文章编号:1007-5321(2021)01-0097-07 **DOI**:10.13190/j.jbupt.2020-006

基于 Sumtree DDPG 的智能交通信号控制算法

黄 浩[1,3,4], 胡智群[2], 王鲁晗[1,3,4], 路兆铭[1,3,4], 温向明[1,3,4]

(1. 北京邮电大学 信息与通信工程学院,北京 100876; 2. 湖北大学 计算机与信息工程学院,武汉 430062;
3. 北京邮电大学 网络体系构建与融合北京市重点实验室, 北京 100876; 4. 北京邮电大学 先进信息网络北京实验室,北京 100876)

摘要: 提出了基于和树—深度确定性策略梯度(Sumtree DDPG)的多路口智能交通信号控制算法,通过对交叉路口数据的实时观测,智能地调控交通信号周期时长、相位顺序以及相位持续时间,提高路口通行效率. 同时,基于和树结构的经验数据存储模式提高采样效率,加速了算法收敛. 仿真结果表明,在动态环境下,该算法在车辆排队长度、车辆等待时间、车辆平均速度等性能指标上均优于现有的固定配时方案和基于流量权重的配时算法.

关 键 词: 智能交通; 交通信号控制; 深度强化学习; 深度确定性策略梯度; 多路口

中图分类号: U491.54 **文献标志码**: A

Intelligent Traffic Signal Control Algorithm Based on Sumtree DDPG

HUANG Hao[1,3,4], HU Zhi-qun[2], WANG Lu-han[1,3,4], LU Zhao-ming[1,3,4], WEN Xiang-ming[1,3,4]

(1. School of Information and Communications Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China;
2. School of Computer and Information Engineering, Hubei University, Wuhan 430062, China;
3. Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, Beijing 100876, China;
4. Beijing Laboratory of Advanced Information Networks, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: A multi-intersection intelligent traffic signal control algorithm based on sumtree deep deterministic policy gradient(Sumtree DDPG) is proposed. Through real-time observation of intersection data, the cycle length, phase sequence and phase duration of the traffic signal can be intelligently adjusted to improve the efficiency of intersections. Meanwhile, the empirical data storage mode based on sumtree structure can improve the sampling efficiency and accelerate the algorithm convergence. Compared with fixed signal timing and signal timing algorithm based on traffic flow weight, a simulation is carried out that the proposed algorithm obtains good performance in vehicle queue length, vehicle waiting time and vehicle average speed in dynamic environment.

Key words: smart transportation; traffic signal control; deep reinforcement learning; deep deterministic policy gradient; multiple intersections

随着全球汽车保有量的持续增长,交通拥堵问题已成为全球各大城市的难点和热点问题. 交通拥堵会影响城市经济发展,造成资源浪费,还会导致严重的环境污染. 因此,交通信号最优控制系统得到广泛地研究. 传统的交通信号控制方案通常是基于历史车流量预设的固定配时,或者基于当前车流状态调整交通信号时长. 绿信比、周期、相位差优化技术[1](SCOOT, split cycle offset optimizing technique)

收稿日期: 2020-01-17
基金项目: 国家自然科学基金项目(61901163); 北京市科技新星计划项目(Z191100001119028)
作者简介: 黄 浩(1997—), 男, 博士生.
通信作者: 胡智群(1989—), 女, 副教授, 硕士生导师, E-mail:zhiqunhu520@163.com.

和悉尼自适应交通控制系统[2](SCATS,Sydney coordinated adaptive traffic system)已在全世界广泛使用. 但是这些信号控制方案缺乏自适应性和预见性,甚至会造成大量的人力负担,因此,需要更加智能的控制系统进行交通信号的控制优化.

近年来,强化学习的研究不断深入,作为一种反馈式学习,强化学习通过"失败与尝试"机制,不断积累经验学得最优策略. 在交通信号控制中,每个交叉路口配时方案的确定都可由一个强化学习中的智能体实时决策,智能体根据路网环境获取该路口的交通状态信息,并根据策略制定该路口信号配时方案,随后智能体会收到环境反馈的奖励,并通过不断调整,使累计奖励最大化,最终实现配时的优化.

1 相关工作

现有文献对道路交叉口的信号控制策略进行了广泛研究. Webster 方法[3]在假设车辆均匀到达的情况下,通过数学模型计算最佳的单路口周期长度和相位配时占比,以最小化车辆通过路口的行驶时间. SCOOT 自适应控制系统根据交通流量的改变,周期性地调整每个相位信号灯的绿灯时长,从而减少车辆在路口的平均等待时间[1]. 但是 SCOOT 系统中的配时方案基于数学模型,当交通条件复杂度提升时,交通模型较难建立. SCATS 系统[2]依据实际交通状况从事先制定好的配时方案中选择最佳配时,但由于方案数量有限,系统的可靠性较差.

近年来,随着人工智能的兴起,越来越多的强化学习算法应用于交通控制领域. Abdulhai 等[4]提出了一种基于 Q 学习的交通信号控制方法,Q 学习通过 Q 值表进行值函数的存储,该方法无法适应复杂的环境,而且若状态空间过大会带来存储与收敛慢的问题. Genders 等[5]在深度 Q 学习(DQN,deep Q network)算法的基础上,利用离散流量状态编码方法,将交叉口的实时状态转化为不同的元胞作为状态输入,相较于之前仅使用排队长度作为环境状态,该算法能够获得更加全面的实时交通信息. 上述 2 种算法只能用于离散动作空间,但离散化会带来量化误差,使算法往往无法获得最佳的配时效果. 基于策略的强化学习算法能有效地解决这个问题,例如策略梯度(PG,policy gradient)算法,深度确定性策略梯度(DDPG,deep deterministic policy gradient)算法等. Richter 等[6]提出了基于 PG 算法的多交叉口协作信号配时方案,每个交叉口与其相邻交叉口共享状态信息,协作进行交通信号配时的学习. Pang 等[7]进一步改进,提出了基于深度确定性策略梯度的智能交通控制算法,但其只在单路口场景下完成了实验验证. Casas[8]在多个路口场景下对基于 DDPG 的智能信号控制策略进行了仿真验证,但是动作空间固定了相位顺序.

为了进一步提高交通信号控制效率,提出了基于和树—深度确定性策略梯度(Sumtree DDPG,sumtree deep deterministic policy gradient)的智能交通信号控制算法,根据不同相位、不同车道的车流量信息,智能决策配时周期、相位顺序以及各相位持续时间,解决了现有智能算法基于离散动作决策空间而带来的次优配时问题. 同时,所采用的 Sumtree 存储结构减少了记忆数据的相关性,提高了采样效率,加速了算法的收敛.

2 系统模型与算法设计

2.1 强化学习

强化学习能很好地解决例如交通信号控制这样的顺序决策问题. 在智能体与环境的整个交互过程中,智能体得到环境状态 $\boldsymbol{s}_t$,基于策略 $\boldsymbol{\pi}$,即智能体从环境感知到的状态和所采取动作的映射,采取动作 $\boldsymbol{a}_t$ 来响应该状态,随后环境对状态进行更新,并将下一状态 $\boldsymbol{s}_{t+1}$ 及奖励值 r_t 反馈给智能体,智能体利用环境返回奖励对上一动作进行评估,并更新其策略,这一循环一直持续,直到环境发出终止信号. 智能体与环境的交互构成了一个具有马尔可夫特性的动态系统.

强化学习的目标是使智能体学习一种最佳策略,以最大化从初始状态开始积累的预期奖励. 值函数 $Q(\boldsymbol{s},\boldsymbol{a})$ 定义为在状态 $\boldsymbol{s}$ 处采取动作 $\boldsymbol{a}$ 时所获得的未来累计奖励,可以用来对当前策略进行评估,具体形式可用式(1)表示:

$$\begin{aligned} &Q^{\pi}(\boldsymbol{s},\boldsymbol{a}) = \\ &E[r_t+\gamma r_{t+1}+\gamma^2 r_{t+2}+\cdots|\boldsymbol{s}_t=\boldsymbol{s},\boldsymbol{a}_t=\boldsymbol{a},\pi] = \\ &E\left[\sum_{k=0}^{\infty}\gamma^k r_{t+k}|\boldsymbol{s}_t=\boldsymbol{s},\boldsymbol{a}_t=\boldsymbol{a},\pi\right] \end{aligned} \tag{1}$$

其中:γ 为折扣因子,$\gamma\in[0,1]$,表明了未来回报相对于当前回报的重要程度;E 为数学期望. 最优值函数为

$$Q^{\pi^*}(\boldsymbol{s},\boldsymbol{a})=E_{s'}[r_t+\gamma\max_{\boldsymbol{a}'}Q^{\pi^*}(\boldsymbol{s}',\boldsymbol{a}')|\boldsymbol{s},\boldsymbol{a}] \tag{2}$$

其中 $\boldsymbol{s}'$ 与 $\boldsymbol{a}'$ 分别为下一步的状态和采取动作. 因此智能体通过不断更新自己的值函数直至逼近最优值

函数，便可学习到最优策略 π^*.

2.2　基于 Sumtree 的 DDPG 算法

DDPG 算法[9]通过 Actor 神经网络进行动作选择，Actcr 网络可参数化为 $\mu(\boldsymbol{s}|\theta^{\mu})$，网络输出为当前动作. 并通过 Critic 神经网络来拟合 Q 函数，将其参数化为 $Q(\boldsymbol{s},\boldsymbol{a}|\theta^{Q})$. 通过目标函数 J 衡量策略 μ 的表现，因此 Actor 网络的训练目标即为找寻最优策略，使得 $\mu=\arg\max_{\mu} J(\mu)$，通过链式规则，Actor 网络更新为

$$\nabla_{\theta^{\mu}} J \approx E\{\nabla_{\theta^{\mu}} Q[\boldsymbol{s},\mu(\boldsymbol{s}|\theta^{\mu})|\theta^{Q}]\} = E\{\nabla_{a} Q[\boldsymbol{s},\mu(\boldsymbol{s})|\theta^{Q}]\nabla_{\theta^{\mu}}\mu(\boldsymbol{s}|\theta^{\mu})\} \tag{3}$$

Critic 网络的更新使用类似于监督学习的方式，定义误差为均方误差形式：

$$L=\frac{1}{N}\sum_{i}[y_i - Q(\boldsymbol{s}_i,\boldsymbol{a}_i|\theta^{Q})]^2 \tag{4}$$

其中：$y_i = r_i + \gamma Q'[\boldsymbol{s}_{i+1},\mu'(\boldsymbol{s}_{i+1}|\theta^{\mu'})|\theta^{Q'}]$，$\mu'$和 Q'分别对应目标 Actor 网络和目标 Critic 网络，Actor 网络和 Critic 网络更新通过从记忆库中采样实现，N 为样本数. 此外，DDPG 算法在更新目标网络时采用了软更新的形式，使得目标网络参数变化小，训练更易于收敛，软更新形式为

$$\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'} \tag{5a}$$

$$\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'} \tag{5b}$$

在记忆库中，不同样本由于时序差分(TD，temporal-difference)误差的不同，在更新网络时对反向传播的作用也是不一样的，TD 误差越大，表明预测精度有更大的上升空间，对于反向传播的作用也越大，算法便能从中获得更多的有用信息. 因此为了进一步提高采样效率，加速算法收敛，采用 Sumtree[10]的形式对数据进行存储，如图 1 所示. 将 TD 误差的绝对值作为 Sumtree 中每个叶节点的存储值. 进行数据采样时，将优先级 p 的总和除以抽样数，得到区间数，然后在每个区间里随机选取一个数，将此数从 Sumtree 的根节点开始，按照一定的规律向下搜索，最后将搜索得到的优先级 p 与样本数据相对应，即可实现更高效的经验回放.

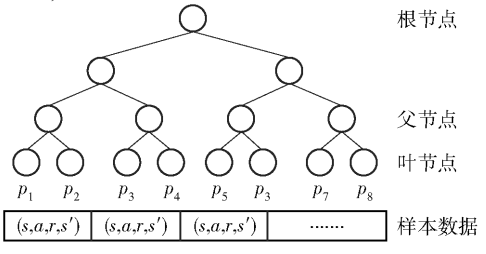

图 1　Sumtree 结构

2.3　智能交通信号控制算法

本小节主要阐述深度强化学习在智能交通系统中的应用，设计其状态、动作、奖励等信息，建立基于 DDPG 的交通信号控制算法 TSC-DDPG(traffic signal control-DDPG).

2.3.1　状态

状态为每一个路口智能体对其所观测到的路网环境的定量表示，在每一时间步，即智能体需要制定动作时，智能体都会通过路口部署的传感器接收路口的状态信息. 车道排队车辆的队列长度能够直观反应出交叉口各方向的拥堵情况，在此基础上，考虑车头时距的影响，车头时距定义为前后两车的前端通过交叉口停车线的时间差，将一周期内道路的平均车头时距作为一部分的状态. 因此，状态表示为

$$\boldsymbol{o}_t = \{q_1, q_2, \cdots, q_n, h_1, h_2, \cdots, h_n\} \tag{6}$$

其中：q_i 为道路 i 的队列长度，h_i 为道路 i 的上一周期内平均车头时距，n 为一个交叉口的道路总数.

2.3.2　动作

动作为路口智能体根据实时状态信息为每一个路口所制定的配时方案，动作的选取直接影响到配时方案及效果，从实用性角度出发，将动作空间定义为

$$\boldsymbol{a}_t = \{c, p_1, \cdots, p_m, d_1, \cdots, d_m\} \tag{7}$$

其中：c 为决定下一周期持续时间的时长因子，为避免出现周期过大或过小的情况，将周期时长限定在 $[c_{\min}T, c_{\max}T]$ 范围内，T 为基准周期长度；m 为相位数；$p_1,\cdots,p_m$ 为下一周期相位顺序；$d_1,\cdots,d_m$ 为通过 softmax 函数进行归一化处理后的比例因子，决定下一周期内各相位的持续时间. 考虑到 5 s 以下的相位持续时间过短，所提算法不进行 5 s 以下的相位，并将该时长按比例分配至其他相位.

2.3.3　奖励

奖励是对执行该动作后所达到的配时效果的评判，引导智能体学习的方向，是决定算法是否收敛以及达到期望目标的关键. 为了提高路口的通行效率，需综合考虑路口状况的各项评价指标. 针对每个交叉口定义其奖励函数为

$$r = w_1 W + w_2 X + w_3 Y + w_4 Z \tag{8}$$

其中：$w_1,\cdots,w_4$ 为权重系数；W 为该路口所有进口车道的车辆队列长度之和；X 为所有进口车道的车辆延迟之和，Y 为所有进口车道的车辆等待时间之

和,其中进口车道包括进口直行道和进口左转道;Z 为周期内该交叉口的吞吐量.

2.3.4 基于全局状态信息的 TSC-DDPG 算法

对于多路口的协同交通信号控制[11-12],TSC-DDPG 算法针对每个交叉口考虑整个路网状态信息,实行状态共享模式,赋予路口智能体更为广阔的观测视野,并由此得出全局最优值. 该方法可以通过估计交叉口间的相互关联来帮助导出全局最优 Q 值,使得多个智能体能够根据全局状态获取彼此间的时空信息,合理调整自身策略,实现路口间的协同优化. TSC-DDPG 算法的伪代码在算法部分给出. 首先随机初始化 Actor 神经网络 μ 及 Critic 神经网络 Q,并初始化目标网络 μ' 和 Q'、记忆库 R 以及用于平衡"利用与探索"的随机噪声 N. 在最初的训练阶段,由于策略远未达到最佳,需要不停地探索各种动作以增加策略的可能性. 随着算法的不断迭代,策略逐渐收敛,因此在后期阶段需要减少对动作的探索,以提高稳定性. 在算法实现中,采用均值为 0,初始方差为 1 的高斯噪声进行探索,当记忆库存满后,在每一个配时周期结束时对方差进行 0.995 倍的缩放,直至最后收敛.

每一回合开始,重新读入车流文件,以周期为更新频率进行信号配时. 当上一周期结束后,通过 Actor网络的输出确定下一周期的时长、相位顺序以及相位持续时间,如此循环直至达到最终仿真时长. 当采取完每一步动作后,由 Critic 网络对动作进行评估,从而使策略得到不断优化. Actor 网络的输入设定为全局状态值,通过对不同智能体的观测值进行拼接,全局状态可表示为 $\boldsymbol{s}=\{\boldsymbol{o}_1,\boldsymbol{o}_2,\cdots,\boldsymbol{o}_l\}$,$\boldsymbol{o}_i$ 为第 i 个智能体的观测值,Critic 网络输入依旧为自身的策略信息. 每个智能体都有一个独立的记忆库,在对记忆库数据进行存储时,状态输入更新为全局状态,即每一条数据可由 $\boldsymbol{e}=\{\boldsymbol{o}_1,\cdots,\boldsymbol{o}_l,\boldsymbol{a},r,\boldsymbol{o}'_1,\cdots,\boldsymbol{o}'_l\}$ 表示,仿真结果显示基于全局状态的 TSC-DDPG 算法具有良好的稳定性和鲁棒性.

TSC-DDPG 算法

初始化:随机初始化 Actor 神经网络 $\mu(\boldsymbol{s}_t|\theta^{\mu})$ 以及 Critic 神经网络 $Q(\boldsymbol{s},\boldsymbol{a}|\theta^{Q})$,初始化目标网络 μ' 和 Q',权重赋值为 $\theta^{Q'}\leftarrow\theta^{Q}$,$\theta^{\mu'}\leftarrow\theta^{\mu}$,初始化记忆库 R,高斯噪声 N.

for episode = 1, M do
在 SUMO 仿真平台导入车流文件
接收到初始观测状态 $\boldsymbol{s}_0$
  for step = 1, 10800 do
    根据策略生成动作 $\boldsymbol{a}_i=\mu(\boldsymbol{s}_i|\theta^{\mu})+N$
    var←var × 0.995
    所有路口智能体根据 $\boldsymbol{a}_i$ 执行相应动作
    根据式(8)计算奖励值
    得到新的状态 $\boldsymbol{s}_{i+1}$
    将样本数据$(\boldsymbol{s}_i,\boldsymbol{a}_i,r_i,\boldsymbol{s}_{i+1})$存储于记忆库 R
    从记忆库 R 中随机抽取样本进行学习训练
    $y_l=r_l+\gamma Q'[\boldsymbol{s}_{l+1},\mu'(\boldsymbol{s}_{l+1}|\theta^{\mu'})|\theta^{Q'}]$
    根据式(4)计算 L,并通过最小化 L 以更新 Critic 网络
    使用策略梯度更新 Actor 网络:

$$\nabla_{\theta^{\mu}}J\approx\frac{1}{L}\{\nabla_a Q[\boldsymbol{s}_l,\mu(\boldsymbol{s}_l)|\theta^{Q}]\nabla_{\theta^{\mu}}\mu(\boldsymbol{s}_l|\theta^{\mu})\}$$

    更新目标网络:

$$\theta^{\mu'}\leftarrow\tau\theta^{\mu}+(1-\tau)\theta^{\mu'}$$
$$\theta^{Q'}\leftarrow\tau\theta^{Q}+(1-\tau)\theta^{Q'}$$

  end for
end for

3 仿真实验

3.1 仿真环境与参数设置

仿真平台基于 SUMO 0.19 以及 Tensorflow 1.12 实现,利用 SUMO 提供的 Traci 接口模块完成两者的在线交互,具体的交通路网仿真设置如下.

1)路口属性设置

针对三路口场景进行仿真分析. 路口部署检测器的范围控制在 150 m 以内,相邻路口间距设置为 600 m,每条道路设置为双向,每向三车道,从内到外分别设置为左转车道、直行车道和右转车道,车道限速为 45 km/h.

2)车流量设置

车辆到达遵循泊松分布,通过更改每个交叉口入口的车辆到达概率来控制不同路段上的车辆数量. 仿真中将东西方向设置为主干道,在高峰时段,大部分车辆为东西走向,南北向车辆通过的概率较低,其中又以直行车辆较左转车辆更多,在平峰或者低峰时段,各个方向上车辆通过的概率均相应降低. 将一天的交通流量进行压缩至 3 h (10 800 s) 进行仿真实验. 在仿真过程中,每隔1 080 s对整个路网的车辆数进行统计,车辆数据如图 2 所示.

针对每个路口统计其整个仿真时段的车流情

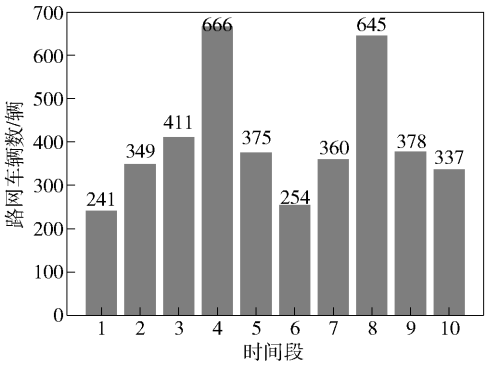

图 2　路网车辆数统计

况，如表 1 所示.

表 1　各路口车流量统计　　辆

路口编号	南北直行	南北左转	东西直行	东西左转
1	424	124	774	407
2	434	248	1196	422
3	427	119	735	381

仿真考虑四相位配时，为南北直行、南北左转、东西直行、东西左转，右转车道由于不会造成相位冲突，因此一直处于放行状态. 车辆随机进入交叉口的入口并提前选择所在车道，驶出路口时随机选择目标车道.

3）TSC-DDPG 算法参数设置

在 TSC-DDPG 算法中，Actor 网络和 Critic 网络均采用前馈全连接神经网络，网络参数、Sumtree、记忆库参数及其他相关参数选取参照文献[13-14]，并根据实验环境进行调整得到适合于此算法的最优参数. 具体的参数设置如表 2、表 3 所示.

表 2　神经网络参数

网络参数		层数	维度	激活函数
Actor 网络参数	输入层	1	24	NA
	隐藏层	2	200，200	Relu，Relu6
	输出层	1	6	Tanh
Critic 网络参数	输入层	1	30	NA
	隐藏层	2	200，200	Relu，Relu6
	输出层	1	1	NA

3.2　对比算法分析

1）固定配时

按照交通流设计的整体规律对每个路口确定固定配时，四个相位南北直行、南北左转、东西直行、东西左转的配时分别为 20 s，10 s，35 s，15 s.

表 3　其余参数设置

参数名	数值
Actor 网络学习率	0.000 5
Critic 网络学习率	0.000 5
折扣因子 γ	0.95
软交换系数 τ	0.99
记忆库大小	12 000
批处理大小	32
奖励值系数 w_1, w_2, w_3, w_4	−1，−1，−0.02，0.02
周期系数 $c_{\min}, c_{\max}$	0.6，2
固定周期时长 T	80

2）流量权重配时算法

流量权重配时算法是通过观测上一周期每个相位的车流量数据，然后遍历方案库中各相位权重，将两者进行对比找到欧式距离最小的方案作为下一周期的最优配时策略，该算法具备一定的自适应性，但由于方案库中方集数的局限，并非每一次都能找到最优策略. 通过给定相位数、周期时长和方案数，利用 *K*-Means 聚类法实现均匀分配时长，生成方案列表，并在此基础上加入固定配时方案数据，形成最终的流量权重配时方案库，如表 4 所示.

表 4　流量权重配时方案库　　s

方案号	相位一	相位二	相位三	相位四
0	20	10	35	15
1	10	10	50	10
2	10	50	10	10
3	10	10	10	50
4	10	25	35	10
5	10	10	30	30
6	10	35	10	25
7	30	30	10	10
8	30	10	25	15
9	30	15	10	25
10	50	10	10	10

3.3　实验评估与结果分析

排队长度、等待时间、车辆平均速度等指标能够很好地反映出交叉口的通行能力，在本小节，就 3 个路口在这几方面的性能进行评估分析.

对 3 个路口场景进行 400 回合训练，每一回合

的奖励值变化如图3所示. 最开始智能体处于探索阶段时,奖励值在 -35 000 ~ -25 000 之间,在120回合附近奖励值开始发生较为明显的改变,并在300回合后逐渐趋于稳定,最终稳定在 -6 500 左右. 接下来将训练好的TSC-DDPG算法与其余2种配时算法进行性能上的对比分析.

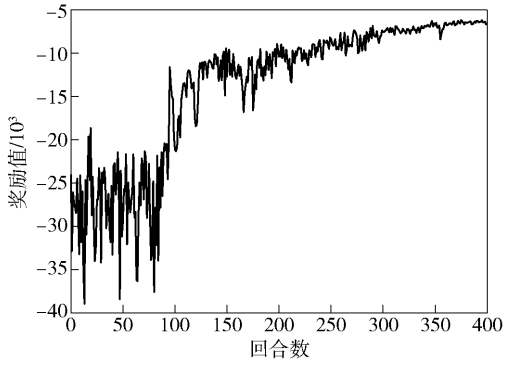

图3 训练过程中回合奖励值变化情况

图4~图6所示分别为多路口场景下在仿真时间段内,3个路口平均队列长度、车辆等待时间和车辆平均速度的变化情况,将10 800 s按照1 080 s为间隔分为10个时间段,针对每个时间段绘制其平均数据. 从图中可以直观地看出不同时间段交通流量的变化情况,由图4~图6可见,在队列长度、车辆等待时间和车辆平均速度3个方面,所提出的TSC-DDPG算法都明显优于其他2种配时算法.

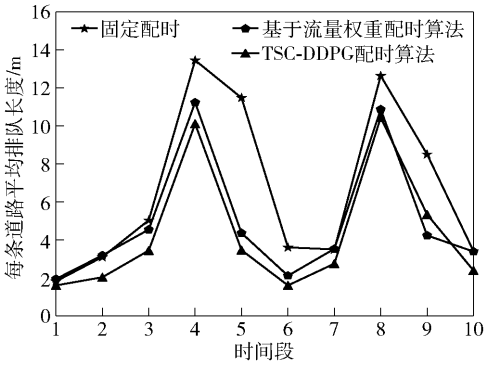

图4 排队长度对比

在使用TSC-DDPG算法的情况下,各条道路的平均排队长度为4.32 m,各条道路上车辆的平均等待时间为4.75 s,车辆平均速度为27.38 km/h. 与固定配时相比,排队长度缩短了35.63%,车辆等待时间减少了40.73%,车辆平均速度提升了8.60%;与流量权重配时算法相比,排队长度缩短了13.45%,车辆等待时间减少了17.79%,车辆平均速度提升了6.31%.

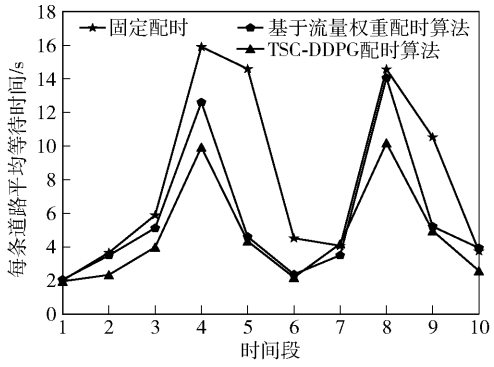

图5 车辆等待时间对比

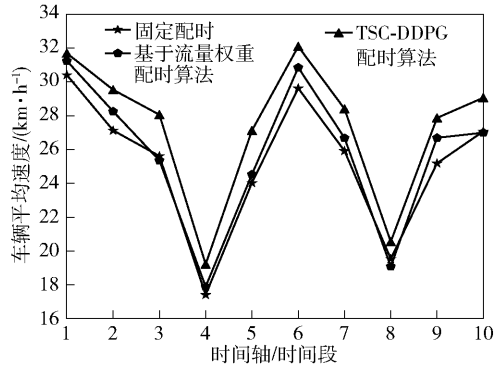

图6 车辆平均速度对比

4 结束语

提出了一种适用于多交叉口的交通信号控制算法TSC-DDPG,以应对日益严重的交通拥堵问题. 相比于传统配时算法,所提算法具有更强的灵活性和自适应性,能够进行高维度的交通特征提取,并能在连续动作空间选取合适的动作,奖励值的定义也充分考虑了路口的通行效率. 仿真结果显示,所提TSC-DDPG算法具有良好的收敛性,并且在队列长度、车辆等待时间、车辆平均速度等指标方面都明显优于其他配时算法.

参考文献:

[1] Araghi S, Khosravi A, Creighton D. A review on computational intelligence methods for controlling traffic signal timing[J]. Expert Systems with Applications, 2015, 42(3): 1538-1550.

[2] Sims A G, Finlay A B. SCATS, splits and offsets simplified (SOS)[J]. Australian Road Research, 1984, 12(4): 17-33.

[3] Lo H K. A reliability framework for traffic signal control

[J]. IEEE Transactions on Intelligent Transportation Systems, 2006, 7(2): 250-260.

[4] Abdulhai B, Pringle R, Karakoulas G J. Reinforcement learning for true adaptive traffic signal control[J]. Journal of Transportation Engineering, 2003, 129(3): 278-285.

[5] Genders W, Razavi S. Using a deep reinforcement learning agent for traffic signal control[EB/OL]. (2016-11-03)[2020-01-10]. https://arXiv.org/abs/1611.01142.

[6] Richter S, Aberdeen D, Yu J. Natural actor-critic for road traffic optimisation[C]//NIPS 2006, Proceedings of the 19th International Conference on Neural Information Processing Systems. Vancouver: MIT Press, 2006: 1169-1176.

[7] Pang Hali, Gao Weilong. Deep deterministic policy gradient for traffic signal control of single intersection[C]// 2019 Chinese Control and Decision Conference (CCDC). Nanchang: IEEE, 2019: 5861-5866.

[8] Casas N. Deep deterministic policy gradient for urban traffic light control[EB/OL]. (2017-08-02)[2020-01-10]. https://arXiv.org/abs/1703.09035v1.

[9] Lillicrap T P, Hunt J J, Pritzel A, et al. Continuous control with deep reinforcement learning[C]//ICLR 2016: International Conference on Learning Representations. San Juan: [s.n.], 2016: 1-14.

[10] Schaul T, Quan J, Antonoglou I, et al. Prioritized experience replay[C]//ICLR 2016: International Conference on Learning Representations. San Juan: [s.n.], 2016: 1-21.

[11] Wang Xiaoqiang, Ke Liangjun, Qiao Zhimin, et al. Large-scale traffic signal control using a novel multi-agent reinforcement learning[J]. IEEE Transactions on Cybernetics, 2021, 51(1): 174-187.

[12] Yang Shantian, Yang Bo, Wong Hau-San, et al. Cooperative traffic signal control using multi-step return and off-policy asynchronous advantage actor-critic graph algorithm[J]. Knowledge-Based Systems, 2019, 183: 1-19.

[13] Wei Hua, Zheng Guanjie, Yao Huaxiu, et al. Intellilight: a reinforcement learning approach for intelligent traffic light control[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM, 2018: 2496-2505.

[14] Zhang Zhi, Yang Jiachen, Zha Hongyuan. Integrating independent and centralized multi-agent reinforcement learning for traffic signal network optimization[C]//AAMAS 2020: Proceedings of the Nineteenth International Conference on Autonomous Agents and Multi-Agent Systems. Auckland: Springer, 2020: 2083-2085.