

文章编号: 1674-599X(2021)02-0098-07

速度协调的 Q 学习模型研究

刘元元¹, 卢守峰¹, 刘肖亮², 朱婷婷¹

(1. 长沙理工大学 交通运输工程学院, 湖南 长沙 410114;

2. 湖南联智科技股份有限公司, 湖南 长沙 410219)

摘 要: 为平滑高速公路瓶颈区域上、下游车流速度, 基于单位距离速度变化量和多项式 SG 卷积平滑处理的 2 个奖励函数, 分别建立了 2 个速度协调的 Q 学习模型。利用 Excel-VBA、VISSIM 和 MATLAB 的集成仿真平台, 对奖励函数的性能进行了仿真分析。研究表明: 基于 SG 卷积平滑的奖励函数, 可以有效地缓解瓶颈上游“走走停停”的现象, 减小速度的波动幅度。速度协调的 Q 学习模型可以根据交通状态实时选择最优的运行速度。

关键词: 速度协调; Q 学习算法; SG 卷积平滑; 集成仿真平台

中图分类号: U491.4

文献标志码: A

DOI:10.16544/j.cnki.cn43-1494/u.2021.02.015

Research on Q-Learning model of speed harmonization

LIU Yuan-yuan¹, LU Shou-feng¹, LIU Xiao-liang², ZHU Ting-ting¹

(1. Traffic and Transportation Engineering College, Changsha University of Science & Technology,

Changsha 410114, China; 2. Hunan Lianzhi Technology Co., Ltd., Changsha 410219, China)

Abstract: In order to optimize the speed of upstream and downstream traffic flow in the bottleneck area of expressway, two reward functions based on unit distance velocity variation and SG convolution smoothing were proposed, and two Q-learning models of speed harmonization were established. The integrated simulation platform combining Excel-VBA, VISSIM and MATLAB was used to simulate the reward functions. The results show that, the reward function based on SG convolution smoothing can effectively relieve the stop-and-go traffic on the upstream of the bottleneck. The fluctuation amplitude of speed was reduced. The Q-Learning model of speed-coordinated can suggest the optimal real time speed according to the traffic state.

Key words: speed harmonization; Q-Learning model; SG convolution smoothing; integrated simulation platform

小汽车拥有量日益骤增, 交通密度增大, 交通流处于不稳定状态, 加之车道变窄、合流分流、不良天气、急刹车等因素, 容易导致车辆加减速循环、走走停停, 不仅延误行程, 降低通行能力, 还增加燃油消耗, 产生大量的有害气体, 污染环境。因此, 如何有效缓解高速公路交通拥堵现象, 已成为许多学者深思的问题。

有学者提出高速公路速度控制方法, 即速度协

调(基于可变速度限制、车辆信息共享和自动车辆控制等策略, 平滑车辆在高速公路运行时的速度变化, 使得交通顺畅, 提高道路通行能力, 缓解交通拥堵)。国内外学者对该方向进行了大量研究。Li^[1-2]等人提出了基于描述函数的理论特性和强化学习的汽车跟随模型, 与传统的建模方法相比, 构建策略可以有效地抑制振荡幅度的发展, 从而减少油耗和废气排放。Ma^[3-4]等人提出了时间-距离线性速

收稿日期: 2020-10-30

作者简介: 刘元元(1995-), 女, 长沙理工大学硕士生。

度协调算法和 Bang-bang 反馈控制速度协调算法, 发现速度协调后的交通流轨迹可降低振荡行为。Ghiassi^[5-6]等人基于智能网联车辆 (connected automated vehicles, 简称为 CAV) 的轨迹平滑概念, 用 CAV 实时协调交通, 通过 CAV 和交通传感器提供的信息, 检测下游速度的下降和振荡, 预测其向上游的传播。该策略可以获得更平滑的轨迹, 提高交通流的总体平稳性。王正武^[7-8]等人构建了基于优化强制换道模型和考虑驾驶风格车辆的换道时间和距离预测模型, 可以较准确地预测和解释换道行为。Malikopoulos^[9]等人提出了可以实时实施的车速控制策略, 实现交通流中的速度协调, 使每辆车实现最佳的加速或减速。Park^[10]等人提出了车速控制策略, 采用最小化自动驾驶环境中的车祸风险, 通过车辆间风险分析, 执行车辆速度控制, 实现交通流中的速度协调。

1 Q 学习模型简介

强化学习可以分为无模型和模型化^[11]。1989 年 Watkins 提出的 Q 学习算法, 是一种基于值函数的典型无模型强化学习算法, 可用于解决马尔可夫决策。其原理是智能体根据当前状态, 选择某一动作作用于环境, 发生状态改变, 同时产生一个强化信号(奖或惩)反馈给智能系统, 智能系统再根据强化信号和当前环境状态, 选择下一个动作, 如此迭代循环, 直至目标获得最大奖赏, 其框架如图 1 所示。

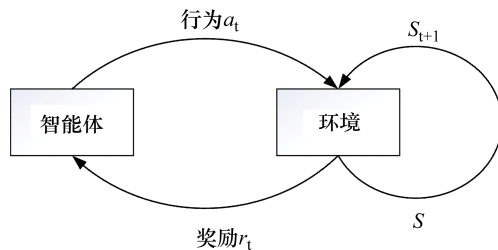


图 1 强化学习框架

Fig. 1 Reinforcement learning framework diagram

$Q(s,a)$ 是指某一时刻的状态 $s(s \in S)$, 采取动作 $a(a \in A)$ 能够获得回报的期望值。环境会根据智能体的动作反馈相应的回报, 可以用矩阵来存储 $Q(s,a)$ 值。然后根据 $Q(s,a)$ 值选取能够获得最大回报的动作。 $Q(s,a)$ 值的更新公式为:

$$Q'(s,a) = Q(s,a) + \partial \left[r + \gamma \cdot \max_{a' \in A} Q(s',a') - Q(s,a) \right]. \quad (1)$$

式中: s 为状态; a 为行为; ∂ 为学习速率, $\partial \in [0,1]$, 控制学习的速度, 取值越大收敛越快, 但易导致不成熟收敛; r 为奖励函数; γ 为折扣因子, $\gamma \in [0,1]$; A 为所有可供选择的行集合; $Q(s,a)$ 为更新前的 Q 矩阵中状态 s 采取动作 a 能够获得的 Q 值; $Q(s',a')$ 为下一状态 s' 时选取动作 a' 能够获得的 Q 值; $Q'(s,a)$ 为更新后的 Q 矩阵中状态 s 采取动作 a 时获得的 Q' 值。

2 速度协调的 Q 学习模型构建

状态、行为、奖赏是 Q 学习模型构建的 3 个主要元素。为实现速度协调, 引入 CAV 作为控制车辆。通过优化 CAV 的速度, 达到平滑速度波动的效果。针对速度协调的特点, 状态选取每个步长下游检测器测得的平均车速, 行为指 CAV 的建议运行速度, 构建了 2 个奖赏函数。

2.1 基于单位距离速度变化量的奖励函数

受线性速度协调算法^[3]的启发, 提出了基于单位距离速度变化量的奖励函数, 具体公式为式(2)~(5)。结合 Q 学习算法, 形成了基于单位距离速度变化量奖励函数的 Q 学习算法, 简称为 D-Q 算法。

$$r = |s_c - s(x,t)|. \quad (2)$$

$$s(x,t) = m(t)x + b(t). \quad (3)$$

$$m(t) = \frac{s_{D_d}(t) - s_{D_u}(t)}{\Delta x_{ud}}. \quad (4)$$

$$b(t) = s_{D_u}(t). \quad (5)$$

式中: r 为奖励函数; s_c 为 CAV 基于 D-Q 算法的建议速度; $s(x,t)$ 为基于单位距离变化量计算得到的速度; $m(t)$ 为单位距离下游检测器 D_d 与上游检测器 D_u 的速度变化量; x 为当实验启动后, 测试车所行驶的距离; $b(t)$ 为每个步长上游检测器 D_u 所测得的平均速度; $s_{D_d}(t)$ 为 t 时刻下游检测器测得的平均车速; $s_{D_u}(t)$ 为 t 时刻上游检测器测得的平均车速; Δx_{ud} 为第 u 组检测器和第 d 组检测器的距离差。

由式(3)可知, 奖励函数 r 的含义为智能网联测

试车的建议速度越接近单位距离速度变化量下的建议速度时, r 越小, 惩罚值也越小。

2.2 基于 SG 卷积平滑算法的奖励函数

VISSIM 仿真分析可知, 下游检测器测得的速度波动幅度大, 导致 $s(x, t)$ 的取值出现大幅的波动。因此, 提出基于多项式(savitsky-golay, 简称为 SG)卷积平滑算法的奖励函数(通过 SG 卷积平滑算法将下游检测器测得的速度进行平滑处理), 结合 Q 学习算法, 形成基于 SG 卷积平滑算法的奖励函数的 Q 学习算法, 简称 S-G 算法。

SG 卷积平滑在于矩阵算子的求解。假设平滑窗口的宽度为 $n=2q+1$, 各测量点 $x=(-q, -q+1, -q+2, \dots, q-1, q)$, 采用 $k-1$ 次多项式对窗口内的数据点进行拟合:

$$y = a_0 + a_1x + a_2x^2 + \dots + a_{k-1}x^{k-1}。 \quad (6)$$

式中: q 为坐标轴上的数据点, $q \in \mathbb{Z}$ 。

由式(6)可得到 n 个方程, 组成 k 元线性方程组, $n \geq k$ 时(一般选择 $n > k$), 方程组有解, 采用最小二乘法拟合确定参数 A 。

$$\begin{aligned} \begin{Bmatrix} y_{-q} \\ y_{-q+1} \\ \vdots \\ y_q \end{Bmatrix} &= \begin{Bmatrix} 1 & -q & \dots & (-q)^{k-1} \\ 1 & -q+1 & \dots & (-q+1)^{k-1} \\ \vdots & \vdots & & \vdots \\ 1 & q & \dots & (q)^{k-1} \end{Bmatrix} \\ &\quad \begin{Bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{k-1} \end{Bmatrix} + \begin{Bmatrix} e_{-q} \\ e_{-q+1} \\ \vdots \\ e_q \end{Bmatrix}。 \end{aligned} \quad (7)$$

式(7)为同时更新窗口 $n=2q+1$ 个 Y 值, 但实际上窗口向前逐一移动, 每次只更新中间点(q 点)。

式(7)用矩阵形式表示为:

$$\mathbf{Y}_{(2q+1) \times 1} = \mathbf{X}_{(2q+1) \times k} \mathbf{A}_{k \times 1} + \mathbf{E}_{(2q+1) \times 1}。 \quad (8)$$

其中, 乘以 1 之后代表矩阵形式。

则式(8)中 A 的最小二乘解为:

$$\tilde{\mathbf{A}} = (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T \cdot \mathbf{Y}。 \quad (9)$$

式中: Y 为平滑前的值。

则 Y 的滤波值为:

$$\tilde{\mathbf{Y}} = \mathbf{X} \cdot \tilde{\mathbf{A}} = \mathbf{X} \cdot (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T \cdot \mathbf{Y} = \mathbf{B} \cdot \mathbf{Y}。 \quad (10)$$

$$\mathbf{B} = \mathbf{X} \cdot (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T。 \quad (11)$$

经 SG 卷积平滑算法分析, 结合本试验实例, 取平滑窗口的宽度为 $q=4$, $k=3$, 则 $n=9$, 得:

$$\mathbf{X}_{(2q+1) \times k} = \mathbf{X}_{nk} = \mathbf{X}_{93} = \begin{Bmatrix} 1 & -4 & (-4)^2 \\ 1 & -3 & (-3)^2 \\ \vdots & \vdots & \vdots \\ 1 & 4 & (4)^2 \end{Bmatrix}。 \quad (12)$$

$$\begin{aligned} [s'_{D_d}(t)_1, s'_{D_d}(t)_2, \dots, s'_{D_d}(t)_9] &= \mathbf{X}_{93} \cdot (\mathbf{X}_{93}^T \cdot \mathbf{X}_{93})^{-1} \cdot \\ &\quad \mathbf{X}_{93}^T \cdot [s_{D_d}(t)_1, s_{D_d}(t)_2, \dots, s_{D_d}(t)_9]。 \end{aligned} \quad (13)$$

式中: $s'_{D_d}(t)_9$ 为下游检测器 D_d 经 SG 卷积平滑后的当前平均速度; $s'_{D_d}(t)_1, s'_{D_d}(t)_2, \dots, s'_{D_d}(t)_8$ 为经 SG 卷积平滑后的前 8 个步长的平均速度。

通过卷积平滑算法, 将下游检测器的速度 $s_{D_d}(t)$ 进行卷积平滑后得到 $s'_{D_d}(t)$, 则奖励函数为:

$$r = |s_c - s'(x, t)|。 \quad (14)$$

$$s'(x, t) = \frac{s'_{D_d}(t) - s_{D_u}(t)}{\Delta x_{du}} x + s_{D_u}(t)。 \quad (15)$$

2.3 基于集成仿真平台的在线仿真

利用集成仿真平台^[12]方法, 构建了速度协调在线仿真模型。基于 VISSIM 的 COM 接口与 Excel VBA、MATLAB 相结合, 对速度协调的 Q 学习模型进行仿真, 运算步骤为:

1) 建立 VISSIM 路网文件(net 文件)。包括建立道路网、设置车辆数、检测器的位置和个数、车辆组成、期望速度等, 设置相应的评估参数, 以便分析后续所需数据。

2) vba 程序中, 声明所使用的 VISSIM 的相关变量与对象, 例如 Vehicles(车辆对象), DataCollection(检测器对象)等; 定义相应的初始化函数, 例如 StartVISSIM(启动 VISSIM 软件)、ReadINPFile(读入 Net 文件)、main(主函数)等。

3) 主程序中, 通过 COM 接口的 Get Data CollectionByNumber 对象, 连接 VISSIM 文件中设置的检测器, 实现数据的实时读取; 通过 AddVehicleAtLinkCoordinate 对象, 将本文定义的自动网联测试车加入路网中, 实现对车辆的控制; 通过检测器的 GetResult 属性, 实现检测器数据的采集; 通过 ActiveWorkbook.Save 功能, 将 VISSIM 仿真的实时数据进行输出。

4) 获取最优 Q 矩阵。采用在线 Q 学习算法进

行 VISSIM 仿真学习, 通过多次运行在线 Q 学习仿真程序, 获得最优 Q 矩阵。首先, 在 Excel VBA 中执行 VISSIM 仿真程序, 以 2 s 为一个步长, 实时获取路网的交通状态、评价指标等信息。然后, 通过式(1), 更新 Q 矩阵, 当其达到终止条件时, 停止运行。单次 VISSIM 仿真停止运行条件为测试车到达下游检测器附近停止运行。如果本次运行的最终 Q 矩阵未达到收敛条件时, 将本次仿真的最终 Q 矩

阵作为下一次仿真时的初始 Q 矩阵, 直到达到收敛条件, 获得最优 Q 矩阵。

5) 执行最优 Q 矩阵方案, 获取最优建议速度。通过 Excel-VBA 程序, 控制 VISSIM 仿真运行。通过最优 Q 矩阵方案, VISSIM 仿真的下游检测器的实时状态, 获取最优建议速度, 并将建议速度应用到 VISSIM 仿真中, 检验速度协调效果。获取最优 Q 矩阵后的 VISSIM 仿真运行流程如图 2 所示。

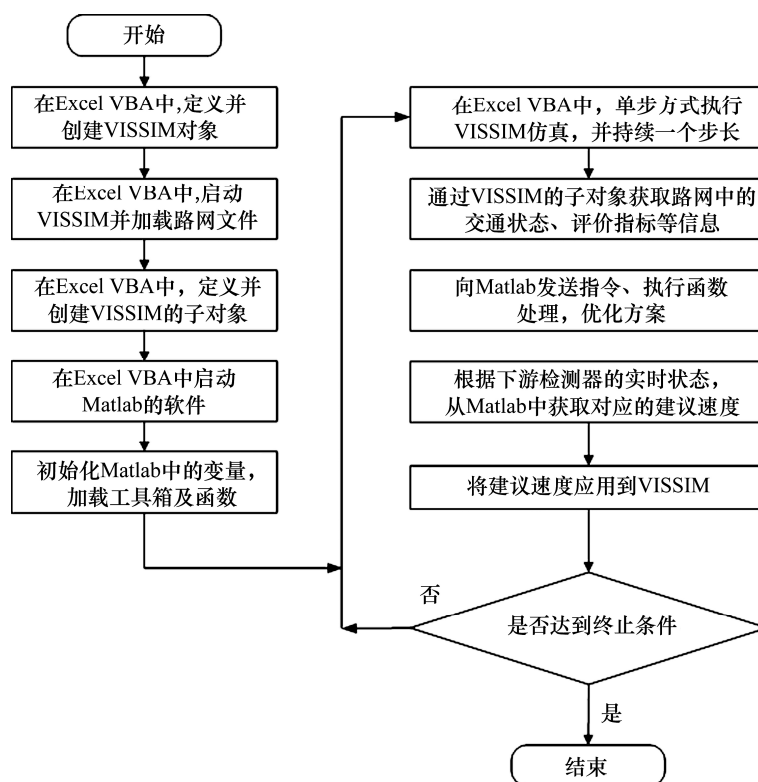


图2 集成仿真平台流程

Fig. 2 Flow chart of integrated simulation platform

3 算例分析

3.1 实验环境

以 VISSIM 为仿真平台, 构建一条三车道的高速公路试验路段。道路 5 km 处, 由三车道变成两车道, 形成物理瓶颈。其中, 仿真路段总长 6.68 km, 三车道路段为 6 km, 瓶颈路段长 600 m, 渐变段长度为 40 m。道路 1 km 处设置上游检测器 D_u , 4.9 km 处设置下游检测器 D_d , 用于检测某一时刻车辆通过该检测器时的平均速度。设置道路的交通流量为 3 780 辆/h, 饱和度为 0.7, 具体道路网的示意如图 3 所示。

当仿真初始化 10 min 后, 由于瓶颈的存在, 道路上出现明显拥堵。此时, 道路的 1 km 处(即上游检测器 D_u), 控制三辆 CAV 车辆(C_1 、 C_2 和 C_3)并排进入实验路段, 下游检测器 D_d 测得的平均速度为 Q 学习算法中的状态 s , 三辆 CAV 车辆采取的建议运行速度均为 Q 学习算法中动作 a , 以 $\Delta T=2$ s 为迭代步长, 每隔 2 s, 由下游检测器测得的平均车速。根据 Q 学习算法, 计算出此时上游三辆 CAV 车辆应采取的建议运行速度, 使当前时刻的奖励函数达到最大, 如此循环迭代, 优化 Q 值函数, 选取最优动作, 直到满足收敛条件, 结束实验。

3.2 状态划分

Q 学习算法的状态为下游检测器的速度值, 即

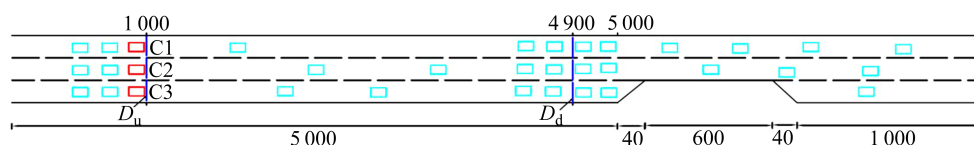


图 3 路段示意图 (单位: m)

Fig. 3 Schematic diagram of road section (unit: m)

下游检测器 D_d 在每个步长所测得的平均速度。通过对 VISSIM 仿真状态进行分析, 然后初始化, 下游检测器的速度值多分布在 20~50 km/h 之间, 因为状态数为一个确定的值或区间, 所以本研究将下游检测器的速度进行分段离散划分, 把下游检测器速度 [20, 50] km/h 划分为 15 个区间, 划分区间如图 4 所示。

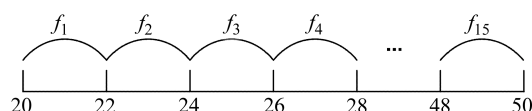


图 4 基于 D-Q 算法的状态划分

Fig. 4 State partition based on D-Q algorithm

划分后, 得到状态集为:

$S = \{s(i) = (20 + 2i, 20 + 2(i + 1)) | i = 0, 1, \dots, 14\}$, 如 $s(5) = (30, 32)$ km/h。

Q 学习算法行为是 CAV 的实时速度, 即测试车 C_1, C_2, C_3 的速度值, 速度区间为 [40, 120]。同时, 将测试车的速度进行分段离散划分, 将速度区间 [40, 120] 划分为 40 个区间, 划分区间如图 5 所示。

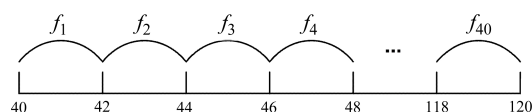


图 5 基于 SG-Q 算法的状态划分

Fig. 5 State partition based on SG-Q algorithm

划分后, 得到状态集为:

$A = \{a(i) = (40 + 2i, 40 + 2(i + 1)) | i = 0, 1, \dots, 39\}$, 如 $a(5) = (50, 52)$ 。Q 学习算法中的参数 $\partial_t = 0.1$, $\gamma = 0.8$ 。

3.3 试验结果分析

根据本研究提出的基于单位距离速度变化量的奖励函数和基于 SG 卷积平滑算法奖励函数的 Q 学习算法, 对 C_1, C_2, C_3 分别进行 2 种奖励函数 Q 学习算法的仿真试验, 并与“无控制情况”的结果做对比。

3.3.1 3 种不同情况下 C_1, C_2, C_3 的速度协调曲线

CAV 在无控制时, 速度协调曲线如图 6 所示。从图 6 中可以看出, 无控制情况下 3 辆控制车辆的初始速度较大, 经过 4.6 km 后, 控制车辆的速度骤降, 速度从 90~120 km/h 降到 0~40 km/h, 车辆的速度波动大, 在瓶颈处形成“交通震荡”, 符合无控制情况下的实际情况。基于单位距离速度变化量奖励函数的 Q 学习算法, 对控制车辆进行控制的位置-速度曲线如图 7 所示。与图 6 相比, 虽然速度呈整体下降趋势, 但是波动减小, 表明: 该算法起到了“速度协调”的作用。基于 SG 卷积平滑算法, 奖励函数的 Q 学习算法对控制车辆进行控制的位置-速度曲线如图 8 所示。从图 8 中可以看出, 其速度曲线比图 6、7 中的曲线更为平滑, 速度波动幅度更小, 有较好的速度协调效果。

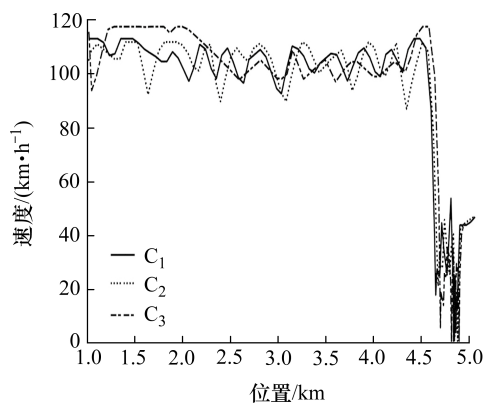


图 6 无控制时速度协调曲线

Fig. 6 Speed harmonization curve without control

3.3.2 三种不同情况下的 C_2 的速度协调曲线

由于 C_1, C_2, C_3 的速度大致相同, 以 C_2 为例, 分析无控制、基于单位距离速度变化量奖励函数的 Q 学习算法、基于 SG 卷积平滑算法奖励函数的 Q 学习算法 3 种不同情况下的 C_2 速度变化, 如图 9 所示。

从图 9 中可以看出, C_2 车辆从起点 1 km 到 5 km 处的速度变化很明显。 C_2 的初始速度很大, 达 110 km/h。由于处于无控制情况下, C_2 无法知道路段下

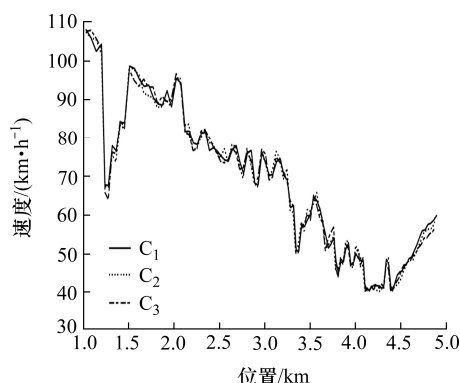


图7 基于D-Q算法的速度协调曲线

Fig. 7 Speed harmonization curve based on D-Q algorithm

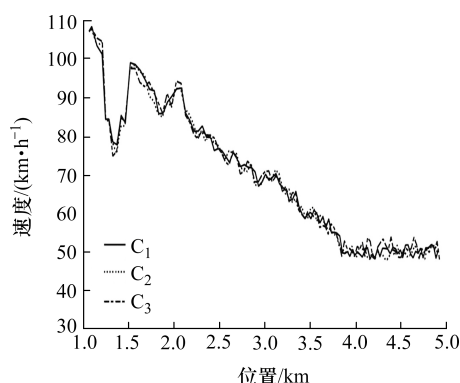
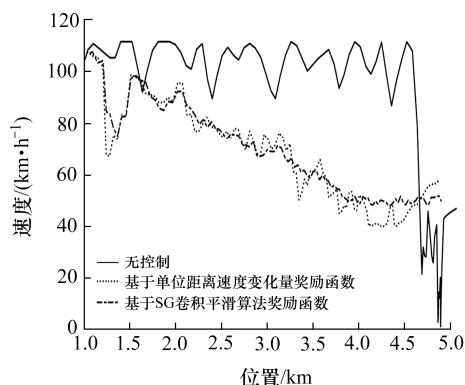


图8 基于SG-Q的速度协调曲线

Fig. 8 Speed harmonization curve based on SG-Q algorithm

图9 C₂速度协调效果对比Fig. 9 Speed harmonization effect comparison chart of the C₂

游的道路状况,影响C₂车速的主要因素是C₂周围的车辆情况,而不是道路下游的车辆情况,导致C₂前期一直处于高速状态,直到经过4.6 km后接近瓶颈,从110 km/h骤降到20 km/h,速度变化波动大,在此处产生典型的交通震荡。

从图9中还可以看出,基于单位距离速度变化量奖励函数的Q学习算法,对C₂进行速度控制达到速度协调效果。由于引入了单位距离速度变化量奖励函数的Q学习算法,系统每间隔2 s,根据道

路下游检测器D_d测得平均速度。通过Q学习算法计算,此时C₂应该采取最优速度。C₂根据计算推荐的最优速度,每隔2 s更新速度,使得C₂的速度波动减小幅度下降,而不是骤降,减小了“交通震荡”。与无控制情况相比,基于单位距离速度变化量奖励函数的Q学习算法,更有效地协调了C₂的速度。

由于基于单位距离速度变化量奖励函数的Q学习算法的速度协调效果曲线中,道路下游检测器D_d测得的平均波动较大。为使速度波动更小,速度曲线更平滑,引入基于SG卷积平滑算法奖励函数的Q学习算法对道路下游检测器D_d进行速度控制,在基于单位距离速度变化量奖励函数基础上,得到平滑下游检测器所测得的平均速度。表明:与基于单位距离速度变化量奖励函数的Q学习算法相比,基于SG卷积平滑算法奖励函数的Q学习算法的速度协调效果曲线更平滑,C₂的速度波动更小,更有效地协调了C₂的速度变化。因此,基于SG卷积平滑算法奖励函数的Q学习算法对C₂的速度协调效果最好。

3.4 Q值收敛情况

以每个步长更新的Q矩阵与上一个步长更新的Q矩阵各个位置元素之差的和的绝对值U判断是否收敛,即 $U = \text{Sum}(\text{Sum}(\text{abs}(Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t))))$,要求更新前、后Q矩阵每个位置的元素值之差的绝对值连续700次均小于0.015,即从第一次收敛值 $U < 15 \times 40 \times 0.015 = 9$ 开始,连续迭代700次小于9,即为收敛。

基于D-Q算法、SG-Q算法的Q值收敛如图10、11所示。从图10中可以看出,基于单位距离速度

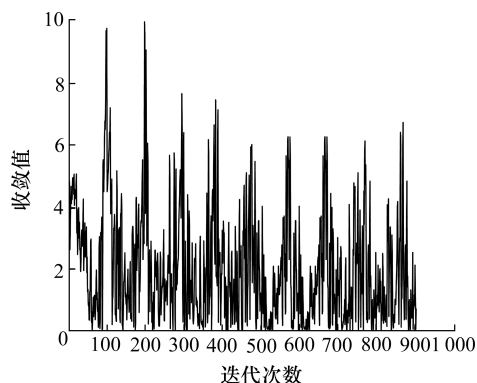


图10 基于D-Q算法的Q值收敛

Fig. 10 The Q value convergence graph based on D-Q algorithm

变化量奖励函数的 Q 学习算法, 迭代次数在 203 与 903 区间中。连续 700 次, 迭代收敛值小于 9。从图 11 中可以看出, 基于 SG 卷积平滑算法奖励函数的 Q 学习算法, 迭代次数在 98 与 798 区间中。连续 700 次, 迭代收敛值小于 9。因此, 可以看出前者迭代次数为 903 次, 后者迭代次数为 798 次, 2 个奖励函数均收敛, 且后者收敛速度更快。

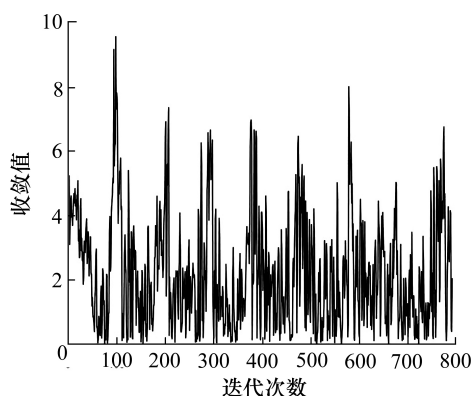


图 11 基于 SG-Q 算法的 Q 值收敛

Fig.11 The Q value convergence graph based on SG-Q algorithm

4 结论

基于 SG 卷积平滑算法奖励函数的 Q 学习算法, 速度波动最小, 速度曲线最平滑。同时, 从收敛情况可知, 基于单位距离速度变化量奖励函数收敛的稳定性更好, 基于 SG 卷积平滑算法奖励函数的收敛速度更快。因此, 提出的 D-Q 算法、SG-Q 算法 2 个奖励函数均能有效地协调高速公路上车辆的速度变化, 减小了“速度震荡”。当高速公路瓶颈处发生严重排队现象时, 上游车辆到达瓶颈处, 将减少“走走停停”的现象。

参考文献(References):

- [1] Li X P, Cui J X, An S, et al. Stop-and-go traffic analysis: Theoretical properties, environmental impacts and oscillation mitigation[J]. Transportation Research Part B: Methodological, 2014, 70: 319–339.
- [2] Qu X B, Yu Y, Zhou M F, et al. Jointly dampening traffic oscillations and improving energy consumption with electric, connected and automated vehicles: A reinforcement learning based approach[J]. Applied Energy, 2020, 257: 114030.
- [3] Ma J Q, Li X P, Shladover S, et al. Freeway speed harmonization[J]. IEEE Transactions on Intelligent Vehicles, 2016, 1(1): 78–89.
- [4] Yang H, Rakha H. Feedback control speed harmonization algorithm: Methodology and preliminary testing[J]. Transportation Research Part C: Emerging Technologies, 2017, 81: 209–226.
- [5] Ghiasi A, Ma J, Zhou F, et al. Speed harmonization algorithm using connected autonomous vehicles[C]// The 96th Annual Meeting of the Transportation Research Board, transportation Research Board, 2017.
- [6] Ghiasi A, Li X P, Ma J Q. A mixed traffic speed harmonization model with connected autonomous vehicles[J]. Transportation Research Part C: Emerging Technologies, 2019, 104: 210–233.
- [7] 王正武, 邹文竹, 郝威. 高速公路交通事故后基于优化的强制换道研究[J]. 交通科学与工程, 2020, 36(1): 87–92.(WANG Zheng-wu, ZOU Wen-zhu, HAO Wei. Research on mandatory lane change based on optimization after traffic accidents on the expressway[J]. Journal of Transport Science and Engineering, 2020, 36(1): 87–92.(in Chinese))
- [8] 刘思源, 喻伟, 刘洁莹, 等. 考虑驾驶风格的车辆换道行为及预测模型[J]. 长沙理工大学学报(自然科学版), 2019, 16(1): 28–35. (LIU Si-yuan, YU Wei, LIU Jie-ying, et al. Characteristics analysis and prediction model of lane changing behavior under different driving styles[J]. Journal of Changsha University of Science & Technology (Natural Science), 2019, 16(1): 28–35.(in Chinese))
- [9] Malikopoulos A A, Hong S, Park B B, et al. Optimal control for speed harmonization of automated vehicles[J]. IEEE Transactions on Intelligent Transportation Systems, 2019, 20(7): 2405–2417.
- [10] Park H, Oh C. A vehicle speed harmonization strategy for minimizing inter-vehicle crash risks[J]. Accident Analysis & Prevention, 2019, 128: 230–239.
- [11] 赵婷婷, 孔乐, 韩雅杰, 等. 模型化强化学习研究综述[J]. 计算机科学与探索, 2020, 14(6): 918–927.(ZHAO Ting-ting, KONG Le, HAN Ya-jie, et al. Review of model-based reinforcement learning[J]. Journal of Frontiers of Computer Science and Technology, 2020, 14(6): 918–927.(in Chinese))
- [12] 卢守峰, 韦钦平, 沈文, 等. 集成 VISSIM、Excel VBA 和 MATLAB 的仿真平台研究[J]. 交通运输系统工程与信息, 2012, 12(4): 43–48, 63. (LU Shou-feng, WEI Qin-ping, SHEN Wen, et al. Integrated simulation platform of VISSIM, excel VBA, MATLAB[J]. Journal of Transportation Systems Engineering and Information Technology, 2012, 12(4): 43–48, 63.(in Chinese))