β为权重系数. 为了进一步确保稳定性,对每个w_j进行归一化,有

$$w_j \leftarrow \frac{w_j}{\max\limits_{j'} w_{j'}}. \tag{17}$$

利用归一化后的权重改进均方差损失函数

$$L(\theta) \leftarrow \frac{1}{x}\sum_{j=1}^{x} w_j(y_j - Q(s_j, a_j; \theta))^2. \tag{18}$$

需要注意的是,在PER中,主Q网络参数θ梯度更新后需要基于式(12)重新计算所有样本的δ_j,并更新SumTree中的p_j,以用于下一次的样本抽样.

3.4 Dueling DQN

Dueling DQN在神经网络结构后面加了两个子网络支路:价值函数网络和优势函数网络. 最终Q网络的输出由价值函数网络的输出和优势函数网络的输出线性组合得到[19],即

$$Q(S, A; \theta, \phi, \varphi) \leftarrow \mathcal{V}(S; \theta, \phi) + \mathcal{A}(S, A; \theta, \varphi). \tag{19}$$

其中:ϕ和φ分别为两个子网络参数;价值函数$\mathcal{V}(S; \theta, \phi)$仅与状态集$S$有关,表示处于当前状态下长远考虑判断;优势函数$\mathcal{A}(S, A; \theta, \varphi)$同时与状态集$S$和动作集$A$有关,表示当前状态下选择不同动作的优劣判断.

3.5 DQL算法框架

参考已有工作[19-20],将DDQN、PER与Dueling DQN进行组合来设计求解抢修队最优调度策略的DQL算法,基本框架如图2所示.

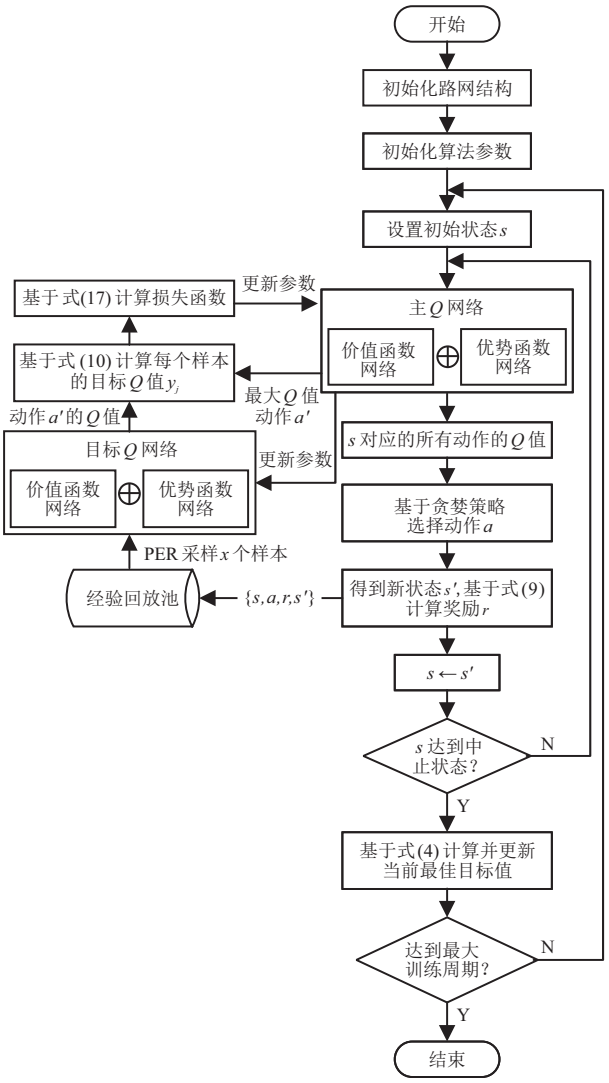


图 2 求解抢修队最优调度策略的DQL算法流程

DQL算法的基本思路为:在每个训练周期,当抢修队遍历的状态空间$S = V$时达到终止状态,根据完整的状态序列基于式(4)计算并更新当前最佳目标值. 基于该思想,求解抢修队最优调度策略的具体流程描述如下.

step 1: 初始化路网结构,包括各应急点受灾程度I_i、各路段长度a_j、b_k,受损路段修复时间t_j,抢修队行进速度v.

step 2: 初始化算法参数:最大训练周期数T,状态集S,动作集A,主Q网络参数θ'(包含θ、ϕ、φ),目标$\vec{Q}$网络参数$\vec{\theta}'$(包含$\vec{\theta}$、$\vec{\phi}$、$\vec{\varphi}$),$\vec{\theta}'$更新频率τ,批量梯度下降样本数x,衰减因子γ,探索率ε,清空经验回放池C,当前训练周期$t \leftarrow 1$.

step 3: 设置初始状态$s \leftarrow 0$,即抢修队从出救点“0”出发.

step 4: 在主Q网络中基于式(18)得到当前状态s下所有可达动作(即当前节点可到达的受损路段)对应的Q值,并根据ε贪婪策略选择一个动作a.

step 5: 根据选定的动作a得到下一个状态s',基于式(9)计算奖励r,并将此次状态转移$\{s,a,r,s'\}$存储到经验回放池C中.

step 6: 依据式(13)从C中采样x个状态转移样本,基于式(10)计算每个样本的目标Q值y_j.

step 7: 基于式(16)和(17)对主Q网络参数θ'进行梯度更新.

step 8: 如果$t\,\%\,\tau = 0$,则更新目标Q网络参数,$\vec{\theta}' \leftarrow \theta'$.

step 9: 如果s'不是终止状态,则$s \leftarrow s'$,转至step 4;否则,根据遍历的状态序列基于式(4)计算并更新当前最佳目标值.

step 10: $t \leftarrow t+1$. 如果$t \leqslant T$,则转至step 3;否则,算法结束,输出当前最佳的目标值及其遍历状态序列对应的路径.

4 仿真实验结果与分析

为了验证所提出模型和DQL算法的有效性,本节首先给出实验环境和参数设置,然后对比分析所设计回报函数的优劣,最后将DQL算法与已有ACO算法[12,13]、BQL算法[14]和IQL算法[15]进行深入对比分析.

4.1 实验环境与参数设置

参考已有工作,为了全面评估所提出算法对不同路网规模下不同受损程度的适应性和鲁棒性,设计逐渐增加的n和E,分别为$n = 50$、$|E| = 75$,$n = 100$、$|E| = 150$,$n = 200$、$|E| = 300$共计3种规模. 此外,路段受损率$|\tilde{E}|/|E| = \{15\,\%, 30\,\%, 50\,\%\}$,即路段受损越来越严重. 需要指出的是,本文针对的是大量需求点下的受损路网修复问题,因此,这里n和E的值要比已有工作大得多. 而且为了减少受灾严重程度、受损路段长度、受损路段修复时间等参数处于区间极端值的概率,采用正态分布随机数,以方便进行统计分析. 具体来说,每个需求点的受灾严重程度I_i在$(0,1]$之间按照正态分布随机生成;受损路段的长度a_j和可通行路段的长度b_k均在$[1,10]$之间按照正态分布随机生成;受损路段的修复时间t_j在$[1,10]$之间按照正态分布随机生成. 抢修队行进速度$v = 1$. 根据上述3种不同的需求点数和路段受损率,在路网结构生成器中随机产生9种不同规模和受损状况的路网实例用于测试.

DQL、BQL和IQL三种算法共同的参数如下:探索率$\varepsilon = 0.1$,衰减因子$\gamma = 0.9$,最大训练周期数$T = \{300, 700, 1\,500; 700, 1\,500, 3\,000; 1\,500, 3\,000, 6\,000\}$,即在3种不同的需求点数下,随着路段受损率的增

加,训练周期数也相应增加. 对于BQL和IQL算法,计算目标Q值的学习程度参数为0.4. 对于DQL算法,$\alpha = 0.6, \rho = 0.01, \beta = 0.4, \pi = 5, |C| = 2\,000$,梯度下降步长为0.001,对应不同的需求点数$x = \{16; 32; 32\}, \tau = \{50; 150; 500\}$. 对于ACO算法,蚂蚁数为80,总迭代次数为300,信息素挥发率为0.5,信息素调控因子为0.003,路程调控因子为0.02.

各测试实例均在Intel Xeon CPU 2.20 GHz、32 GB内存、Windows Server 2012操作系统的个人计算机上独立运行30次,根据30次不同结果进行统计分析.

4.2　不同回报函数的对比

修复当前受损路段后,文献[14-15]认为,即使没有新需求点被打通,也应该考虑路网中可达需求点到储备点的最短路径长度减小而给予一定的奖励,并依此设计了一种正反馈回报函数. 本文考虑没有新需求点被打通时应给予一定的惩罚,设计了一种正负双反馈回报函数.

图3给出了两种回报函数在每个训练周期上的累积回报值. 为了消除不同数量级引起的误差,对累积回报值进行最大最小归一化处理. 由图3可见,在9种路网中,双反馈回报函数在达到一定训练周期时,累积回报值基本趋于平缓,只是在$n = 200, |\tilde{E}|/|E| = 50\%$的路网中有非常细微的波动,而正反馈回报函数得到的累积回报值一直处于震荡中. 此外,随着训练周期的增加,双反馈的累积回报值呈逐渐上升趋势,表明逐步收敛于最佳调度策略,而正反馈在6种路网上的累积回报值却震荡且逐渐下降,表明正反馈回报函数在后期的刺激作用逐渐衰弱,收敛性不足. 还很容易看到,双反馈的累积回报值不受需求点数和路段受损率的影响,而正反馈的累积回报值在不同需求点数和路段受损率下差异很大,非常不稳定.

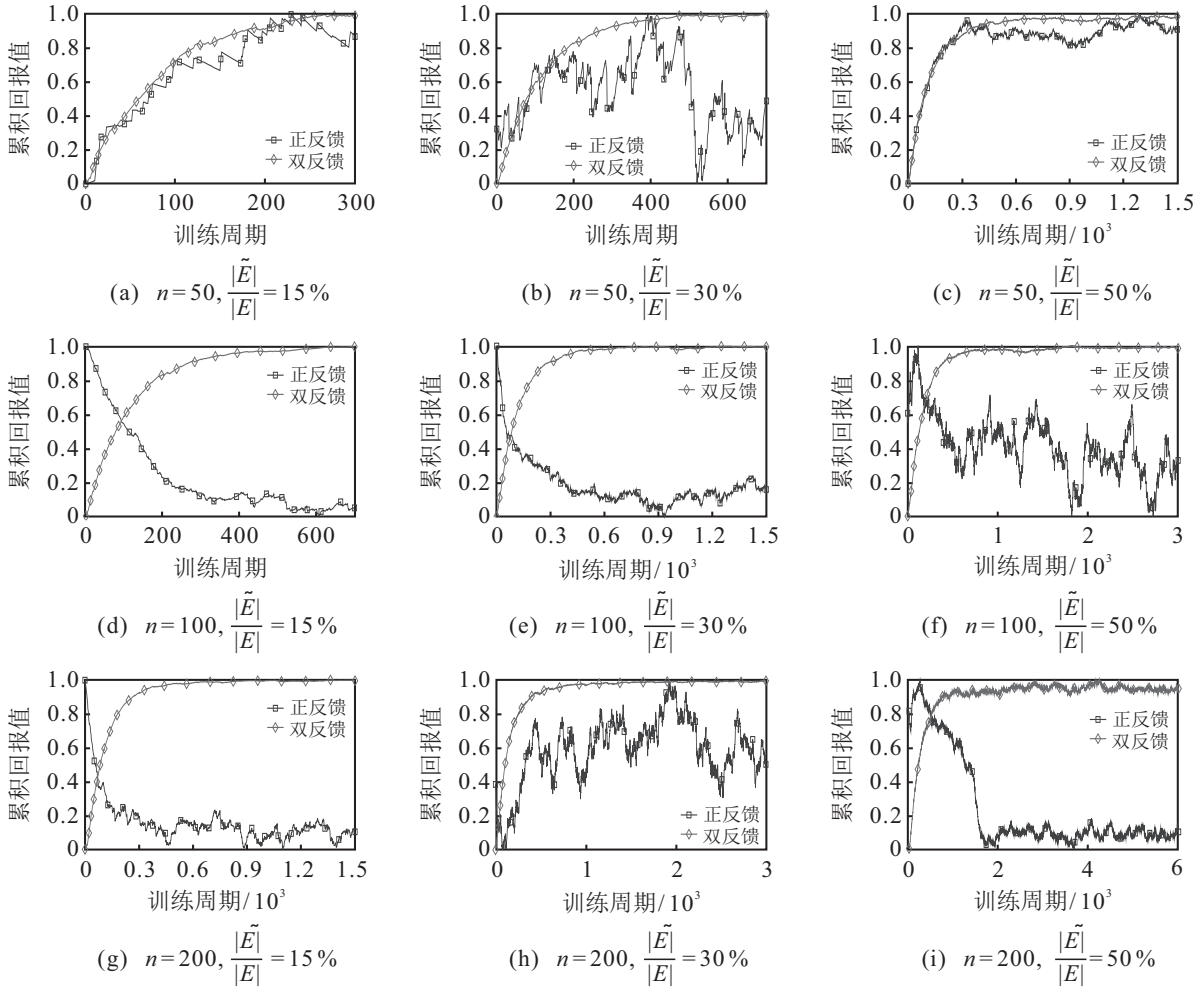


图3　不同回报函数下的累积回报值

表1给出了不同回报函数下抢修队最优调度策略对应的总通行距离(均值±标准差)和目标函数值(均值±标准差),其中较优值加粗显示. 可以看出,与正反馈相比,在9种路网中,双反馈的修复代价更小,可以使抢修队节省更多的时间,所得调度策略也明显优于正反馈,其目标函数值仅在$n = 200, \dfrac{|\tilde{E}|}{|E|} = 50\%$的路网中稍低于正反馈.

表1 不同回报函数下的总通行距离和目标函数值

	n	$\lvert\tilde{E}\rvert/\lvert E\rvert=15\%$		$\lvert\tilde{E}\rvert/\lvert E\rvert=30\%$		$\lvert\tilde{E}\rvert/\lvert E\rvert=50\%$	
		正反馈	双反馈	正反馈	双反馈	正反馈	双反馈
总通行距离	50	44.29±1.59	**43.42±0.63**	157.41±17.38	**135.82 ±8.06**	501.85±48.35	**459.01±71.42**
	100	103.23±17.24	**70.11±5.59**	886.13±60.95	**613.78±40.16**	1 565.1±124.22	**1 207.54±88.93**
	200	249.92±43.04	**148.53±6.02**	1 278.92±114.32	**1 243.32±58.66**	4 808.72±102.02	**4 762.3±452.15**
目标函数值	50	10.48±4.73	**8.58±2.26**	56.56±6.39	**49.12±5.21**	252.04±16.83	**209.95±18.07**
	100	49.65±7.92	**40.52±1.01**	308.22±25.27	**227.63±19.64**	617.91±24.46	**502.63±14.6**
	200	104.64±14.3	**69.38±2.34**	433.59±27.94	**417.07±47.04**	**1 926.64±59.42**	2 015.57± 115.02

表2 不同算法得到的总通行距离和目标函数值(均值±标准差)

	n	$\lvert\tilde{E}\rvert/\lvert E\rvert=15\%$				$\lvert\tilde{E}\rvert/\lvert E\rvert=30\%$			
		BQL	IQL	ACO	DQL	BQL	IQL	ACO	DQL
总通行距离	50	44.15±3.96	44.29±1.59	39.89	**37.54±1.85**	153.49±20.81	157.41±17.38	136.93	**127.97±13.44**
	100	98.55±15.62	103.23±17.24	65.81	**63.29±4.53**	930.14±105.58	886.13±60.95	795.36	**675.1±36.88**
	200	348.81±53.42	249.92±43.04	202.72	**180.97±13.15**	1 341.86±138.40	1 278.92±114.32	1 349.83	**1 152.4±64.23**
目标函数值	50	10.87±5.01	10.48±4.73	8.17	8.17±0	**48.45±8.06**	56.56±6.39	49.17	49.26±3.66
	100	48.14±6.68	40.34±0	40.34	**36.31±2.24**	305.1±16.66	308.22±25.27	**259.58**	270.06±14.48
	200	138.7±25.1	104.64±14.3	84.62	**68.13±7.25**	430.70±51.77	433.59±27.94	431.02	**387.02±21.66**

	n	$\lvert\tilde{E}\rvert/\lvert E\rvert=50\%$			
		BQL	IQL	ACO	DQL
总通行距离	50	–	501.85±48.35	599.33	**500.56±31.14**
	100	–	1565.1±124.22	1607.43	**1 080.33±39.73**
	200	–	4 808.72±218.57	5 591.77	**4 365.24±226.07**
目标函数值	50	–	252.04±16.83	236.3	**223.01±12.75**
	100	–	617.91±24.46	588.85	**459.56±17.51**
	200	–	1 926.64±59.42	2 046.38	**1 620.02±83.94**

上述实验结果表明,与单纯的正反馈相比,双反馈回报函数具有更强的可感知能力,能够更加有效地刺激抢修队的策略探索,更适合本文的抢修队调度问题.

4.3 不同算法的对比

第2个实验基于9种不同路网,对比分析ACO[12-13]、BQL[14]、IQL[15]和本文DQL的性能.

表2给出了4种算法在不同路网中所得调度策略的通行距离(均值±标准差)和目标函数值(均值±标准差). 其中,由于ACO算法过于耗时,因此对于每种路网,将ACO算法独立运行3次并取3次运行结果的均值. 从表2可见,IQL在9种路网中所得调度策略的修复代价均最小,在6种路网中得到的目标函数值更优;ACO、BQL和IQL均较DQL耗费更多的修复代价,且ACO和BQL仅在一种网路中获得了更优的目标函数值,而IQL在所有9种路网中都表现不佳;特别地,BQL在$\lvert\tilde{E}\rvert/\lvert E\rvert = 50\%$时算法失效,找不到有效解.

图4给出了不同需求点数和不同受损率下4种算法所得最佳调度方案在修复过程中的需求点连通

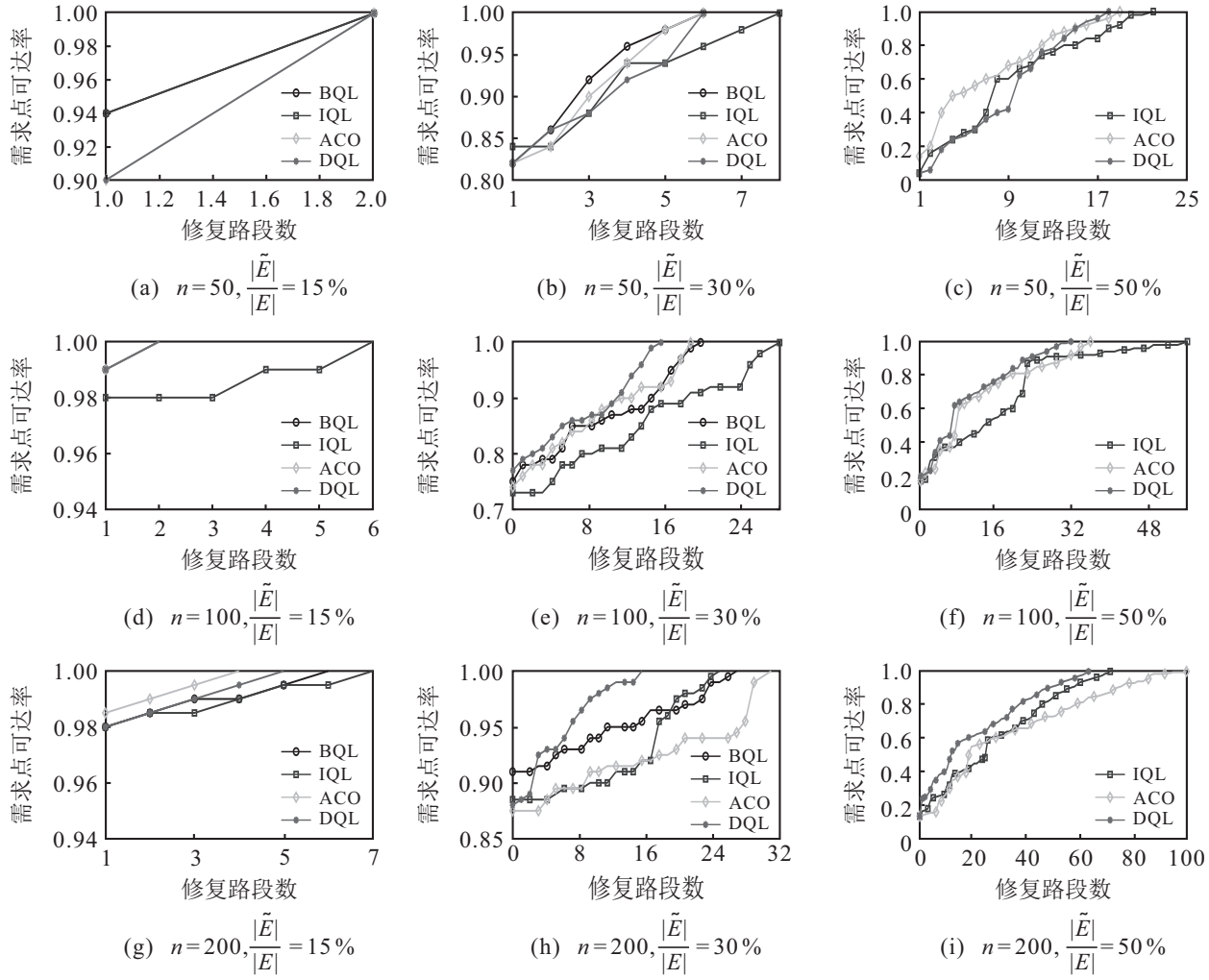
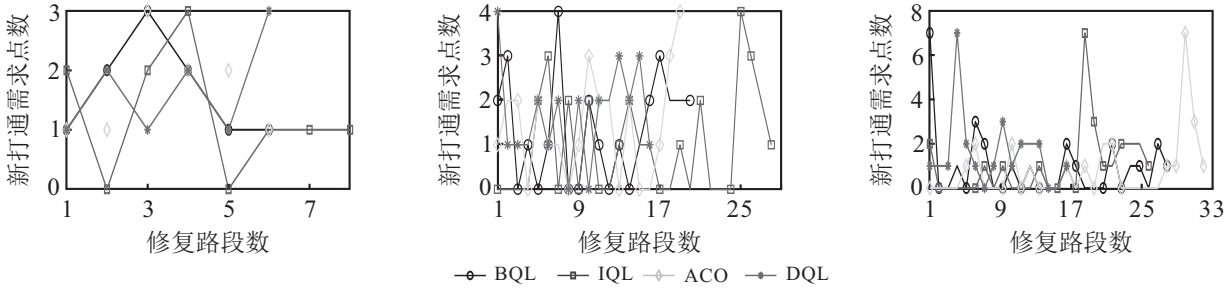


图 4　不同算法得到的需求点连通率变化

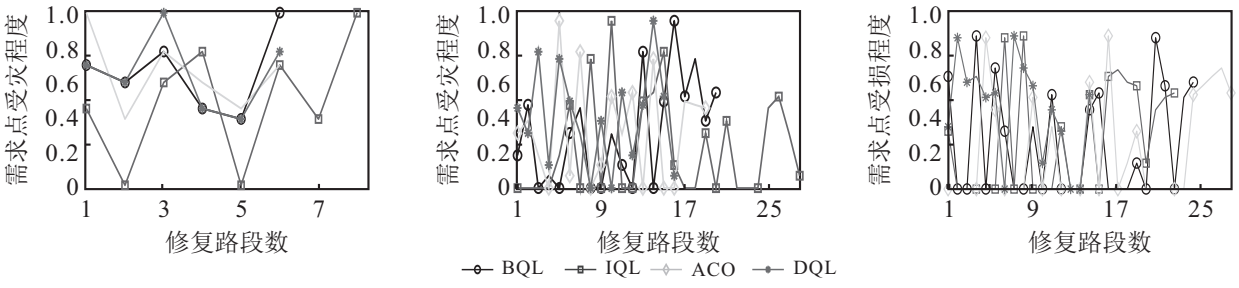
率变化情况,即每修复一条受损路段后,整个路网中与出救点连通的需求点占比. 可以看出,在9种路网环境中,DQL算法的需求点连通率相比ACO、BQL和IQL增长趋势更快,能够在修复较少受损路段时便能使需求点连通率快速达到100%. 具体而言,当$|\tilde{E}|/|E| = 15\%$时,DQL与ACO、BQL与IQL之间的差距不是很大,4种算法均只需修复几个路段即可实现所有需求点可达,其中DQL、ACO略好于BQL、IQL;当$|\tilde{E}|/|E| = 30\%$时,随着需求点数的增加,DQL优势明显,例如$n = 200$时,DQL仅修复16条路段即打通了所有需求点,而ACO、BQL和IQL此时的需求点连通率分别为92%、95.5%和92%;当$|\tilde{E}|/|E| = 50\%$时,随着需求点数的增加,DQL的优势更加明显,例如$n = 200$, $|\tilde{E}|/|E| = 50\%$时,DQL仅修复63条路段即打通了所有需求点,而ACO和IQL则分别需要修复99和71条路段. 此外,还可以看出,越到后期,DQL效率越高,需求点连通率上升得越快,而ACO和IQL在后期的需求点连通率增长缓慢.

图5～图7分别给出了4种算法在$|\tilde{E}|/|E| = 30\%$下的新打通需求点数、需求点平均受灾程度和受损路段修复时间的变化情况. 由图5可见,当$n = 50$时,DQL、ACO和BQL三种算法在修复一条路段时平均能打通约2个需求点,而IQL算法只能打通约1个需求点;当$n = 100$和$n = 200$时,DQL算法在修复一条路段时平均能打通约2个需求点,而BQL、IQL和ACO三种算法只能打通约1个需求点. 且随着n的增加,DQL算法在早期打通的需求点数越来越多. 由图6可以明显地看到,DQL算法可以在早期尽可能地先打通受灾程度较大的需求点,而BQL、IQL和ACO三种算法在中后期才能响应受灾程度大的需求点. 图7中,当$n = 50$时,DQL算法修复一条受损路段的平均时间要比BQL和IQL稍高,ACO算法的平均修复时间最少;当$n = 100$时,4种算法的平均修复时间非常接近;当$n = 200$时,DQL算法平均修复时间明显少于BQL、IQL和ACO三种算法. 可见,随着n的增加,DQL会倾向于优先选择修复时间少的受损路段. 上述实验结果表明,在式(9)的几个应急因素的综合作用下,DQL算法会随着n的增加尽快优先打通受灾程度高的需求点,能够更好地契合应急响应需求.



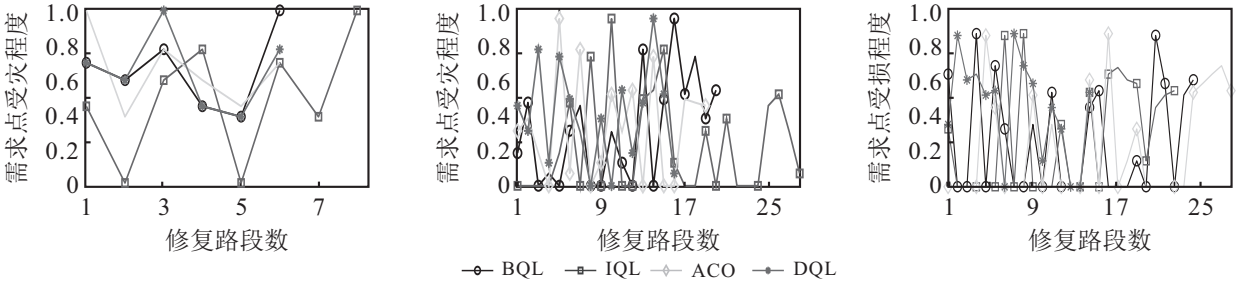
(a) $n=50$ (b) $n=100$ (c) $n=200$

图5 不同算法的新打通需求点数变化



(a) $n=50$ (b) $n=100$ (c) $n=200$

图6 不同算法的需求点平均受灾程度变化



(a) $n=50$ (b) $n=100$ (c) $n=200$

图7 不同算法的修复时间变化

5 结 论

在地震、洪水等自然灾害发生后,如何规划道路抢修队的抢修活动以快速打通生命通道、及时输送救援队伍和应急物资是灾害应急响应中的一个重要环节. 本文针对重特大自然灾害下的大量需求点复杂应急场景,首先简化了路网模型和决策模型,并设计了一种双反馈回报函数;然后基于深度Q学习求解抢修队的最优调度策略. 实验结果表明,在大量需求点环境下,所提出方法具有很好的稳定性和可靠性,能够以更小的修复代价迅速使所有需求点可达. 但是,本文只是对复杂应急场景下受损路网抢修问题的一个初步探索,未来仍有许多问题需要进一步深入研究. 首先,需要从相关单位调研和收集小、中、重大灾害历史案例的真实数据,以验证本文算法的实用性;其次,将基于连通图的生成树理论和方法详细分析本文模型的理论基础,尤其是最优解所在的区域,以挖掘有用的先验知识用于指导算法优化、提升算法效果和效率;进一步,将深入测试和分析回报函数(9)中每个因素不同优先级对策略搜索的影响.

参考文献(References)

[1] Yan S, Lin C K, Chen S Y. Optimal scheduling of logistical support for an emergency roadway repair work schedule[J]. Engineering Optimization, 2012, 44(9): 1035-1055.

[2] Yan S Y, Lin C K, Chen S Y. Logistical support scheduling under stochastic travel times given an emergency repair work schedule[J]. Computers & Industrial Engineering, 2014, 67: 20-35.

[3] Yan S Y, Shih Y L. An ant colony system-based hybrid algorithm for an emergency roadway repair time-space network flow problem[J]. Transportmetrica, 2012, 8(5): 361-386.

[4] 陈钢铁, 帅斌. 震后道路抢修和应急物资配送优化调度研究[J]. 中国安全科学学报, 2012, 22(9): 166-171. (Chen G T, Shuai B. Optimizing emergency road repair

and distribution of relief supplies after earthquake[J]. China Safety Science Journal, 2012, 22(9): 166-171.)

[5] Liberatore F, Ortuño M T, Tirado G, et al. A hierarchical compromise model for the joint optimization of recovery operations and distribution of emergency goods in humanitarian Logistics[J]. Computers & Operations Research, 2014, 42: 3-13.

[6] Tuzun Aksu D, Ozdamar L. A mathematical model for post-disaster road restoration: Enabling accessibility and evacuation[J]. Transportation Research — Part E: Logistics and Transportation Review, 2014, 61: 56-67.

[7] Akbari V, Salman F S. Multi-vehicle synchronized arc routing problem to restore post-disaster network connectivity[J]. European Journal of Operational Research, 2017, 257(2): 625-640.

[8] Iloglu S, Albert L A. An integrated network design and scheduling problem for network recovery and emergency response[J]. Operations Research Perspectives, 2018, 5: 218-231.

[9] Moreno A, Munari P, Alem D. A branch-and-Benders-cut algorithm for the crew scheduling and routing problem in road restoration[J]. European Journal of Operational Research, 2019, 275(1): 16-34.

[10] Moreno A, Alem D, Gendreau M, et al. The heterogeneous multicrew scheduling and routing problem in road restoration[J]. Transportation Research — Part B: Methodological, 2020, 141: 24-58.

[11] Maya D P A, Dolinskaya I S, Sörensen K. Network repair crew scheduling and routing for emergency relief distribution problem[J]. European Journal of Operational Research, 2016, 248(1): 272-285.

[12] Kim S, Shin Y, Lee G M, et al. Network repair crew scheduling for short-term disasters[J]. Applied Mathematical Modelling, 2018, 64: 510-523.

[13] Shin Y, Kim S, Moon I. Integrated optimal scheduling of repair crew and relief vehicle after disaster[J]. Computers & Operations Research, 2019, 105: 237-247.

[14] 苏兆品, 李沫晗, 张国富, 等. 基于Q学习的受灾路网抢修队调度问题建模与求解[J]. 自动化学报, 2020, 46(7): 1467-1478.
(Su Z P, Li M H, Zhang G F, et al. Modeling and solving the repair crew scheduling for the damaged road networks based on Q-learning[J]. Acta Automatica Sinica, 2020, 46(7): 1467-1478.)

[15] 张国富, 涂冰花, 苏兆品, 等. 一种面向严重受损路网的抢修队调度算法[J]. 控制与决策, 2021, 36(7): 1663-1671.
(Zhang G F, Tu B H, Su Z P, et al. An algorithm for repair crew scheduling on severely damaged road network[J]. Control and Decision, 2021, 36(7): 1663-1671.)

[16] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540): 529-533.

[17] van Hasselt H, Guez A, Silver D. Deep reinforcement learning with double Q-learning[C]. Proceedings of the 30th AAAI Conference on Articial Intelligence. Phoenix: AAAI, 2016: 2094-2100.

[18] Schaul Tom, Quan J, Antonoglou I, et al. Prioritized experience replay[C]. Proceedings of the 4th International Conference on Learning Representations. San Juan: ICLR, 2016: 1-21.

[19] Wang Z, Schaul T, Hessel M, et al. Dueling network architectures for deep reinforcement learning[C]. Proceedings of the 33rd International Conference on Machine Learning. New York: JMLR, 2016: 1995-2003.

[20] Hessel M, Modayil J, van Hasselt H, et al. Rainbow: combining improvements in deep reinforcement learning[C]. Proceedings of the 32nd AAAI Conference on Articial Intelligence. New Orleans: AAAI, 2018: 3215-3222.

作者简介

张国富 (1979−), 男, 教授, 博士, 从事智慧应急、软件工程等研究, E-mail: zgf@hfut.edu.cn;

常加远 (1996−), 男, 硕士生, 从事机器学习、应急决策的研究, E-mail: 1982826624@qq.com;

苏兆品 (1983−), 女, 副教授, 博士, 从事音频安全、机器学习等研究, E-mail: szp@hfut.edu.cn;

沈宇锋 (1996−), 男, 硕士生, 从事强化学习、应急决策的研究, E-mail: 1731843273@qq.com.

(责任编辑: 郑晓蕾)