DOI:10.19651/j.cnki.emt.2211922

基于深度双 Q 网络的权值时变路网路径规划*

何 鑫 马 萍
(新疆大学电气工程学院 乌鲁木齐 830017)

摘 要:针对传统路径规划方法无法根据城市路网权值时变特性规划最优路径的问题,提出了基于深度双 Q 网络的权值时变路网路径规划方法。首先,构建权值时变的城市路网模型,其中,路段各时间段权值由随机函数产生。然后,设计了状态特征、交互动作和奖励函数对权值时变路网路径规划问题进行建模,利用 DDQN 算法训练智能体来学习路网权值时变特性,最后根据建模后的状态特征实现权值时变路网的有效路径规划。实验结果表明,DDQN 算法训练的智能体在权值时变路网中具有较好全局寻优能力。相比于滚动路径规划算法,所提方法在不同情况下规划的路径均最优,为权值时变路网的路径规划提供了一种新思路。

关键词:路径规划;权值时变路网;DDQN 算法;深度强化学习
中图分类号:TP181 **文献标识码**:A **国家标准学科分类代码**:580.20

Time-varying road network path planning based on double deep Q-network

He Xin Ma Ping
(School of Electrical Engineering, Xinjiang University, Urumchi 830017, China)

Abstract: Aiming at the problem that the traditional path planning method can not plan the optimal path according to the time-varying characteristics of urban road network weight, a time-varying road network path planning method based on double deep Q-network was proposed. Firstly, the urban road network model with time-varying weights is constructed, in which the weights at each time period of the road segment are generated by random functions. Then, the state features, interaction actions and reward functions are designed to model the time-varying weight network path planning problem, and DDQN algorithm is used to train the agent to learn the time-varying weight characteristics of the road network. Finally, the path is planned according to the modeled state features to realize the effective path planning of the time-varying weight network. The experimental results show that the agent trained by DDQN algorithm has better global optimization ability in the time-varying weight road network. Compared with the rolling path planning algorithm, the proposed method can plan the optimal path under different circumstances, which provides a new idea for the path planning of the road network with time-varying weights.

Keywords: path planning; time-varying road network; DDQN algorithm; deep reinforcement learning

0 引 言

无人驾驶作为工业革命和信息化的重要产物,是战略性新兴产业的重要组成部分,引起了各学术团队和互联网、汽车等领域公司的广泛关注,Google、Tesla、Baidu 等争相研发无人驾驶汽车。无人驾驶软件层面可分为感知预测、地图定位、决策规划、行为控制等。

无人驾驶车辆决策规划分为 3 块,分别是全局路径规划、行为规划与运动规划。合理的路径规划有助于解决城市交通问题和乘客高效出行[1]。出行时间长短作为乘客最关心的问题之一,将起讫点之间路径的累积行程时间作为路径评价指标。全局路径规划根据路段权值是否随时间变化分为静态路径规划(static path planning, SPP)和滚动路径规划(rolling path planning, RPP)。静态路径规划主要研究路段权值不随时间变化的情况下起始点到目标点路径累积权值(时间、距离等)最小问题。较为经典的 SPP 算法有 Dijkstra 算法[2]、A * 算法[3-4]、蚁群算法[5-6]等。

随着城市交通拥挤日趋“常态化”,城市交通网络的路段行程时间不再是静态不变的,而是随着时间不断变化的。基于静态网络的路径分析结果越来越不符合实际情况,因

收稿日期:2022-11-02
* 基金项目:国家自然科学基金(51967019)、国家自然科学基金(52065064)、天山青年计划(2020Q066)项目资助

此适用于城市道路时变网络路径分析的方法具有研究价值和意义。文献[7]中每一次路网权重更新，都会以当前节点作为起点利用 Dijkstra 算法重新规划路径。李军[8]提出通过神经网络预测道路未来时刻交通流信息，然后转换为动态时间权重，多个周期内调用路径规划算法。文献[9]通过 V2V/V2I(vehicle to vehicle/vehicle to infrastructure)设施获取实时交通信息做出路径选择决策。常盟盟等[10]提出的动态路径规划方法在计算过程中融合了时空变化特征，周期性的对路网状态进行更新。文献[11]提出了一种基于群体感知的动态路径规划系统。张晓楠等[12]考虑车辆旅行速度的时变特征，计算动态旅行时间，用蚁群算法规划每个时段的最优路径。Zhao 等[13]提出在车联网中引入信息中心网络架构，并进行大数据采集和分析，滚动规划无人车路径。文献[14]建立了一种限制搜索区域的时变权重有向图模型，根据不断变化的交通流自适应地选取最优路径。综上，对于时变路网路径规划问题，RPP 的核心思想是反复调用路径规划算法，不断更新汽车行驶路线，因此汽车实际行驶路线是由多阶段规划的局部最优路径拼接而成。但 RPP 并不能得到全局最优路径，因为 RPP 在每个时段规划路径仅根据当前时段路径上的权值没有考虑未来多时段路径权值变化信息。

深度强化学习将深度学习的感知能力和强化学习的决策能力相结合。为了在规划路径时考虑路网权值时变特性，本文用深度双 Q 网络(double deep Q-network，DDQN)训练的智能体进行路径规划，并与 RPP 算法规划的路径做对比。实验显示，DDQN 算法训练的智能体规划的路径上累积行程时间比 RPP 算法更小，对缩短出行时间、促进城市路网的交通便利具有一定的研究价值。

1　权值时变城市路网

在城市路网中实现全局路径规划之前，将城市路网视为一个有向加权图，图定义为：

$$G=(V,E,W) \tag{1}$$

式中：V 代表道路交点合集，E 代表道路集合，W 代表道路权值的集合[15]。

对于 W 集合中权值为当前道路长度时，路网表现为静态，当使用道路通行时间 h 作为权值时(即 $W=h$)，因为城市道路不同时间段拥挤程度不一样所以权值也会随时间段不同发生变化，此时路网表现为动态。已有研究表明交通流存在周期的重复性或相似性，文献[16]提出利用行程时间波动性建立道路阻抗函数模型也呈现周期相似。

1.1　时变路网中车辆时间状态更新

以图 1 所示路网为例，各边的 3 个权值分别代表其边上时间段 1、时间段 2 和时间段 3 的时间权值(单位为 min)。

设路网权值变化周期为 30 min，在时间段 a 第 t_{now} min 车辆位置为节点 node1，准备前往节点 node2，则时间状态更新如图 2 所示。

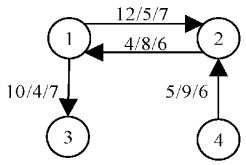

图 1　路网结构图

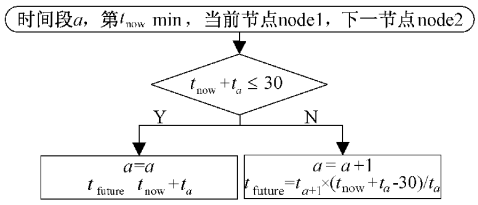

图 2　时间状态更新图

图 2 中 t_a 为在时间段 a 时车辆从节点 node1 移动到节点 node2 所花的时间，t_{a+1} 为在时间段 a+1 车辆从节点 node1 移动到节点 node2 所花的时间，t_{future} 为车辆到达节点 node2 时所在时间段的第 t_{future} 分钟。如图 1 所示路网：1)若小车在时间段 1 第 10 min 从节点 1 出发去往节点 3，可知当前时间段下需花费 10 min，则小车到达节点 3 时时间状态为时间段 1 的第 20 min。2)若小车在时间段 1 第 25 min 从节点 1 出发去往节点 3，可知在 5 min 后，进入时间段 2，节点 1 到节点 3 的路段权值由 10 变为 4，则根据时间状态更新规则小车会在时间段 2 的第 2 min 到达节点 3。

1.2　构建权值时变路网

随着云计算、大数据、移动计算和物联网的兴起，作为物联网分支在汽车领域的发展，车联网是以行驶中的车辆为信息感知对象，借助新一代信息通信技术，实现车与 X(即车与车、人、路、服务平台)之间的网络连接，提升车辆整体的智能驾驶水平，为用户提供安全、舒适、智能、高效的驾驶感受与交通服务，同时提高交通运行效率，提升社会交通服务的智能化水平，但复杂的城市路网历史统计数据目前情况下难以得到，因此建立如图 3 所示 6×6 路网来验证深度强化学习在全局路径规划上的应用。时间上分为时间段 1、时间段 2 与时间段 3，每个时间段时长为 30 min，在不同时间段同一路段时间权值不一，每个路段的权值为随机函数产生的随机数。

2　深度 Q 网络

强化学习中，t 时刻回报表示为：

$$U_t=\sum_{k=t}^{n}\gamma^{k-t}R_k \tag{2}$$

式中：γ 为奖励折扣率，R_k 为 k 时刻获得的奖励。

$t+1$ 时刻回报表示为：

$$U_{t+1}=\sum_{k=t+1}^{n}\gamma^{k-t-1}R_k \tag{3}$$

根据式(2)、(3)可以得到：

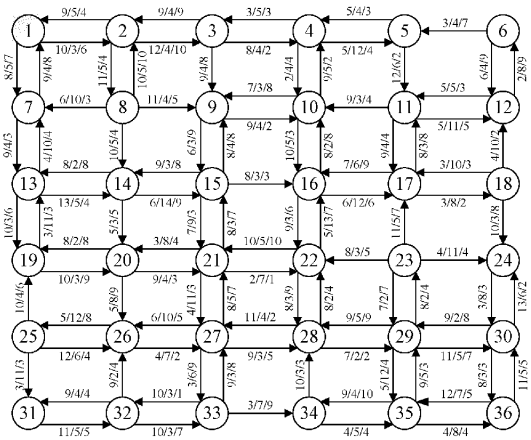

图 3　6×6 路网

$$U_t = R_t + \gamma \sum_{k=t+1}^{n} \gamma^{k-t-1} R_k = R_t + \gamma U_{t+1} \tag{4}$$

最优状态－动作价值函数定义如下：

$$Q_*(s_t, a_t) = \max_{\pi} Q^{\pi}(s_t, a_t) = \max_{\pi} E\{U_t \mid S_t = s_t, A_t = a_t\} \tag{5}$$

根据式(4)、(5)可以得到最优贝尔曼方程(optimal Bellman equations)：

$$Q_*(s_t, a_t) = E_{S_{t+1} \sim p(\cdot \mid s_t, a_t)}\{R_t + \gamma \max_{\pi} Q_*(S_{t+1}, A) \mid S_t = s_t, A_t = a_t\} \tag{6}$$

贝尔曼方程右边是个期望，对期望做蒙特卡洛近似。

$$Q_*(s_t, a_t) = r_t + \gamma \max_{a \in A} Q_*(s_{t+1}, a) \tag{7}$$

深度强化学习中用一个函数 $Q_{\varphi}(s,a)$ 来近似计算最优状态动作价值函数 $\max_{\pi} Q^{\pi}(s,a)$，称为最优状态-动作价值函数近似，其中 s、a 分别是状态 s 和动作 a 的向量表示，π 表示当前采用的策略，φ 是神经网络。φ 输出为一个实数，称为 Q 网络。深度 Q 网络(deep q-network，DQN)是 value-based 的方法，该类算法学习的不是策略而是 critic，critic 对某个状态下执行某个动作打分来评价这个状态下此动作的好坏，即最优状态-动作价值评估(optimal state-action value evaluation)。

将式(7)中最优动作价值函数 $Q_*(s,a)$ 替换成神经网络 $Q(s,a;\omega)$，得到：

$$Q(s_t, a_t; \omega) \approx r_t + \gamma \max_{a \in A} Q(s_{t+1}, a; \omega) \tag{8}$$

式(8)左边 $\hat{q}_t \triangleq Q(s_t, a_t; \omega)$ 是神经网络在 t 时刻做出的预测，其中没有任何事实成分。右边的 TD 目标 $\hat{y}_t \triangleq r_t + \gamma \max_{a \in A} Q(s_{t+1}, a; \omega)$ 是部分基于神经网络在 $t+1$ 时刻做出的预测，部分基于真实观测到的奖励 r_t。

定义损失函数：

$$L(\omega) = \frac{1}{2}[Q(s_t, a_t; \omega) - \hat{y}_t]^2 \tag{9}$$

L 关于 ω 的梯度为：

$$\nabla_{\omega} L(\omega) = (\hat{q}_t - \hat{y}_t) \nabla_{\omega} Q(s_t, a_t; \omega) \tag{10}$$

做一步梯度下降，可以让 $\hat{q}_t$ 更接近 $\hat{y}_t$：

$$\omega \leftarrow \omega - \alpha(\hat{q}_t - \hat{y}_t) \nabla_{\omega} Q(s_t, a_t; \omega) \tag{11}$$

文献[17]指出 DQN 算法训练出来的神经网络对最优状态动作价值往往是被高估的，并且提出加入目标网络。经过理论和实验论证，加入目标网络能有效减小 Q 网络预估出来的 Q 值和真实 Q 值之间的差异。所以用 DDQN 来训练智能体相对于 DQN 会得到更好的效果。深度神经网络作为有监督学习模型，要求数据满足独立同分布，为了打破数据间的关联性，Experience Replay 方法通过存储-采样的方法打乱数据间的关联性，此外还提高了数据的利用率[18]。

3　基于 DDQN 的权值时变路网路径规划

首先定义权值时变路网路径规划问题的强化学习形式，需要设计一个与智能体交互的环境(包括状态、动作、回报函数)，然后用 DDQN 算法训练智能体，最后利用智能体规划路径。将在权值时变路网上寻找累积行程时间最小的路径问题变为智能体在环境中做出决策动作获得最大累积奖励问题。

3.1　智能体交互环境设计

1) 状态的设计

在马尔可夫决策过程(Markov decision process，MDP)中，状态信息代表了 Agent 所感知到的环境信息及其动态变化，是深度强化学习(deep reinfocement learning，DRL)算法生成决策和评估长期收益的依据，而状态空间的设计直接决定 DRL 算法能否收敛、收敛速度以及最终性能。严格来讲，MDP 中的状态包含了完整描述当前环境和 Agent 所需的全部信息，但状态信息里的无用成分会导致算法难以收敛或性能不佳，所需训练时间也会显著延长，会损害算法的实用性，而且在样本量有限时可能诱使 DRL 算法学习到虚假的决策相关性并造成局部过拟合。一些高误差、高漂移、高噪声的状态信息还会直接对模型学习起到反作用，因此需要输入高效状态信息。状态信息应该能够有效反映动作对状态的影响。

通过分析权值时变路网路径规划的任务，设计了一个 8 维向量表征智能体状态，包含起讫点信息、时间段信息、时间段内时间信息、当前节点动作规则信息。当前节点动作规则信息直接与回报函数中奖励项和惩罚项即时联动属于直接相关信息。起讫点信息、时间段信息与时间段内时间信息使得 Agent 能够直接认识到当前自身的情况，但并没有在回报函数中即时变现出来，故属于间接相关信息。智能体状态设计如式(12)所示。

$$state = (S, E, T_s, T_t, action) \tag{12}$$

式中：S 为智能体当前所在节点，E 为终止节点（即目标节点），T_s 为当前所处时间段，T_t 为时间段内某一时刻，$action$ 为一个四维的动作向量（智能体当前节点可执行的动作）。以图 3 所示路网为例，若小车当前位于节点 5，目标节点为 36，当前时间状态为时间段 2 的第 22 min，则当前小车状态可用向量(5,36,2,22,0,1,1,0)表示，(0,1,1,0)代表动作上下左右是否可行(1 表示可以，0 表示不可以)。如果当前状态执行向下这个动作，那么下一状态为(11,36,T_s,T_t,0,1,1,1)，其中 T_s、T_t 可由第 1.1 节时间状态更新算得。

2) 动作的设计

精心设计的动作空间能够显著提升 DRL 算法的探索效率，从而降低算法的学习难度，并提升其最终性能。对于特定任务而言，动作空间在事实上决定了任何算法所能达到的性能上限。动作空间主要包括离散式和连续式，考虑城市路网节点的结构特点，故选取离散式动作空间。

3) 回报函数的设计

Agent 根据探索过程中来自环境的反馈信号持续改进策略和评价。完整的回报信号生成规则被称作回报函数，DRL 算法学习过程的本质是回报函数指引下的神经网络对输入状态信息的特征深加工，以及这些深层特征与值估计和决策相关性的建立过程。定性目标的达成或者定量目标的改善统称为任务的主线事件，主线事件产生的直接回报称为主线回报。主线回报相对于任务目标来说往往是无偏的。在强化学习中，具有较高探索难度的任务缺乏反馈信号造成学习困难的现象被称作稀疏回报问题。稀疏奖励是强化学习应用在机器人路径规划中比较棘手的问题，在稀疏奖励环境下机器人与环境交互过程十分耗时，如果交互样本无法获得奖励，那么该样本对于算法训练的贡献量将很小[19]。式(13)为本文设计回报函数。

$$R = \begin{cases} R_{main} + R_{sup}, & \text{action is allowable} \\ R_{pun}, & \text{action is not allowable} \end{cases} \tag{13}$$

式中：R 表示获得的奖励；R_{main} 为主线奖励，本文中主要取决于经过路段所消耗时间，且当车辆到达目标点时会获得一个较大的正奖励；R_{sup} 是为了刺激智能体不断靠近目标节点的辅助奖励，实验中采用曼哈顿距离来设计 R_{sup}；R_{pun} 为惩罚，主要是避免智能体做出不合法的动作。

3.2　DDQN 算法训练智能体

DDQN 算法训练智能体如图 4 所示，详细步骤如下所示：

步骤 1)初始化主网络和目标网络参数，EPISODE∈[0,MAXEPISODE]为算法训练次数；

步骤 2)将路径规划问题转化为训练智能体需要的状态特征 s_0；

步骤 3)用当前状态 s_t 作为主网络输入(若为一个新 episode，则输入为 s_0)，得到深度 Q 网络对所有动作相应的 Q 值输出。根据“ε-贪婪”策略以 1−ε 的概率选择 Q 值最大的动作 a；

步骤 4)在状态 s_t 下执行动作 a，得到新状态 s_{t+1} 和奖励 r，将本次产生的经验 (s, a, r, s_{t+1}) 放入经验回放池中；

步骤 5)从经验回放池中采样 m 个样本，计算目标 Q 值与主网络 Q 值，使用均方差损失函数通过神经网络的梯度反向传播来更新主网络参数；

步骤 6)每隔常数个 episode 将主网络参数赋值给目标网络参数；

步骤 7)检查 s_{t+1} 是否为终止状态。若是，判断是否达到设定的训练次数，若未到达则当前 episode 结束跳转到步骤 2)，否则结束算法训练。若不是，令 $s_t = s_{t+1}$ 跳转到步骤 3)。

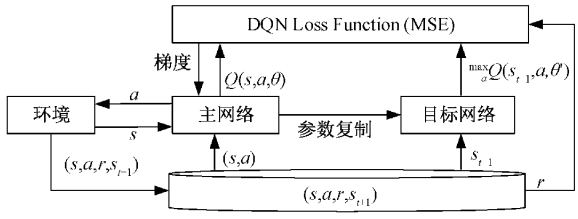

图 4　更新神经网络参数

3.3　智能体规划路径

DDQN 算法训练的智能体规划路径如图 5 所示。智能体收集当前状态输入到深度 Q 网络中，经深度 Q 网络输出各动作的 Q 值，选择最大 Q 值的动作由智能体执行，计算当前动作下的奖励，更新环境并产生新的状态，往复循环，直到达到目标节点。

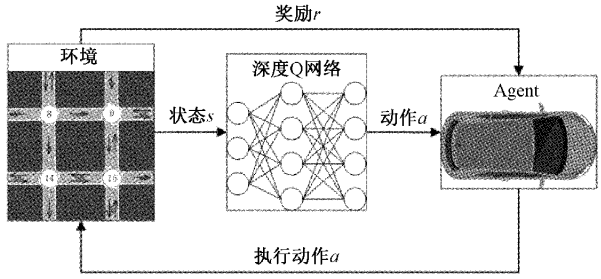

图 5　智能体规划路径

4　仿真实验及对比分析

4.1　智能体训练

1) 实验设置

实验使用 Python 编程语言，在 Windows11 系统上运行 Pycharm 使用 Pytorch 实现整体框架。硬件条件为 AMD Ryzen 7 5800H，NVIDIA Geforce RTX 3060(显存为 6 G)，16 GB RAM。

2) 参数设置

本文所提出算法包含 3 个关键参数：(1)学习率(learning rate，LR)；(2)奖励折扣率 γ；(3)经验回放区容

量(capacity of repaly buffer, CORB)。采用实验设计(design of experiment, DOE)方法分析 3 个参数对算法性能的影响。每个参数设置 3 个水平,LR∈{1×10^{-3},1×10^{-4},1×10^{-5}},γ∈{0.5, 0.7, 0.9},CORB∈{5×10^3, 5×10^4, 5×10^5}。选择正交数组 $L_9(3^3)$,包含 9 组参数水平组合。对于每个组合在相同情况下计算规划路径的累积行程时间作为该组参数响应值(response value, RV),如表 1 所示。

表 1　正交实验表及各参数组合响应值

编号	参数因子组合			RV
	CORB	γ	LR	
1	1	1	1	86
2	1	2	3	88
3	1	3	2	52
4	2	1	3	66
5	2	2	2	60
6	2	3	1	64
7	3	1	2	78
8	3	2	1	61
9	3	3	3	88

对表 1 中的数据进行极差分析可得表 2,通过极差分析,能够分析各参数间的优势及参数间具体水平的优劣。表 2 中 K 值为某参数某水平时试验数据求和;K_avg 为对应的平均值;最佳水平指某参数的最佳 K_avg 对应的水平标号;R 指参数的极差值,该值越小则该参数对于算法性能影响越大。参数的趋势变化如图 6 所示。根据表 2 以及图 6 可得,学习率对于算法性能影响最大,经验回放区容量和奖励折扣率次之。学习率过大会造成网络不收敛,太小会导致网络收敛非常缓慢。

表 2　参数各水平的极差分析

值	水平	CORB	γ	LR
K 值	1	226	230	211
	2	193	209	190
	3	226.8	203.8	241.8
K_avg 值	1	75.33	76.67	70.33
	2	63.33	69.67	63.33
	3	75.6	67.93	80.6
最佳水平		2	3	2
R		−12.27	−8.73	−17.27
水平数量		3	3	3
每水平重复数		3	3	3

经过对参数实验分析,DDQN 算法训练智能体采用参数如下:训练回合数为 50 000,学习率为 0.000 1,奖励折扣率为 0.9,经验回放区容量为 50 000,batch size 大小选择 2 048,目标 Q 网络参数更新频率为 500。

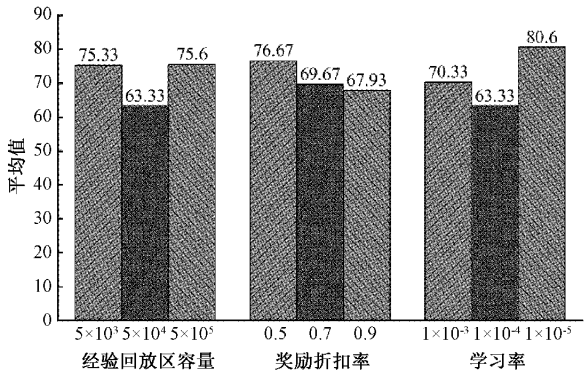

图 6　参数各水平均值

3) 训练智能体

DDQN 算法训练智能体过程中探索率变化趋势与单个 episode 获得奖励变化如图 7、8 所示。

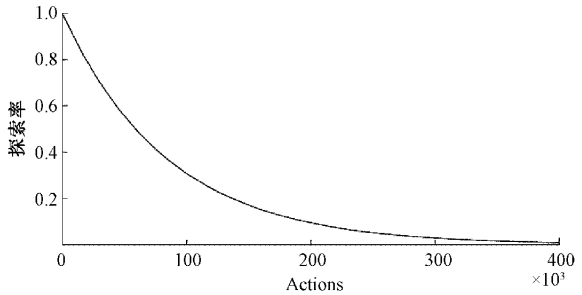

图 7　探索率变化图

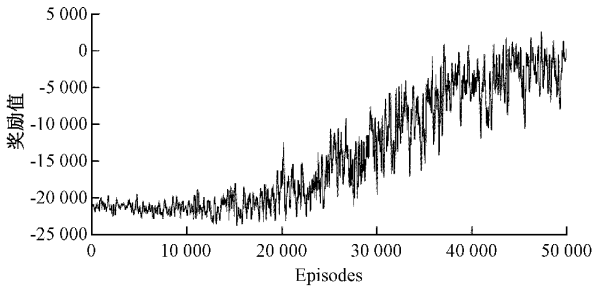

图 8　单个 episode 获得奖励值变化图

从图 7 可以看出,训练智能体过程中探索率会随着迭代次数增大而变小:(1)训练前期探索率大,智能体尝试不同的行为继而收集更多的信息;(2)训练后期探索率小,智能体收集到了足够的经验,需要利用当前信息做出最佳决定。图 8 显示在训练智能体前期,因为主网络不能输出一个好的动作和较大的探索率,智能体或做出不合法的动作或执行的动作不能获得正奖励,所以前期单个 episode 获得奖励很低。但随着神经网络的参数迭代更新,探索率的不断减小,单个 episode 获得奖励大体上呈现出稳定上升的趋势,在神经网络对最优状态-动作价值近似估计的基础上智

能体在每个状态做出的动作越来越好。图 9 显示 DDQN 算法在 6×6 的拓扑网络中训练智能体能很好的收敛。

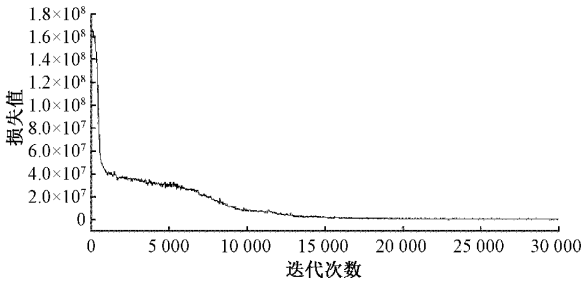

图 9　损失函数值变化图

4.2　实验对比与分析

1) RPP 内嵌算法选择

对比算法选择时变路网路径规划中常用的 RPP 算法。基于 RPP 算法特点，在用 RPP 解决时变路网路径规划问题前，需确定 RPP 内嵌算法。蚁群算法作为智能启发式算法代表之一，将蚁群算法与 Dijkstra 算法在路径寻优能力上进行对比，两者间寻优能力更强的算法作为 RPP 算法的内嵌算法。基于第 1 节提到的 6×6 路网的第 1 时段进行仿真实验，起讫点选择(1，36)。路径规划结果如表 3 所示。图 10 为蚁群算法迭代更新过程中路径累积行程时间变化图。

表 3　Dijkstra 算法与蚁群算法规划路径

算法	规划路径	路径累积行程时间/min	规划耗时/s
蚁群算法	1→7→13→14→15→16→17→18→24→30→36	74	0.073
Dijkstra 算法	1→7→13→14→20→26→27→33→34→35→36	58	0.001

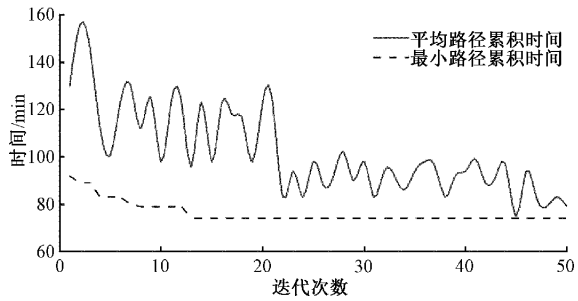

图 10　路径累积行程时间随迭代次数的变化

由表 3 可知，Dijkstra 算法规划路径优于蚁群算法规划的路径。分析两者规划的路径，可以发现蚁群算法规划的路径在某些路段与 Dijkstra 算法规划路径重合，说明其具有一定的寻优能力。但蚁群算法自带随机因子，在路径搜索过程中受随机因素影响，蚂蚁信息素的引导加快收敛的同时，算法也容易陷入局部最优。如图 10 可知迭代了 12 次以后，蚁群算法陷入了局部最优，但 Dijkstra 算法作为广度优先搜索算法则能找到全局最优。Dijkstra 算法与蚁群算法规划路径分别用时 0.001、0.073 s。Dijkstra 算法以较小的搜索时间获得了最优路径，故选取 Dijkstra 算法作为 RPP 算法的内嵌算法。

2) 不同方法规划路径对比

分别使用 RPP 算法与 DDQN 算法训练的智能体在 1.2 节提到的 6×6 的拓扑网络中规划路径。为了不失一般性，起讫点选择了两组，分别为(1，36)、(6，31)，出发时间点选择两组。两种方法进行路径规划结果如表 4 所示。

表 4　不同起讫点不同方法规划路径

起讫点	出发时间	算法	路径	路径累积行程时间/min
(1，36)	时间段 1 第 0 min	RPP 算法	1→7→13→14→20→21→22→28→29→30→36	57.0
		DDQN 算法	1→2→8→9→15→16→22→28→29→30→36	52.7
	时间段 1 第 15 min	RPP 算法	1→7→13→19→20→21→22→28→29→30→36	45.9
		DDQN 算法	1→2→3→9→15→16→22→28→29→30→36	43.3
(6，31)	时间段 1 第 0 min	RPP 算法	6→5→4→10→9→15→21→27→33→32→31	54.0
		DDQN 算法	6→5→4→10→16→22→28→27→33→32→31	49.6
	时间段 1 第 15 min	RPP 算法	6→5→4→10→9→15→16→22→28→27→33→32→31	44.9
		DDQN 算法	6→5→4→10→16→22→28→27→33→32→31	40.5

表 4 显示当起讫点为(1，36)出发时间分别为时间段 1 第 0 min 和时间段 1 第 15 min 时，DDQN 训练的智能体规划的路径在累积行程时间上要比 RPP 算法分别减少 4.3 与 2.6 min。当起讫点为(6，31)出发时间分别为时间段 1 第 0 min 和时间段 1 第 15 min 时，DDQN 训练的智能体规划的路径在累积行程时间上要比 RPP 算法分别减少 4.4 与 4.4 min。相同起讫点不同出发时间，智能体规划的路径有的一样，有的不一样，但都保证了路径的最优性。相

对于 RPP 算法,DDQN 算法训练的智能体能够很好的学习到权值时变路网的时变特性,在综合考虑当前时段及未来时段路径上的权值变化基础上,一开始就能做出合理的路径规划,所以 DDQN 算法训练的智能体在不同情况下均能规划出比 RPP 算法更优的路径。

5　结　　论

根据权值时变路网特征设计了智能体交互环境,将权值时变路网路径规划问题转换为深度强化学习决策问题。实验证明了 DDQN 算法训练的智能体在权值时变路网中进行路径规划具有独特优势。相对于其他算法,DDQN 算法训练的智能体根据初始状态就可以规划一条更好的路径,并且没有类似于 Dijkstra 算法的搜索过程,具有高效性,但是随着路网规模的扩大,也会对训练智能体的算法提出挑战。

参考文献

[1] ABDALLAOUI S, AGLZIM E H, CHAIBET A, et al. Thorough review analysis of safe control of autonomous vehicles: Path planning and navigation techniques[J]. Energies, 2022, 15(4): 1358.

[2] 巩慧, 倪翠, 王朋, 等. 基于 Dijkstra 算法的平滑路径规划方法[J]. 北京航空航天大学学报, 2022, 48, DOI:10.13700/j.bh.1001-5965.2022.0377.

[3] 姜媛媛, 张阳阳. 改进 8 邻域节点搜索策略 A* 算法的路径规划[J]. 电子测量与仪器学报, 2022, 36(5): 234-241.

[4] 张建光, 张方, 陈良港, 等. 基于改进 A* 算法的自动引导车的路径规划[J]. 国外电子测量技术, 2022, 41(1): 123-128.

[5] 肖金壮, 余雪乐, 周刚, 等. 一种面向室内 AGV 路径规划的改进蚁群算法[J]. 仪器仪表学报, 2022, 43(3): 277-285.

[6] LIU J, ANAVATTI S, GARRATT M, et al. Modified continuous ant colony optimisation for multiple unmanned ground vehicle path planning[J]. Expert Systems with Applications, 2022, 196: 116605.

[7] GAO L N, TAO F, MA P L, et al. A short-distance healthy route planning approach [J]. Journal of Transport & Health, 2022, 24: 101314.

[8] 李军. 城市智能交通中的动态路径规划研究[D]. 杭州: 杭州电子科技大学, 2016.

[9] CHAI H, ZHANG H M, GHOSAL D, et al. Dynamic traffic routing in a network with adaptive signal control[J]. Transportation Research Part C: Emerging Technologies, 2017, 85: 64-85.

[10] 常盟盟, 袁磊, 丁治明, 等. 交通路况感知下的自适应动态路径规划方法[J]. 交通运输系统工程与信息, 2021, 21(4): 156-162,247.

[11] EL-WAKEEL A S, NOURELDIN A, HASSANEIN H S, et al. IDriveSense: Dynamic route planning involving roads quality information [C]. 2018 IEEE Global Communications Conference (GLOBECOM), IEEE, 2018: 1-6.

[12] 张晓楠, 王陆宇, 谭昕妮, 等. 时变条件下道路网的车辆路径优化[J]. 机械科学与技术: 1-9[2023-11-22] https://doi.org/10.13433/j.cnki.1003-8728.20220230.

[13] ZHAO C, DONG M, OTA K, et al. Edge-MapReduce-based intelligent information-centric IoV: Cognitive route planning[J]. IEEE Access, 2019, 7: 50549-50560.

[14] 贾新春, 彭登永, 李雷, 等. 城市路网的一种最优路径搜索算法[J]. 山西大学学报(自然科学版), 2020, 43(1): 58-64.

[15] 肖浩, 廖祝华, 刘毅志, 等. 实际环境中基于深度 Q 学习的无人车路径规划[J]. 山东大学学报(工学版), 2021, 51(1): 100-107.

[16] 温惠英, 卢德佑, 汤左淦. 考虑行程时间波动性的城市道路阻抗函数模型[J]. 公路工程, 2019, 44(3): 27-32.

[17] VAN H H, GUEZ A, SILVER D. Deep reinforcement learning with double q-learning [C]. Proceedings of the AAAI Conference on Artificial Intelligence, 2016.

[18] 曹茜. 离线策略下强化学习经验重放机制优化方法研究[D]. 北京: 北京交通大学, 2020.

[19] 王军, 杨云霄, 李莉. 基于改进深度强化学习的移动机器人路径规划[J]. 电子测量技术, 2021, 44(22): 19-24.

作者简介

何鑫,硕士研究生,主要研究方向为无人车路径决策规划。E-mail: 18846055337@163.com

马萍,副教授,硕士生导师,主要研究方向为深度学习、机械信号处理与分析、大数据下智能故障诊断与状态检测。E-mail: 694073078@qq.com