耦合深度学习-运动学的自动驾驶一体化换道研究

熊明强^{1,2}, 谯杰¹, 王亮³, 夏芹¹, 江萌¹

(1.中国汽车工程研究院股份有限公司,重庆 401122;2.汽车噪声振动和安全技术国家重点实验室,重庆 401122;3.中国城市规划设计研究院西部分院,重庆 401122)

摘 要:现阶段主流算法都面临着数据量不足导致的模型无法完全应对所有场景,从而造成换道失败或者影响换道过程中的安全性。提出了一种新的耦合深度学习-运动学的自动驾驶换道一体化模型。该模型基于经典BP神经网络,对神经元中的激活函数进行替换,从而克服已有模型的不足。通过CarSim仿真来研究了本模型产生的自动驾驶车辆换道轨迹的可跟踪性,结果显示本模型可以更好地学到人类的安全换道行为,可推广使用。

关键词:深度学习;一体化换道轨迹规划;自动驾驶;安全性

中图分类号: U491.2 文献标志码: A DOI: 10.3969/j.issn.2095-1469.2022.06.12

Lane Changing Model Coupling Deep Learning and Kinematics for Autonomous Vehicles

XIONG Mingqiang^{1, 2}, QIAO Jie¹, WANG Liang³, XIA Qin¹, JIANG Meng¹

(1. China Automotive Engineering Research Institute Co., Ltd., Chongqing 401122, China;

2. State Key Laboratory of Vehicle NVH and Safety Technology, Chongqing 401122, China;

3. Western Branch of China Academy of Urban Planning and Design, Chongqing 401122, China)

Abstract: The mainstream algorithms are faced with the problem that the model could not cope with every driving scenarios for lane change due to insufficient data. This paper proposes a novel lane-changing model coupling deep learning and kinematics for automatic driving. Based on the classical BP neural network, the activation function in neurons is replaced to improve the model performance. The trajectories generated by this model are simulated by using CarSim. The results show that the smooth lane changes are achieved efficiently and safely. The trajectory output of this model can be widely used in automatic driving.

Keywords: deep learning; integrated lane-changing trajectory planning; automatic driving; safety

近年来,自动驾驶技术引起了世界各国的广泛 关注^[1-10],被认为是缓解交通拥堵,减少交通事故 和环境污染的重要技术^[1, 6, 11]。目前一些自动驾驶 已经进行了大规模的道路测试,比如谷歌自动驾驶

基金项目:国家重点基础研究发展计划(973计划)(2017YFB0102604)

参考文献引用格式:



熊明强,谯杰,王亮,等.耦合深度学习-运动学的自动驾驶一体化换道研究[J].汽车工程学报,2022,12(6):825-836. XIONG Mingqiang,QIAO Jie,WANG Liang, et al.Lane Changing Model Coupling Deep Learning and Kinematics for Autonomous Vehicles[J].Chinese Journal of Automotive Engineering,2022,12(6):825-836.(in Chinese)

收稿日期:2021-09-24 改稿日期:2021-12-07

和苹果自动驾驶。研究表明^[12],在如今的交通事 故中,有超过30%的道路交通事故是由不合理的换 道行为引起的。

目前基于机器学习的方法对自动换道轨迹进 行的研究还比较少[13-15],利用机器学习进行自动 驾驶车辆换道轨迹规划的模型需要经过大量已有 的换道数据训练来确定。YAO Wen 等^[13]将 k 最 近邻算法应用到换道轨迹规划中,但是该模型所 用到的数据较为有限且覆盖场景不足。在此基础 上, DING Chenxi 等^[14]构建了两层 BP 神经网络 对NGSIM 数据^[16]进行学习,从而对车辆换道进 行端到端的轨迹预测, 但是k最近邻算法和BP神 经网络对于换道这类时间序列数据处理能力有 限,通常是单一地复制已有的数据,并未对数据 前后帧的内部逻辑进行判断和处理^[17]。XIE Dongfan 等^[15]利用经典LSTM 神经网络预测自动 驾驶车辆换道轨迹, 该模型对换道轨迹曲线的学 习结果取得了较高的精度,但是没有对训练过程 中的一些不安全因素进行限制。考虑到现有基于 深度学习的自动驾驶车辆换道轨迹规划算法只是 单一地复制已有的换道轨迹,忽略了换道过程中 的动态变化,这样的模型对于换道条件过于苛 刻,无法用于现实环境。反观人类在对于新事物 的学习过程中,一方面由已有的理论经验对新事 物的反应进行指导,另一方面,面对新事物时, 人类也在不断的实践过程中形成自己的经验。本 文借鉴了人类对于新事物的学习过程,将所训练 的BP神经网络进行嵌套,从而建立一种可嵌入 的深度学习网络模型,该模型有效地继承了以往 数据的换道行为特征,同时能缩短对新数据的学 习过程。

1 嵌入神经网络换道模型

随着近年神经网络的不断发展,各种被用于不 同领域的神经网络被不断地提出和优化,但无论是 哪种神经网络,其方法大致上都在模仿人的学习过 程。人在最初对一个事物进行学习时,实际上是从 两个方面来进行学习的,一个方面是对该事物已经 形成的知识进行学习,另一个方面是通过不断的实 践进行学习,借鉴这种思路,本文提出了一种新的 神经网络模型,该神经网络模型旨在更加具体地模 仿人的学习过程,如图1所示。



图1 神经网络嵌入过程

该神经网络模型有两级,第1级横向上是一 个3层的全连接网络,第2层是嵌入层,嵌入层 里面每一个神经元都是一个单独的神经网络,第 2级是需要嵌入的神经网络。本文把这些嵌套的 神经网络暂定为BP神经网络,这些BP神经网络 分为两种:(1)根据需要已经训练好;(2)尚未 进行训练。由此实现在学习过程中既能够学到已 有的知识,又能通过尚未训练的神经网络进行学 习的目的。所以神经网络参数分为横向参数和纵 向参数。

1.1 横向参数

一般而言,本文设计的神经网络主要由以下3 个部分组成:输入层、嵌入层和输出层。

输入层:根据不同学习对象的实际情况,设定 神经元数量。输入的数据为交通环境数据,即上一 时刻换道车辆和换道车辆周边车辆的位置、速度、 加速度等数据。

嵌入层:由于嵌入层内部各个神经元可能用到 的输入变量长度不一样,所以本文设置了嵌入层, 嵌入层神经元的数量根据实际情况进行寻优及 测试。

输出层:根据实际问题需要设置数量,输出为 神经网络预测的下一时刻换道车辆位置和速度。

1.2 纵向参数

纵向上,各层神经元由两部分构成,一部分是

已经训练好的神经网络(图1主神经网络),以激 活函数的形式嵌入,但这些神经网络必须针对同样 的问题,利用不同数据进行训练后得到的,这些神 经网络在训练的过程中不需要进行参数调整,只需 要利用输入,得到输出结果即可。如有需要,可嵌 入没有训练过的神经网络,在主神经网络训练的过 程中进行同步训练。

1.3 数学模型

耦合的表现形式为两个或两个以上的系统构成 一个网络时,若其中某一子系统中数据发生变化, 能影响到其他子系统数据也发生类似的变化。其机 理也多种多样,本节主要采取数据耦合的方式。基 于深度学习的方式无法规避换道过程中的危险因 素,单一地复刻已有数据,且是黑盒模式导致无法 控制训练参数。基于规则的模型都是对显性数据进 行计算,无法研究交通环境中隐形数据对换道过程 的影响。本文将二者结合起来,试图寻找通过一种 机制让二者互补的方式,如图2所示。



图2 神经网络和规则的耦合过程

深度学习模块的训练是持续的,深度学习模块 在训练的过程中,运动学模块不断地向深度学习模 块传递安全性的数据,让深度学习也不断对安全性 数据进行学习,虽然无法控制训练过程中的参数, 但是这种数据耦合的方式可以在训练中潜移默化地 让模型更具安全性。

机器学习的过程,就是通过数据的前向传播求 得预测值和真实值的误差,用反向传播算法缩小误 差的一个过程。在进行前向传播之前,本文会根据 特定的问题和基于规则的换道数据和基于人类的换 道数据训练若干个 BP 神经网络 net₁, net₂, …net_n, 本文将这些神经网络以替换的方式,将经典的 BP 神经网络中的激活函数替换,原激活函数的输入变 为嵌入神经元的输入,原激活函数的输出变为嵌入 神经元的输出。 1.3.1 前向传播

如果一个经典 BP 神经网络有若干个输出神经 元,该网络每一次前向传播后,任意一个神经元*j* 的输入是:

$$S_j = \sum_{k=0}^{q} W_{jk} x_{k\circ} \tag{1}$$

第j个神经元的输出是:

$$y_j = f\left(S_j\right)_{\circ} \tag{2}$$

式中: f() 是激活函数,在本方法中激活函数为:

$$f_k(x) = \operatorname{net}_k(x)_\circ \tag{3}$$

式中:net为嵌入的BP经网络。

本文令隐藏层任意神经元h的输入为:

$$\alpha_h = \sum_{i=1}^n w_{mn} x_{i\circ} \tag{4}$$

式中: wm 是隐藏层与输入层的连接权重。

令输出层任意神经元j的输出为:

$$\beta_j = \sum_{h=1}^m w_{hj} S_{h^\circ} \tag{5}$$

式中: S_h为隐层神经元h的输出。

$$S_h = f(\alpha_h)_\circ \tag{6}$$

式中: f()为激活函数。

$$f(x) = \frac{1}{1 + e^{-x}}$$
(7)

最终得到嵌入神经网络的p个输出 $\hat{y}_{k} =$ $(\hat{y}_{2}^{k}, \hat{y}_{2}^{k}, ..., \hat{y}_{p}^{k}), 即:$

$$\hat{y}_k = f\left(\beta_j - \theta_j\right)_\circ \tag{8}$$

所以主神经网络神经元j的输入值为:

$$\rho_j = \sum_{h=1}^p w_{hj} \, \hat{y}_{h\circ} \tag{9}$$

主神经网络与真实值相比误差为:

$$E_{p} = \frac{1}{2} \sum_{j=1}^{m} \left\| t_{jp} - y_{jp} \right\|^{2}$$
(10)

1.3.2 反向传播

至此,一次前向传播结束,神经网络的最终目标是要使误差尽可能地小,所以需要将前向传播所得到的误差进行反向传播,训练得到相应的参数w,设第一层神经网络权值矩阵为W₁=

$$\begin{bmatrix} w_{41} & w_{42} & w_{4m} \\ w_{51} & w_{52} & w_{5m} \\ \vdots & \vdots & \cdots & \vdots \\ w_{n1} & w_{n2} & w_{nm} \end{bmatrix}, \quad 嵌入神经网络权重矩阵为 $W_q = \begin{bmatrix} w_{81} & w_{82} & w_{8m} \\ w_{91} & w_{92} & w_{9m} \\ \vdots & \vdots & \cdots & \vdots \\ w_{p1} & w_{p2} & w_{pm} \end{bmatrix}, \quad 嵌入层神经网络参数调整方 式为:$$$

$$\boldsymbol{W}_{q} = \boldsymbol{W}_{q} - \delta \frac{\partial \boldsymbol{E}_{p}}{\partial \boldsymbol{W}_{q}}^{\circ}$$
(11)

输出层第j个神经元所对应的误差为:

$$\delta_{yj} = -\frac{\partial E_p}{\partial S_j} \,\,. \tag{12}$$

以图1中权重矩阵 Wpm 为例进行说明:

$$\Delta w_{pm} = -\eta \frac{\partial E_k}{\partial w_{pm}}$$
(13)

$$\frac{\partial E_k}{\partial w_{pm}} = \frac{\partial E_k}{\partial \hat{y}_p^k} \cdot \frac{\partial \hat{y}_p^k}{\partial \beta_j} \cdot \frac{\partial \beta_j}{\partial w_{pm}} \circ$$
(14)

根据 β_i 的定义,显然有:

$$\frac{\partial \beta_j}{\partial w_{hj}} = S_{j\circ} \tag{15}$$

根据式(7)和式(8)可得:

$$\frac{\partial E_{k}}{\partial \hat{y}_{jm}^{k}} \cdot \frac{\partial \hat{y}_{jm}^{k}}{\partial \beta_{j}} = \hat{y}_{jm}^{k} \left(1 - \hat{y}_{jm}^{k}\right) \left(y_{jm}^{k} - \hat{y}_{jm}^{k}\right)_{\circ}$$
(16)

将式(14)和式(15)代入式(13),得到*W*_{pm}的更新公式为:

$$\Delta W_{pm} = \eta S_{j} \hat{y}_{pm}^{k} (1 - \hat{y}_{pm}^{k}) (y_{pm}^{k} - \hat{y}_{pm}^{k})_{o}$$
(17)

式中:学习率 $\eta \in (0, 1)_{\circ}$

将式(12)具体化为:

$$\partial_{yj} = \frac{\partial E_p}{\partial S_j} = (t_j - y_j) f_{yj}'(S_j)_{\circ}$$
(18)

式中: $f_{yj}'(S_j)$ 为输出层第j个神经元对应的 S_j 的传播方程导数,得到损失矩阵为:

$$\boldsymbol{W}_{z}^{T}\boldsymbol{\delta}_{y}\boldsymbol{f}_{z}^{'} = \begin{bmatrix} \left(\sum_{j=1}^{m} \delta_{yj} \boldsymbol{w}_{j1}\right) \boldsymbol{f}_{z1}^{'}\left(\boldsymbol{S}_{1}\right) \\ \cdots \\ \left(\sum_{j=1}^{m} \delta_{yj} \boldsymbol{w}_{jk}\right) \boldsymbol{f}_{zk}^{'}\left(\boldsymbol{S}_{k}\right) \\ \cdots \\ \left(\sum_{j=1}^{m} \delta_{yj} \boldsymbol{w}_{jp}\right) \boldsymbol{f}_{zp}^{'}\left(\boldsymbol{S}_{p}\right) \end{bmatrix} = \begin{bmatrix} \delta_{z1} \\ \cdots \\ \delta_{zk} \\ \cdots \\ \delta_{zp} \end{bmatrix} = \delta_{Z^{\circ}} \quad (19)$$

主神经网络权重的更新方式为:

$$W_{l} = W_{l} - \delta \frac{\partial E'}{\partial \operatorname{net}_{l}} \frac{\partial \operatorname{net}_{l}}{\partial W_{l}} \circ$$
(20)

式中: $\frac{\partial \operatorname{net}_{l}}{\partial W_{l}}$ 为嵌入神经网络中输出对输入的导数。

输出层神经元j对应的误差为:

$$\delta_j = -\frac{\partial E'}{\partial \operatorname{net}_l} \frac{\partial \operatorname{net}_l}{\partial S_j} \circ$$
(21)

以图1中权重W。为例进行说明:

$$\Delta w_p = -\eta \frac{\partial E'}{\partial w_p}$$
(22)

$$\frac{\partial E'}{\partial w_p} = \sum \frac{\partial E_k}{\partial \hat{y}_p^k} \cdot \frac{\partial \hat{y}_p^k}{\partial \operatorname{net}_p} \cdot \frac{\partial \operatorname{net}_p}{\partial \beta_j} \cdot \frac{\partial \beta_j}{\partial w_p}$$
(23)

根据α,的定义,显然有:

$$\frac{\partial \alpha_h}{\partial w_{hj}} = S_j \ . \tag{24}$$

根据式(7)和式(8)可得:

$$\frac{\partial E_k}{\partial \hat{y}_p^k} \cdot \frac{\partial \hat{y}_p^k}{\partial \alpha_h} = \sum \hat{y}_p^k \left(1 - \hat{y}_p^k\right) \left(y_p^k - \hat{y}_p^k\right)_\circ \qquad (25)$$

将式(23)和式(24)代人式(22),得到了 W_m的更新公式:

$$\Delta W_p = \sum \eta S_j \hat{y}_p^k \Big(1 - \hat{y}_p^k \Big) \Big(y_p^k - \hat{y}_p^k \Big) \,. \tag{26}$$

式中: 学习率η∈(0, 1)。

得到主神经网络的误差项为:

$$\partial_{p} = \frac{\partial E_{p}}{\partial S_{j}} = (t_{j} - y_{j}) f_{p}'(S_{j}) \,. \tag{27}$$

隐藏层所对应的误差矩阵为:

$$\boldsymbol{W}_{h}^{T}\boldsymbol{\delta}_{h}\boldsymbol{f}_{h}^{\prime} = \begin{bmatrix} \left(\sum_{j=1}^{n} \delta_{yj} W_{j1}\right) f_{m1}^{\prime} \left(S_{1}\right) \\ \cdots \\ \left(\sum_{j=1}^{n} \delta_{yj} W_{jk}\right) f_{mk}^{\prime} \left(S_{k}\right) \\ \cdots \\ \left(\sum_{j=1}^{n} \delta_{yj} W_{jn}\right) f_{mn}^{\prime} \left(S_{n}\right) \end{bmatrix} = \begin{bmatrix} \delta_{x1} \\ \cdots \\ \delta_{xk} \\ \cdots \\ \delta_{xm} \end{bmatrix} = \delta_{H^{\circ}} \quad (28)$$

得到损失矩阵之后,需要调整各层神经元的权 重矩阵:

$$\boldsymbol{W}_{o} = \boldsymbol{W} + \eta \boldsymbol{\delta}_{Z} \boldsymbol{Z}^{T}_{\circ} \tag{29}$$

$$\boldsymbol{W}_{i} = \boldsymbol{W} + \eta \delta_{H} \boldsymbol{X}^{T}_{\circ} \qquad (30)$$

1.3.3 最优轨迹模型

自动驾驶车辆在换道过程中,会根据实时交通 环境进行轨迹规划,每个时间步长内会为车辆规划 出最优的换道轨迹,由于基于时间多项式的换道轨 迹算法计算较为复杂,可能存在较大延迟,本文选 择形式较为简单的3次多项式作为基于规则的轨迹 规划算法,其表达式为:

 $y_n(x_n) = a_0 + a_1 x_n + a_2 x_n^2 + a_3 x_n^3$ 。 (31) 式中: $a_0 \, x_1 \, x_2 \, x_3$ 均为待确定的参数; x_n 为车辆 n的纵向位置; y_n 为车辆n的横向位置。确定各项参 数有:

$$y_{n}(x_{n}) = \tan \theta_{i}x_{n} + \frac{3y_{n}^{f} - 2x_{n}^{f}\tan \theta_{i}}{\left(x_{n}^{f}\right)^{2}}x_{n}^{2} + \frac{x_{n}^{f}\tan \theta_{i} - 2y_{n}^{f}}{\left(x_{n}^{f}\right)^{3}}x_{n}^{3} \circ$$
(32)

式中: θ_i 为规划步长起点的航向角; y'_n 为终点横向 坐标,均为已知量。

1.3.4 避障算法

在图3中,换道车辆为主车辆SV,当前车道前 车为PV,以及目标车道前车LV和目标车道后车 FV。其中,穿过车道变化的虚线是车辆SV的换道 轨迹,包括换道准备OP和换道执行PD两个阶段, 车辆SV在O点开始换道准备,在P点完成换道准 备开始换道执行,在D点完成换道。若定义换道执 行开始点为坐标原点,那么换道执行开始点P的位 置坐标为(0,0),换道执行结束点的坐标为 (*x_f*,*y_f*)。车辆SV在换道准备过程中受到PV的影 响,当换道条件满足的时候进行换道执行,车辆从 当前车道中心线移动到目标车辆中心线,根据 Gipps模型^[18]的推演,在执行换道过程中受到车辆 LV和FV的影响,车辆SV在换道过程中主要受这3 辆车的影响。车辆在换道过程中,可能的冲突区域 如图3所示.

Gipps模型是领域内经典的车辆安全距离模型, 可以较好地拟合两辆跟驰车辆之间的行驶状态,但 没有考虑车辆车身长度。所以本文在经典Gipps模 型的基础上加入车身长度进行改进,作为3次轨迹 曲线的约束条件。

车辆换道过程中受到周围车辆实时的影响,为 了保证换道过程的安全性,需要实时检测周围车辆 的行驶状态,并对周围车辆的行驶状态进行预测。





$$\operatorname{gap}_{\operatorname{safe}}^{n-1} = \frac{v_{n-1}^2}{2b_{n-1}} + \frac{\tau}{2} \left(2v(t) - a\tau \right) - \frac{v^2(t)}{2b}_{\circ} \quad (37)$$

式中: l,,,为前车长度; l为后车长度。

由此计算公式,得到SV与其他3辆车的安全 距离为:

$$g_{\text{safe}}^{p} = x^{p}(t) - x(t) - \frac{(l^{p} + l)}{2} = \frac{(v^{p}(t))^{2}}{2b^{p}} + \tau v(t) - \frac{(v(t))^{2}}{2b} \circ$$
(38)

$$g_{\text{safe}}^{1} = x^{1}(t) - x(t) - \frac{(l^{1} + l)}{2} = \frac{(v^{1}(t))^{2}}{2b^{l}} + \tau v(t) - \frac{(v(t))^{2}}{2b^{l}} + \frac{(v(t))^{2}}{2b^{l}} + \tau v(t) - \frac{(v(t))^{2}}{2b^{l}} + \frac$$

$$\frac{(v(t))^2}{2b}^{\circ}$$
(39)

$$g_{\text{safe}}^{f} = x(t) - x^{l}(t) - \frac{(l^{t} + l)}{2} = \frac{(v(t))^{2}}{2b} + \tau v^{f}(t) - \frac{(v^{f}(t))^{2}}{2b} \circ$$

$$(40)$$

式中: x^{p} (t_{stop})、x (t_{stop})分别为车辆 PV 和车辆 SV 紧急制动时的纵向坐标; x^{p} (t)、x (t)分别为紧急 制动开始时车辆 PV 和车辆 SV 的纵向位置坐标; v^{p} (t)、v (t)分别为车辆 PV 和车辆 SV 的速度; b^{p} 、 b分别为车辆 PV 和车辆 SV 的最大减速度; τ 为车辆 的反应时间; x^{l} (t)为车辆 LV 的纵向位置坐标; v^{l} (t)为车辆 LV 的速度; b^{l} 为车辆 LV 的最大减速度; l^{l} 为车辆 LV 的车身长度; l^{l} 为车辆 FV 的车身长度; x^{r} (t)为车辆 FV 的纵向位置坐标; v^{l} (t)为车辆 FV 的速度; b^{l} 为车辆 FV 的最大减速度。

车辆换道的安全性与车辆速度也有关,本文根

$$x_{j}(t) = x_{j}(t_{0}) + v_{j}t$$

s.t. $j = n - 1, n - 2, n + 2^{\circ}$ (33)

式中: j为编号为n-1的车辆; $x_j(t_0)$ 为 t_0 时刻车辆 j的纵向位置; v_j 为车辆j的速度; $x_j(t)$ 为t时刻车 辆j的纵向位置。

Gipps模型求解的是前车紧急停车时,后车在 经过反应时间τ后也采取紧急停车,恰好不追尾所 采取的速度。在经典的Gipps模型中,车身的长度 没有被考虑到,也没有考虑到实际跟驰过程中前后 两车的时变速度,本文对经典Gipps模型进行改进, 如图4所示。

后车紧急制动距离为:

$$x_{n-1}^{\text{stop}} = x_{n-1}(t) - \frac{v_{n-1}^2(t)}{2b_{n-1}} \,. \tag{34}$$

$$x_{n}^{\text{stop}} = x_{n}(t) + \frac{\tau}{2} \left(2v_{n}(t) - a\tau \right) - \frac{v_{n}^{2}(t)}{2b_{n}}$$
(35)

式中: x_{n-1}^{stop} 为前车从采取制动动作到车辆停止期间 行驶过的距离; x_n^{stop} 为后车从采取制动动作到车辆 停止期间行驶过的距离; x_{n-1} (t) 为前车在t时刻的 位置; x (t) 为后车在t时刻的位置; v_{n-1} (t) 为前 车在t时刻的速度; v (t) 为后车在t时刻的速度; b_{n-1} 为前车最大制动加速度; b为后车最大制动加速 度; a为后车实时加速度; τ 为后车的反应时间。

后车采取制动动作后,两车刚好不相撞所需的 距离为gapⁿ⁻¹,gapⁿ⁻¹就是Gipps最小安全距离。

$$x_{n-1}^{\text{stop}} - \frac{(l_{n-1}+l)}{2} - x_n^{\text{stop}} \ge 0 \quad . \tag{36}$$





据改进的 Gipps 安全模型计算车辆的安全速度。由 式(37)的最小安全间隙 gapⁿ⁻¹,推算出车辆 n的 安全速度范围。车辆 n 受到车辆 n-1影响的最大安 全速度 $v_n^{n-1}(t)$ 为:

$$\sqrt{b^{2}\tau^{2} + 2b\left(x(t) + \frac{\left(l_{n-1} + l\right)}{2} - x_{n-1}(t) + \frac{v_{n-1}^{2}}{2b_{n-1}} - \frac{a\tau^{2}}{2}\right)}_{\circ}$$
(41)

根据 g^p_{safe} 可以推算出车辆在换道准备的最大安 全速度 v^p_{safe},由下式计算得到:

 $v_{\text{safe}}^p = \tau b +$

 $v_n^{n-1}(t) = \tau b +$

$$\sqrt{b^{2}\tau^{2} + 2b\left(x(t) + \left(l + l^{p}\right)/2 - x^{p}(t) + \left(v^{p}(t)\right)^{2}/2b^{p}\right)}_{\circ}$$
(42)

同理,根据 g_{safe}^{f} 和 g_{safe}^{l} 计算车辆SV在执行过程 中的最大安全速度 v_{safe}^{l} 和最小安全速度 v_{safe}^{f} : $v_{safe}^{l} = \tau b +$

$$\sqrt{b^{2}\tau^{2} + 2b\left(x + (l + l^{1})/2 - x(t)^{1} + (v(t)^{1})^{2}/2b^{1}\right)}$$

$$v_{\text{safe}}^{f} = \sqrt{2b\left(x - (l^{f} + l)/2 - x(t)^{f} - \tau v(t)^{f} + (v(t)^{f})^{2}/2b^{f}\right)}$$

$$\circ (44)$$

2 仿真验证

2.1 数据介绍

本模型使用的是美国联邦高速公路公布的 NGSIM (Next Generation Simulation)数据^[19]。 NGSIM数据依据其真实性、高可靠性和高精度的 特点,广泛用于自动驾驶的各类应用和交通研究领域,是目前行业内较为权威的车辆行驶数据。 NGSIM数据是使用摄像头来搜集的交通信息,其 中包括车辆的位置、速度、车型、车道等,时间的 分辨率为0.1 s。数据采集选用的两段高速公路路段 如图5所示。

2.2 训练结果

根据Kolmogorov^[20]定理,本文构建的网络为 3层神经网络,包括一个输入层、一个隐藏层以及 一个输出层,隐藏层节点个数*l*是根据输出层节点 个数决定的,通过经验公式确定*l*的值,其中经验 公式为:

$$l = \sqrt{n + m}_{\circ} \tag{45}$$

式中: *n*为输入层神经元的数量; *m*为输出层神经 元的数量。

本文在以上定理的基础上,拟用典型的非线性 函数—对数函数对神经网络进行参数调优,对在以 e为底x的对数函数,将嵌入神经网络前后拟合的 结果进行对比。本次试验使用的传统 BP 神经网络 的参数是3层神经网络,其中输入神经元和输出神 经元数量都是1,隐层神经元设置为90个,而本模 型提出的神经网络第一级的参数和传统 BP 神经网 络完全相同,第2级神经网络选择了一个已经对目 标函数训练好了之后的4层神经网络,其中输入层 和输出层神经元均为1个,两个隐层神经元均设置 成90个。图6是传统 BP 神经网络和本文提出的神 经网络对log(x)的学习结果,可以看出,在训练



(a) I-80 高速公路



图 5 NGSIM 数据收集场景图

次数明显较少的情况下,嵌入神经网络后在精度方 面明显优于传统神经网络。



神经网络参数调整好后,对基于耦合深度学 习-运动学的自动驾驶一体化换道模型进行训练和 测试,本节共选用了100辆车的换道数据,其中70 组数据作为训练集,剩余30组数据作为验证数据 集。如图7所示,图中蓝色曲线表示本模型训练误 差随着训练次数增加而变化的趋势,可知训练误差 随着训练次数快速下降,训练超过20次时,损失值 稳定并且趋于收敛。考虑时间成本和梯度下降所需 的时间,将搭建的神经网络训练次数设置为25次。

图 8 为本模型预测的自动驾驶换道轨迹规划曲 线,其中黄色曲线为原始轨迹曲线,蓝色曲线为本



模型训练之后得到的学习结果。从图中细节可知, 相较于车辆启动换道阶段,利用本模型可以更快地 启动换道,其原因主要是机器决策所需的反应时间 比人类决策所需时间缩短了;在换道过程中间阶 段,车辆稳定地发生横向和纵向位移,总趋势同人 类换道一致;在换道结束阶段,本模型更早地完成 了换道,其原因也是在保证与后车安全性的前提 下,及早完成换道,增大与前车的距离。由此可证 明本模型可以规划出平滑的轨迹曲线,且由于缩短 了反应时间和人为差异等因素,使换道过程变得更 可控、更高效。

2.3 误差统计

为了直观地体现训练数据和原始数据的区别, 本节摘抄部分名义值和预测值数据,如图9和表1~3 所示。

32号车



图8 车辆成功换道场景仿真

| | 名义值/m | 预测值/m | |
|------|-------------|---------------|--|
| | 0 | 0 | |
| | 0.812 902 0 | 0.455 108 643 | |
| | 1.630 375 0 | 0.918 304 443 | |
| 32号车 | 2.445 106 0 | 1.384 490 967 | |
| | 3.043 072 0 | 1.850 708 008 | |
| | 4.019 702 0 | 2.716 009 521 | |
| | 4.780 483 2 | 3.780 700 684 | |

表1 32号车横向位移对比表

表2 32号车纵向位移对比表

| | 名义值/m | 预测值/m |
|------|-----------|---------------|
| 32号车 | 0.090 526 | 0 |
| | 0.089 002 | 0.026 999 474 |
| | 0.087 173 | 0.071 999 550 |
| | 0.085 649 | 0.119 699 478 |
| | 0.083 820 | 0.164 699 554 |
| | 0.082 296 | 0.209 699 631 |
| | 0.081 077 | 0.254 699 707 |

| 名义值/ (m·s ⁻¹) | 预测值 (m·s ⁻¹) |
|---------------------------|--------------------------|
| 8.138 160 | 8.415 |
| 8.135 112 | 8.262 |
| 8.110 728 | 8.358 |
| | |

表3 32号车速度对比表

8.019 288

7.872 984

7.729 728

7.638 288

本节使用了 70 组换道数据进行训练和验证, 拟用平均绝对误差(Mean Absolute Error, MAE) 以及平均绝对相对误差(Root Mean Squared Error, MARE)这两个统计学中常用的统计量对模型进行 误差统计。

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |d_{r,i} - d_{s,i}|_{\circ}$$
(46)

MARE =
$$\frac{1}{N} \sum_{i=1}^{N} \frac{|d_{r,i} - d_{s,i}|}{d_{r,i}}$$
 (47)

式中: N为测试数据样本数; d_{r.};为第*i*辆车的名义 值; d_{s.i}为第*i*辆车的预测值。

针对实际情况,做如下定义:MAE_{real}和 MARE_{real}分别表示预测值与名义值的平均绝对误差 和平均绝对相对误差,这两个统计量可以表征实际 值和预测值的误差。

本节对基于嵌入神经网络训练之后的一万余条 预测数据和原始数据进行比较,结果见表4。

| IDO | XO | YO | V1 | X1 | A1 | H1 | V2 | X2 |
|-----|---------|--------|--------|---------|---------|--------|--------|---------|
| 32 | 653.840 | 65.413 | 18.430 | 745.381 | -11.200 | 91.541 | 24.910 | 615.723 |
| 32 | 656.341 | 65.424 | 16.770 | 747.130 | -11.200 | 90.789 | 25.080 | 618.241 |
| 32 | 658.814 | 65.470 | 14.960 | 748.721 | -11.200 | 89.907 | 25.110 | 620.764 |
| 32 | 661.244 | 65.530 | 13.780 | 750.139 | -7.570 | 88.895 | 25.040 | 623.271 |
| 32 | 663.652 | 65.577 | 13.510 | 751.414 | 2.510 | 87.762 | 25.020 | 625.766 |
| 32 | 666.067 | 65.591 | 13.840 | 752.743 | 6.640 | 86.676 | 25.030 | 628.268 |
| 32 | 668.510 | 65.587 | 14.330 | 754.175 | 3.660 | 85.665 | 24.960 | 630.782 |
| 32 | 670.966 | 65.593 | 14.550 | 755.656 | -0.270 | 84.690 | 24.690 | 633.289 |
| 32 | 673.416 | 65.599 | 14.500 | 757.116 | -0.870 | 83.700 | 24.260 | 635.748 |
| 32 | 675.866 | 65.604 | 14.490 | 758.551 | 0.500 | 82.685 | 23.910 | 638.132 |
| 32 | 678.319 | 65.610 | 14.740 | 759.982 | 3.780 | 81.663 | 23.900 | 640.476 |
| 32 | 680.772 | 65.620 | 15.380 | 761.442 | 8.840 | 80.670 | 24.260 | 642.856 |
| 32 | 683.218 | 65.626 | 16.370 | 763.010 | 11.200 | 79.792 | 24.710 | 645.315 |
| 32 | 685.646 | 65.607 | 17.300 | 764.719 | 8.310 | 79.073 | 24.900 | 647.837 |
| 32 | 688.048 | 65.559 | 17.640 | 766.549 | -0.860 | 78.501 | 24.670 | 650.355 |

图9 NGSIM原始数据

8.535

8.715

8.754

8.589

| 亦县 | 误差指标 | | | |
|------|---------------------|----------------------|--|--|
| 文里 | MAE _{real} | MARE _{real} | | |
| 横向位置 | 0.017 | 0.009 | | |
| 纵向位置 | 0.214 | 0.031 | | |

表4 误差对比表

2.4 基于CarSim的仿真验证

CarSim 是用于车辆动力学的专用仿真软件,该 软件可以对驾驶员、路面条件以及空气动力等方面 进行仿真输入,对车辆制动性、稳定性参数等进行 输出。CarSim可以方便灵活地定义试验环境和试验 过程^[21]。

将提出的深度学习模型输出的车辆轨迹数据作为CarSim输入,拟用该软件对模型轨迹进行可跟 踪性、平稳性等参数的评估。

本节对训练的70组车辆换道轨迹随机选择一 组输入到CarSim仿真软件中进行动力学仿真,图 10为车辆SV在成功换道场景下本模型预测模型的 仿真结果。图10a为车辆SV在成功换道场景下,本 模型输出轨迹的仿真结果,其中蓝色曲线为本模型 输出的轨迹,亦即目标轨迹,红色轨迹是CarSim 仿真条件下的跟踪轨迹,从图中可知,本模型输出 的轨迹在动力学仿真中变化平稳,可以和目标轨迹 保持较小误差的条件下被跟踪。图 10b 为车辆 SV 在成功换道场景下,本模型输出速度的仿真结果, 其中蓝色曲线为本模型输出的速度,即目标跟踪速 度,红色曲线是CarSim 仿真条件下的跟踪速度, 由图可知,本模型输出的速度在动力学仿真中在驻 点以及拐点处同目标速度有一定差异,但是同样较 为平滑,在可接受的范围内。图10c为换道过程中 车辆前轮转向角随时间变化情况,红色曲线代表了 车辆的右前轮转向角变化情况,蓝色曲线代表了车 辆的左前轮转向角变化情况,车辆在换道准备阶段 前轮转向角为0,在换道执行阶段,车辆前轮由0 逐渐增大到最大值后,逐渐减小至0,然后向相反 方向行驶至较大值后减小至0。此外,车辆左前轮 与右前轮的转向角存在差异,与车辆真实换道情况



相符合。图 10d 显示了车辆轮胎的侧向滑移角,4 个车轮的变化保持一致,变化的数值较小。

为了对比曲线的平滑程度,本节将原始数据中 的加速度值和神经网络预测结果进行对比,可见, 本模型预测加速度较真实平均值结果较小,结果 见表5。

表5 加速度对比表

| | 真实加速度/ (m·s ⁻²) | 神经网络预测结果加 速度/ (m·s ⁻²) |
|-----|-----------------------------|---------------------------------------|
| 最小值 | -3.413 760 | -3.154 20 |
| 最大值 | 3.315 420 | 2.985 71 |
| 平均值 | 0.277 503 | 0.229 10 |

3 结论

本文提出的自动驾驶领域提出一种耦合深度学

参考文献 (References)

- JACOBSTEIN N. Autonomous Vehicles: An Imperfect Path to Saving Millions of Lives [J]. Science Robotics, 2019,4(28):aaw8703.
- [2] LI Xiaopeng, GHIASI A, XU Zhigang, et al. A Piecewise Trajectory Optimization Model for Connected Automated Vehicles: Exact Optimization Algorithm and Queue Propagation Analysis [J]. Transportation Research Part B: Methodological, 2018, 118: 429–456.
- [3] HAN Y J, AHN S Y. Stochastic Modeling of Breakdown at Freeway Merge Bottleneck and Traffic Control Method Using Connected Automated Vehicle [J]. Transportation Research Part B: Methodological, 2018, 107: 146–166.
- [4] AMIR G, OMAR H, QIAN Zhen, et al. A Mixed Traffic Capacity Analysis and Lane Management Model for Connected Automated Vehicles: A Markov Chain Method
 [J]. Transportation Research Part B: Methodological, 2017, 106: 266–292.
- [5] DE ALMEIDA C G H, VAN AREM B. Solving the User Optimum Privately Owned Automated Vehicles Assignment Problem (UO-POAVAP) : A Model to Explore the Impacts of Self-Driving Vehicles on Urban Mobility[J].Transportation Research Part B:Methodological, 2016, 87: 64–88.
- [6] RAHMAN M S, ABDEL-ATY M, LEE J Y, et al.

习-运动学的自动驾驶一体化换道算法,建立了嵌 入神经网络模型,并综合考虑了目前主流换道模型 的不足,从而引入基于规则的训练模型,对于有限 数据下神经网络如何能适应和学习到更多驾驶技巧 的问题提出了解决思路。得到以下主要结论:

(1)利用本模型可以规划出一条合适的自动驾驶车辆换道轨迹,且能保证换道车辆能够更安全、 更舒适地完成换道。

(2)仿真发现,构建的换道轨迹规划模型可以用于自动驾驶车辆换道的场景。车辆可以在有限的数据量条件下对新的交通场景做出合理的反应。

(3) CarSim的仿真显示,本章节提出的模型所 规划出的换道的轨迹和速度能够被自动驾驶车辆跟 踪,车辆行驶稳定性良好。

Safety Benefits of Arterials' Crash Risk Under Connected and Automated Vehicles[J]. Transportation Research Part C: Emerging Technologies, 2019, 100: 354–371.

- [7] LI Li, HUANG Wuling, LIU Yuehu, et al. Intelligence Testing for Autonomous Vehicles: A New Approach [J].
 IEEE Transactions on Intelligent Vehicles, 2016, 1(2): 158–166.
- [8] LYGEROS J, GODBOLE D N, SASTRY S. Verified Hybrid Controllers for Automated Vehicles [J]. IEEE Transactions on Automatic Control, 1998, 43 (4) : 522-539.
- [9] YANG Da, ZHENG Shiyu, WEN Cheng, et al. A Dynamic Lane-Changing Trajectory Planning Model for Automated Vehicles [J]. Transportation Research Part C: Emerging Technologies, 2018, 95: 228–247.
- [10] GLASER S, VANHOLME B, MAMMAR S, et al. Maneuver-Based Trajectory Planning for Highly Autonomous Vehicles on Real Road with Traffic and Driver Interaction[J]. IEEE Transactions on Intelligent Transportation Systems, 2010, 11(3): 589–606.
- [11] WINKLE T. Safety Benefits of Automated Vehicles: Extended Findings from Accident Research for Development, Validation and Testing[M]//Autonomous Driving: Technical, Legal and Social Aspects. Heidelberg:

836

Springer, 2016: 335-364.

- [12] DE DIEGO I M, SIORDIA O S, CRESPO R, et al. Analysis of Hands Activity for Automatic Driving Risk Detection[J]. Transportation Research Part C: Emerging Technologies, 2013, 26: 380–395.
- [13] YAO Wen, ZHAO Huijing, BONNIFAIT P, et al. Lane Change Trajectory Prediction by Using Recorded Human Driving Data [C]//2013 IEEE Intelligent Vehicles Symposium (IV), June 23–26, 2013, Gold Coast, OLD, Australia.Piscataway NJ:IEEE, c2013:430–436.
- [14] DING Chenxi, WANG Wuhong, WANG Xiao, et al. A Neural Network Model for Driver's Lane-Changing Trajectory Prediction in Urban Traffic Flow[J]. Mathematical Problems in Engineering, 2013, 2013:1–8.
- [15] XIE Dongfan, FANG Zhezhe, JIA Bin, et al. A Data-Driven Lane-Changing Model Based on Deep Learning
 [J].Transportation Research Part C: Emerging Technologies, 2019, 106: 41–60.
- [16] FHWA. Next Generation Simulation (NGSIM) [EB/ oL].(2010-11-02)https://ops.fhwa.dot.gov/trafficanalysis tools/ngsim.htm.

- [17] WANG Yuan, ZHANG Dongxiang, LIU Ying, et al. Enhancing Transportation Systems Via Deep Learning: A Survey [J]. Transportation Research Part C: Emerging Technologies, 2019,99:144–163.
- [18] WANG Jiangfeng, ZHANG Qian, ZHANG Zhiqi, et al. Structured Trajectory Planning of Collision-Free Lane Change Using the Vehicle-Driver Integration Data [J]. Science China Technological Sciences, 2016, 59 (5): 825-831.
- [19] YANG Da, ZHU Liling, YANG Fei, et al. Modeling and Analysis of Lateral Driver Behavior in Lane-Changing Execution [J]. Transportation Research Record: Journal of the Transportation Research Board, 2015, 2490: 127-137.
- [20] KŮRKOVÁ V. Kolmogorov's Theorem and Multilayer Neural Networks [J]. Neural Networks, 1992, 5 (3) : 501–506.
- [21] XU Liang, GUO Rui, LIU Xiao. Virtual Validation and Verification Method of Brake System Model [J]. Advanced Materials Research, 2014, 1056:177–181.

作者简介



熊明强 (1993-), 男, 重庆市人, 硕士, 工程师, 主要研究方向为自动驾驶换道轨迹规划及神经网络在车路协同中的应用。 Tel: 13617620897 E-mail: john4016@163.com