

控制与决策

Control and Decision

基于生存理论训练机器学习的智能驾驶路径生成方法

刘磊, 杨晔, 刘赛, 高岩, 王富正, 王亚刚

引用本文:

刘磊, 杨晔, 刘赛, 等. 基于生存理论训练机器学习的智能驾驶路径生成方法[J]. *控制与决策*, 2020, 35(10): 2433–2441.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2019.0059>

您可能感兴趣的其他文章

Articles you may be interested in

基于神经网络的多类别目标识别

Multi-category target recognition based on neural network

控制与决策. 2020, 35(8): 2037–2041 <https://doi.org/10.13195/j.kzyjc.2019.0577>

智能制造系统基于数据驱动的车间实时调度

Data-based real-time scheduling in smart manufacturing

控制与决策. 2020, 35(3): 523–535 <https://doi.org/10.13195/j.kzyjc.2018.0849>

一种基于Dyna-Q学习的旋翼无人机视觉伺服智能控制方法

A visual servo intelligent control method for rotor UAV based on Dyna-Q learning

控制与决策. 2019, 34(12): 2517–2526 <https://doi.org/10.13195/j.kzyjc.2018.0342>

基于虚拟领队的 uncertain 轮式移动机器人自适应编队控制

Adaptive control of wheeled mobile robot formation with uncertainties based on virtual leader

控制与决策. 2017, 32(7): 1203–1209 <https://doi.org/10.13195/j.kzyjc.2016.0481>

基于生物启发模型的AUV三维自主路径规划与安全避障算法

Algorithm for AUV's 3-D path planning and safe obstacle avoidance based on biological inspired model

控制与决策. 2015, 30(5): 798–806 <https://doi.org/10.13195/j.kzyjc.2014.0339>

基于生存理论训练机器学习的智能驾驶路径生成方法

刘磊^{1,2†}, 杨晔², 刘赛², 高岩¹, 王富正¹, 王亚刚²

(1. 上海理工大学管理学院, 上海 200093; 2. 上海理工大学光电学院, 上海 200093)

摘要: 机器学习技术广泛应用于车辆的智能驾驶, 其中模型训练是该技术的关键, 由于训练数据难以覆盖全部驾驶情况, 使得极端状态下基于机器学习的智能驾驶系统存在失效风险, 会造成重大交通事故. 生存理论应用于车辆的道路安全态势感知具有理论优势, 能客观地计算出车辆最大的高维生存空间, 但该理论迭代计算繁琐, 输出结果所需时间较长, 无法满足高速车辆的实时控制, 且生存核表面复杂, 智能驾驶系统难以直接使用, 需要将生存核转化为局部最优路径. 鉴于此, 设计一种基于生存理论的局部路径规划机器学习训练方法, 通过对多种机器学习方法的特点进行分析, 最终选定径向基神经网络来输出生存核中线投影. 通过对比两种网络训练数据的输出效果, 分析参数敏感性以及泛化能力, 论证所提出训练方法的合理性. 仿真实验表明, 所训练的机器学习模型可快速输出高精度、大裕度的道路优化路径, 即使使用简单的控制律也能实现无人车辆的大曲率转弯. 由于所提出机器学习方法的安全性具有理论保障, 又能大幅提升安全计算的实时性, 在智能驾驶领域拥有广阔的应用前景.

关键词: 智能驾驶; 路径规划; 生存理论; 径向基神经网络

中图分类号: TP273

文献标志码: A

DOI: 10.13195/j.kzyjc.2019.0059

引用格式: 刘磊, 杨晔, 刘赛, 等. 基于生存理论训练机器学习的智能驾驶路径生成方法[J]. 控制与决策, 2020, 35(10): 2433-2441.

Path generation method for intelligent driving based on machine learning trained by viability theory

LIU Lei^{1,2†}, YANG Ye², LIU Sai², GAO Yan¹, WANG Fu-zheng¹, WANG Ya-gang²

(1. School of Management, University of Shanghai for Science and Technology, Shanghai 200093, China; 2. School of Optical-electrical, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: Machine learning has been widely used in the area of intelligent driving. Model training is the key point for the application of machine learning. Due to the insufficient training data for all driving situations, serious traffic accidents would happen if the intelligent driving system fails under the extreme conditions. By applying the viability theory to the dynamic adaptive analysis of the vehicle, the maximum safety space in high-dimension can be objectively calculated, which reflects the special advantages of this theory in the field of vehicle's safety. However, due to the high complexity of computation and long analyzing time, the iterative algorithm of this theory cannot meet the real-time control requirements of high-speed vehicles. The surface of the viability kernel is also too complex to be used directly by the intelligent driving system, for which the viability kernel needs to be transformed into the optimal path of the road. Therefore, we design a training method for machine learning of local path planning by the viability theory. According to the character analysis of multiple machine learning, we finally select radial basis function (RBF) neural network to output the central line projection of the viability kernel. By comparing two training methods, analyzing the parameter sensitivity and generalization ability, the proposed training method is approved to be rational. Simulation experiments show that the data-driven model trained by our method can output the safety results quickly and precisely. Meanwhile, the control margin is large, so it can help to drive the unmanned vehicles in a high challenging environment by a simple control rule. Since the safety of the machine learning method can be guaranteed by the theory, and the computing speed is enhanced, the method has a wide application in the area of intelligent driving.

Keywords: intelligent driving; path planning; viability theory; radial basis function neural network

收稿日期: 2019-01-11; 修回日期: 2019-05-08.

基金项目: 上海市自然科学基金项目(17ZR1419000); 国家自然科学基金项目(61074087, 11502145, 61703277).

责任编辑: 侯忠生.

†通讯作者. E-mail: liulei@usst.edu.cn.

0 引言

当前人工智能、机器学习技术大规模地应用于各行各业,尤其是汽车领域的智能化技术正蓬勃发展,智能驾驶技术呈现出全面井喷的态势,其结果将对整个交通运输行业乃至人类社会造成积极深远的影响.现代汽车通过配装专业的传感器和电子控制单元来提升整车的安全性水平,例如刹车防抱死系统ABS^[1]、车辆稳定控制系统EPS^[2]、高级驾驶辅助系统ADAS(巡航控制,车道保持,侧风辅助等)^[3],随着视觉、雷达、地图技术的提高,现代车辆逐渐具备了道路场景理解能力,结合车辆内部的数据采集,做到了更加全面地内窥外视,如果控制器能有效处理内外数据,再实时计算安全驾驶策略,则对实现车辆的自动驾驶意义重大^[4].

智能驾驶系统的关键在于高速、全面、客观地评估车辆相对于道路的环境适应性,并根据评估结果给出最优的驾驶参考,后续无论是用于系统安全辅助,还是接管驾驶都能游刃有余.车辆的实时驾驶参考主要表现为局部路径规划,自动驾驶的本质是局部规划路径跟踪,可见局部路径规划是实现智能驾驶系统的核心科学问题,其定义为获取车辆从一个可行状态到下一个可行状态过渡的安全路径,并同时满足车辆动力学约束^[5].

学术界在路径规划领域已经积累了较多成果^[6-9],常用的方法包括人工势场法(artificial potential fields)、单元分解法(cell decomposition)、最优控制法(optimal control)、模型预测控制(MPC)等.人工势场法结构简单,但算法安全性和准确性取决于势场的人工定义,所以主观性较大,无法应用于高速驾驶场景^[10].快速扩展随机树(RRT)是一种广泛使用的单元分解算法,其在高维空间的搜索速度较慢,且对道路曲率敏感,当面对狭窄通道时,算法收敛速度慢^[11].最优控制算法能考虑车辆的运动学约束,但对模型非线性以及车辆动力学求解存在不足^[12].近年来,模型预测控制方法被广泛应用于局部路径规划,取得了较好的效果^[13],并在有限时间尺度上使动态车辆具备一定的道路环境适应力,所以在自动驾驶领域得到了广泛应用.但MPC难以在车辆状态的高维空间内全面评估所有驾驶行为的最优路径,另外,有限时域内的代价函数优化无法在理论上保证控制过程一直存在可行解,因此存在较大的理论风险.通过研究发现,采用生存理论进行局部道路分析具有独特的理论优势^[14],生存理论是Aubin^[15]首先提出并用于分析动态系统演化与环境适应性的专门理论,为稳定性分析、安全性设计等领域提供了新的方法,该理论可对系统高维状态进行全面安全分析.研究发现,使

用生存理论不需要传统的优化计算便能客观地给出最优规划路径,相比于MPC计算结果更加有效^[16].但是,生存理论计算需要大量迭代,其计算复杂度随车辆模型维度的增长呈指数增加^[17],所以应用生存理论虽然具有理论优势,但难以满足自动驾驶的实时性要求.

鉴于此,本文对生存理论信息进行有效处理,拟使用机器学习方法直接给出局部规划路径^[18],以期提升路径规划效率.目前成熟的机器学习算法有决策树算法、贝叶斯算法、支持向量机(SVM)、人工神经网络算法等.通过理论对比,首次采用人工神经网络方法对生存理论规划的安全路径进行模型学习,该方法将机器学习模型与理论分析计算的优势相结合,能高速地规划出当前驾驶状况下的最优路径,是一种灵活应用生存理论的有益尝试.

1 生存理论及其数据驱动模型训练

1.1 生存理论概述

考虑到道路生存性分析的相关研究较少,又是本文工作的基础,这里对之前的工作进行简要介绍^[15-16,19-21].考虑一般控制系统

$$\dot{x}(t) = f(x, u), u \in U. \quad (1)$$

其中: $x \in R^n$ 为系统状态, $u \in U \subset R^m$ 为输入控制; $f(x, u)$ 为 R^{m+n} 到 R^n 上的Lipschitzian函数.系统(1)对于紧凸集合 K 的生存性定义如下.

定义1^[15] 考虑集合 $K \subset R^n$,对于任意初始状态 $x_0 \in R^n$,使得 $x(t) \in K, \forall t \geq 0$,其中 $x(t)$ 是系统(1)的解,则称系统(1)在集合 K 中是生存的.

为判定系统的生存性需要定义集合的切锥.

定义2^[21] 令 $x \in K \subset R^n$,集合 K 在 x 处的切锥定义为

$$T_k(x) = \left\{ v \in R^n \mid \liminf_{t \rightarrow 0^+} \frac{1}{t} d_k(x + tv) = 0 \right\}, \quad (2)$$

其中 $d_k(y)$ 为点 $y \in R^n$ 到集合 K 的距离.

在切锥的基础上,系统生存域定义如下.

定理1^[20] 闭集 $J \subset K$ 关于系统(1)是生存的充要条件是对于任意 $x \in J$,有 $(\bigcup_{u \in U} f(x, u)) \cap T_J(x) \neq \emptyset$,其中 \emptyset 为空集.集合 J 可看作系统(1)在 K 中的一个生存域.环境集合 K 可能存在多个生存域,其中最大的生存域称为生存核,用 $\text{viab}(K)$ 表示.

定义3^[14] 系统相对集合 K 的生存核为

$$\text{viab}(K) = \{x_0 \in K \mid \exists x(t, x_0 \in K, \forall t)\}, \quad (3)$$

其中 $x(t, x_0)$ 是系统从 x_0 开始演化 t 时间的位置.式(3)表示状态集合(约束) K 的生存核是其内部的一个子集,系统(1)从该子集内部任何状态开始运动都能长时间地保持在 K 中,一旦超出这个不可见的子集,

就不可避免地将来脱离 K . 具体到车辆, 车辆状态包括车辆道路横向位置 Y (垂直道路方向位置)、车辆航向角 ψ 、车辆航向角速度 $\dot{\psi}$ 以及车辆侧偏角 α , 由于道路中车辆一直沿道路前进, 计算道路生存核时不考虑车辆的道路纵向位置状态 X (沿道路方向的位置), 仅将其作为随动变量考虑. 车辆道路生存核是车辆状态投影 $[Y \ \psi \ \dot{\psi} \ \alpha]$ 在预定道路约束集合 K 中的一个子集, 由于转弯道路具有非凸特性, 无法直接使用生存核算法计算, 将转弯道路生存核定义为除道路纵向位置状态 X 以外, 车辆状态 $[Y \ \psi \ \dot{\psi} \ \alpha]$ 能够始终保持在道路合法空间 K 集合内的初始状态集合, 这些初始状态连同其对应的状态 X 构成转弯道路生存核, 计算过程将在下文简要表述, 具体可参见文献 [16]. 图 1 为不同车辆速度在相同转弯道路环境下的生存核投影, 车辆速度从左至右分别为 0、2 m/s、5 m/s、10 m/s. 车辆模型参数以及道路约束见表 1.

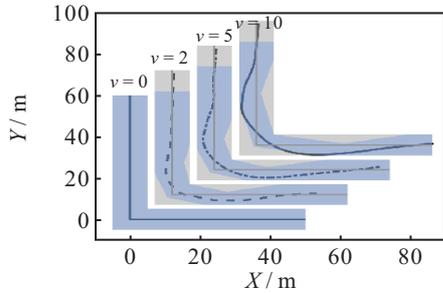


图 1 转弯道路生存核及其中线投影

表 1 车辆模型与约束参数表

	单位	数值	描述
M	kg	1485	质量
I_{zz}	kg · m ²	2570	绕重心的转动惯量
α_1	m	1.09	重心至前轮轴距
α_2	m	1.49	重心至后轮轴距
D_1	N	131,500	前轮侧偏刚度
D_2	N	117,000	后轮侧偏刚度
β	rad	$[-0.2, +0.2]$	转向角范围
X	m	$[0, +60]$	道路长度范围
Y	m	$[-6, +6]$	道路宽度范围
ψ	rad	$[-1, +1]$	航向角约束
$\dot{\psi}$	rad	$[-0.5, +0.5]$	航向角速度约束
α	rad	$[-0.1, +0.1]$	侧偏角范围

图 1 中, 浅灰色区域表示原始道路、深灰色区域为道路生存核投影, 当车辆静止时, 原始道路上的任何一点都是可生存的, 所以道路生存核投影便是原始道路空间本身; 一旦车辆具有速度, 原始道路上的某些转角区域 (图 1 中浅灰色区域) 就不可生存, 即车辆一旦处于这些区域, 一定会与道路边界相撞, 为此必须将车辆状态约束在生存核中.

定义 4^[14] 集合 K 的可生存控制映射为

$$R_{\text{viab}(K)}(x) = \{u \in U | f(x, u) \in T_{\text{viab}(k)}(x)\}. \quad (4)$$

当 $\text{viab}(K) \neq \emptyset$ 且 $R_{\text{viab}(K)}(x) \neq \emptyset$ 时, 动态系统存

在某个控制输入 u , 可使系统速度方向保持在核切锥内, 即 $f(x, u) \in T_{\text{viab}(k)}(x)$. 由上述分析可知: 生存理论构建了动态系统与外界环境的适应性桥梁, 为系统安全优化控制打下了坚实基础. 下面简要介绍车辆动力学模型, 在此基础上进行生存理论迭代计算.

1.2 车辆模型建立

车辆动力学分析采用如图 2 所示的双轮模型. 其中: ψ 为车辆航向角, α 为质心侧偏角 (速度方向与自身纵向轴线间的夹角), α_1 、 α_2 分别为车辆重心与前后轮轴的距离, β 为前轮转向角. 假设车辆正常行驶的侧偏角很小, 车辆动力学模型可由状态空间模型 $\dot{x} = Ax + Bu$ ^[16] 表示为

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{\psi} \\ \dot{\psi} \\ \dot{\alpha} \end{bmatrix}^T = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & V & 0 & V \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -\frac{D_1\alpha_1^2 + D_2\alpha_2^2}{VI_{zz}} & -\frac{D_1\alpha_1 + D_2\alpha_2}{I_{zz}} \\ 0 & 0 & 0 & -\left(1 + \frac{D_1\alpha_1 - D_2\alpha_2}{MV^2}\right) & -\frac{D_1 + D_2}{MV} \end{bmatrix} \begin{bmatrix} X \\ Y \\ \psi \\ \dot{\psi} \\ \alpha \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & -\frac{D_1\alpha_1}{I_{zz}} \\ 0 & -\frac{D_1}{MV} \end{bmatrix} \begin{bmatrix} V \\ \beta \end{bmatrix}. \quad (5)$$

其中: A 为状态转移矩阵, B 为输入矩阵, Y 为车辆道路横向位置 (垂直道路方向的位置), X 为车辆的道路纵向位置, M 为车辆的质量, V 为车辆速度, D_1 、 D_2 分别为前后轮侧偏刚度, I_{zz} 为车辆绕重心的横摆转动惯量, δ_1 、 δ_2 分别为前后轮的侧偏角. 由模型 (5) 可知, 车辆的速度变化会导致系统时变, 如果保持车辆速度恒定 (V 为常数), 则该模型可简化为线性时不变系统, 有利于计算道路生存核; 另一方面, 若限制速度恒定, 车辆安全则将全部依赖转向控制, 文献 [22] 表明, 车辆避障采用转向策略相比制动策略更加有效.

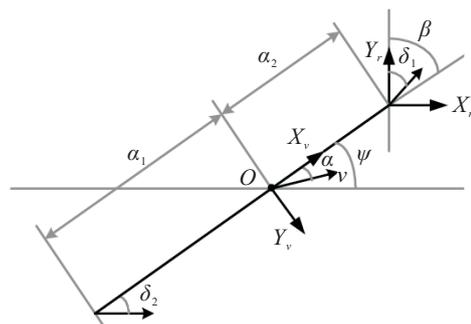


图 2 车辆双轮模型及其坐标系

系统的状态与控制约束体现了车辆的安全区域、机械限制、电气特性以及舒适区间,以表1为例,列出一组车辆模型的计算参数以及相关约束。

1.3 转弯道路生存核计算

由于十字路口处的直角转弯较为普遍,本文以直角弯道自动驾驶作为典型应用场景,使用生存理论分析道路安全和局部路径规划。直角弯道可看作两个相互垂直的直线道路连接,如图3所示,出口与入口直线道路构成转弯道路,生存分析转弯道路之前需要先计算直线道路生存核。

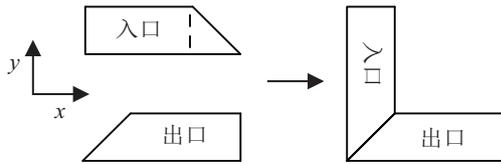


图3 转弯道路组成

直线道路生存核计算是基于凸紧集生存核计算,为表明生存核计算过程,下面简要介绍车辆模型相对一般凸紧集的生存核计算方法。

生存核迭代算法建立在离散时间系统之上,所以将车辆模型(5)离散化为

$$x_{k+1} = Gx_k + Hu_k. \quad (6)$$

其中: $x_k = [X_k \ Y_k \ \psi_k \ \dot{\psi}_k \ \alpha_k]^T, u_k = [V \ \beta]^T \in U$. 离散的状态矩阵与输入矩阵分别为

$$G \approx I + AT_S, \quad H \approx BT_S. \quad (7)$$

其中: I 为单位矩阵, T_S 为采样时间。由于速度 V 恒定,模型控制 u_k 仅为转向 β 。将离散系统(6)写成微分包含的形式,有

$$x_{k+1} \in F(x_k), \quad k > 0, \quad (8)$$

其中 $F: R^5 \mapsto R^5$ 为上半连续集值映射。离散系统(8)相对凸紧集 K 的生存核可通过圣皮埃尔(Saint Pierre)生存核迭代算法计算得到^[23],具体迭代过程如下:

$$\begin{cases} K_{n+1} = x \in K_n | K_n \cap F(x) \neq \emptyset, \\ K_0 = K. \end{cases} \quad (9)$$

其中: \emptyset 为空集; $K_0, K_1, \dots, K_n, \dots$ 是一组子集,随着迭代的深入,这些子集会逐渐收缩,如果 K_{n+1} 收敛到与 K_n 相同,则 K_n 为原空间 K 的生存核。该迭代算法也可使用向后一步可达计算得到^[16],即

$$\begin{cases} K_{n+1} = K_0 \cap \text{reach}_1(K_n), \\ K_0 = K. \end{cases} \quad (10)$$

其中 $\text{reach}_1(\cdot)$ 为集合向后一步可达运算。对于系统(6)相对集合 T 的向后一步可达运算可定义为

$$\text{reach}_1(T) = x \in R^5 | Gx + Hu \in T, u \in U. \quad (11)$$

可见,生存核迭代算法(10)需要计算集合求交与集合向后一步可达,为方便这两项计算,紧凸集合 K 可用多面体 P 表示。多面体表示方法有两种:

1) 多面体的 H 表示法由半平面空间包围而成,表示如下:

$$P = \bigcap_{i=1}^f \{x \in R^5 | h_i \cdot x \leq b_i, i = 1, 2, \dots, 5\} = \{x \in R^5 | Hx \leq b\}. \quad (12)$$

2) 设 v_1, v_2, \dots, v_k 为一组多面体顶点,则多面体的 V 表示法为

$$P = \text{conv}(v_1, v_2, \dots, v_k), \quad (13)$$

其中 $\text{conv}(\cdot)$ 为凸包运算。

多面体的 H 表示方法与 V 表示方法可以相互转换,具体参见文献[24]。例如,将表1状态约束转化为多面体 P 的 H 表示法,式(12)的具体参数为

$$\begin{aligned} h_1 &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 60 \\ 0 \end{bmatrix}; \\ h_2 &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \end{bmatrix}, \quad b_2 = \begin{bmatrix} 6 \\ 6 \end{bmatrix}; \\ h_3 &= \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix}, \quad b_3 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}; \\ h_4 &= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 \end{bmatrix}, \quad b_4 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}; \\ h_5 &= \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix}, \quad b_5 = \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix}. \end{aligned} \quad (14)$$

约束多面体 P 相对车辆系统(6)的向后一步可达运算为

$$\text{reach}_1(P) = G^{-1}(P \oplus (-H)U). \quad (15)$$

其中: G^{-1} 为矩阵 G 的逆, U 为控制约束多面体。当速度恒定 $v = 8 \text{ m/s}$, 转向控制 β 约束参见表1,则控制约束 U 采用多面体 H 表示法(12)为

$$U = \bigcap_{i=1}^f \{u \in R^2 | h_i u \leq b_i, i = 1, 2\}. \quad (16)$$

其中

$$\begin{aligned} h_1 &= \begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 8 \\ -8 \end{bmatrix}, \\ h_2 &= \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix}, \quad b_2 = \begin{bmatrix} 0.2 \\ 0.2 \end{bmatrix}. \end{aligned}$$

操作符 \oplus 表示集合闵可夫斯基求和操作,考虑两个多面体集合 P 与 W , 有

$$P \oplus W = \{x + w \in R^5 | x \in P, w \in W\}. \quad (17)$$

可见,多面体生存核计算可进一步分解为线性变换、集合求交以及闵可夫斯基求和运算.考虑 V 表示法多面体 P, M 为线性矩阵,线性变换可由下式计算:

$$MP = \{x \in \text{conv}(Mv_1, Mv_2, \dots, Mv_k)\}. \quad (18)$$

考虑两个采用 H 表示法的多面体 $P_1 = \{x \in R^5 | H_1x \leq b_1\}, P_2 = \{x \in R^5 | H_2x \leq b_2\}$,则 P_1, P_2 的交集计算方法为

$$P_1 \cap P_2 = \{x \in R^5 | H_1x \leq b_1 \text{ and } H_2x \leq b_2\} = \left\{x \in R^5 : \begin{bmatrix} H_1 \\ H_2 \end{bmatrix} x \leq \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}\right\}. \quad (19)$$

考虑两个采用 V 表示法的多面体 P_1, P_2 ,闵可夫斯基求和计算如下:

$$P_1 \oplus P_2 = \text{conv}\{v_i + u_j | i = 1, 2, \dots, k, j = 1, 2, \dots, l\}, \quad (20)$$

其中 v_i, u_j 分别为 P_1, P_2 的顶点.

多面体表示车辆状态约束不仅包括低维的道路边界限制($[X \ Y]^T$),也同时包括朝向角(角速度)和侧偏角 $[\psi \ \dot{\psi} \ \alpha]^T$ 约束用以保证驾驶舒适.

直线道路由于允许车辆的道路纵向位置状态 X 自由移动,在计算直线道路生存核时需要特殊处理,转弯道路非凸,其生存核计算需要先计算出口道路的生存核,然后结合出口入口交界面的集合关系确定入口道路的生存核,具体计算原理和处理过程见文献[16].

1.4 基于机器学习的生存路径生成方法

以文献[16]给出的转弯道路生存核为例,车辆速度恒定为8 m/s,根据图4设定状态约束多面体和模型参数,转弯道路生存核投影在 $[X \ Y \ \psi]^T$ 空间如图4所示.可见转弯道路生存核描述了所有控制行为下所有高维状态空间内车辆的安全信息,为智能车辆全面、客观地理解道路环境动态适应性提供了无可比拟的理论工具.

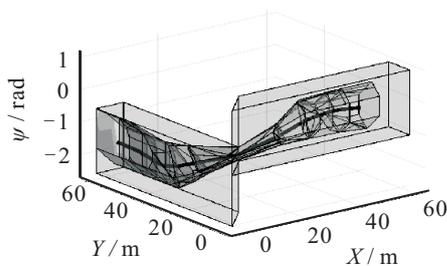


图4 生存核在 $[X \ Y \ \psi]$ 空间投影

直接将道路生存核信息用于智能车辆系统存在两个问题:一是计算复杂度过高,迭代算法(10)每一次计算都涉及集合的求交与反向可达计算,而每次求

交后集合的顶点会增多,这将增加下次迭代的计算负荷.如果模型本身维度过高,同时模型收敛次数过大,也会造成计算时间呈指数级增长,在模型(6)离散时间 T_s 为0.2 s的情况下,图4算例的计算时间为2.2 s,即车辆至少行驶16 m,才能给出2 s前的道路生存核,减少 T_s 会获得更精细的生存核输出,但是计算时间将呈指数增长.另外,图4展示的弯道生存核顶点复杂,这是迭代过程中集合相交的结果,难以通过函数直接输出,由于核边界是高维空间曲面,直接放入车辆智能系统中使用,仅能对已发生的危险状态进行报警,不像传统模型预测方法能规划出局部道路轨迹,从而为智能驾驶系统提供最优参考.既然生存核边界标记了车辆最极端的高维状态,这些状态或者属于道路位置极限,或者属于控制极限,为此尝试使用高维生存核中线来生成局部道路轨迹.图4折线是生存核中线,由于远离极端状态,可以作为车速与道路环境(高维)匹配的最优路径.

生存核中线是高维状态中心,需向道路空间 $[X \ Y]^T$ 投影才能被智能驾驶系统使用,为此需要计算不同车速相同道路环境下的生存核.如图1所示,当车辆静止时,生存核投影即是道路空间本身,生存核中线投影便是道路的几何中线,提升速度后,生存核开始收缩,某些转弯区域变得不可生存,生存核中线的投影曲率也随着车速的提升逐渐变小.根据前述分析,当车辆在转弯过程中变速时,理论算法计算时间无法保证实时输出局部最优路径.机器学习模型可以快速输出训练模式,由于生存核高维中心线具有高度的驾驶安全性,拟采用机器学习模型输出该中心线在道路 $[X \ Y]^T$ 平面的投影,以实现智能驾驶目的.

常见的机器学习方法有回归分析、SVM(支持向量机)、贝叶斯预估、决策树、神经网络、深度学习等.其中回归分析需要变量之间存在较强的相关关系,否则回归结果会有较大误差;SVM需要通过核函数向高维空间映射来解释局部道路的非线性,但其多用于解决分类问题,较少用于拟合;由于生存核相关的参数较多,训练贝叶斯模型比较复杂;决策树中的CART算法既能做分类也能做回归分析,不过完整的决策树模型包含大量过于复杂的分支,属于非必须的结构,算法容易过拟合,导致泛化能力不强;深度学习可以应对大量特征输入,本身需要多层神经元来参加计算,对车载硬件性能要求较高,适用于道路场景图像理解、语音、文本等非结构化数据学习;对于规划路径的拟合反而是浅层神经网络具有较好的实用效

果,尤其是针对局部多变量非线性拟合问题,径向基神经网络(RBF)具有较好的表现力,RBF神经网络属于局部信息拟合网络,相对于全局信息作用的BP神经网络,其学习速度更快,并可避免局部极小问题,而且RBF网络可以根据具体问题来确定相应的网络拓扑,具有自学习、自组织、自适应的能力,对非线性连续函数具有逼近一致性,学习速度快,可适应大范围的数据融合,又能高速输出结果,适合实时控制需要。

1.5 机器学习模型训练方法

由于首次将生存理论引入机器学习训练,以图1所示的典型直角转弯道路作为训练目标,将生存核中心路径作为神经网络输出,网络模型输入则为车辆的内部状态与外部情况,包括车辆速度、道路宽度、车辆位置等信息,其他参数如车身质量,轮胎刚度等相对具体车型而言可看作常量,其中车辆速度、道路宽度较易作为变量输入网络,但是车辆相对道路位置需要特殊量化方法才能将其转化成输入变量,同理,模型输出的规划路径也需要适当形式来表征。为此,以道路几何中线为基准,将转弯中心点设为核心点,从核心点出发沿道路几何中心向出口与入口两侧拓展,共选取11个特征中心点,其中出口几何中心线与入口几何中心线各5点,核心点1点,如图5所示,这样道路位置信息便可以包含在这些特征点中。另一方面,生存核中心投影输出可以表征为一组相对特征中心点的道路径向偏置,如图5实线段所示。如果车辆静止,道路最优路径便是几何中心线,当车辆运动,道路最优路径会随车辆模型(5)变化发生偏移,考虑到转弯处路径具有非线性,均匀选取特征中心点相对在弯道核心点附近密集取点会获得不同的训练效果。

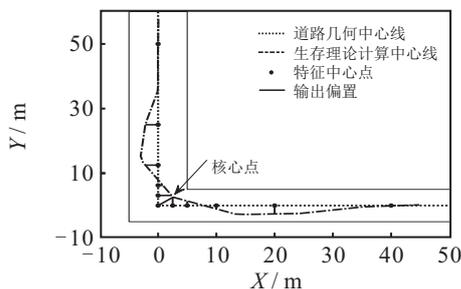


图5 RBF神经网络特征数据选取

特征数据选取如图5所示,其中点划线为理论计算的生存核中心投影,其特征点处的径向偏置可用作模型训练的目标数据。模型输入为车辆速度、道路宽度以及11个特征中心点在道路坐标系下的 X 坐标,模型输出与训练目标一致,即相对特征中心点的径向偏置。特征变量获取完毕后,需要对理论特征数据进行预处理,包括数据清洗(离群点检测、异常值处理)

和数据变换。

理论计算的生存核中线是由局部生存核的高维中心点连接而形成的折线,由于模型简化、线性化、离散化以及集合运算的误差,随着迭代过程的加深,某些位置会产生较大偏离。为保证训练精度,先将偏离几何中心超过3m的离群点剔除,然后对剩余的折线点使用插值方法,从而将理论计算的折线中心折线转为较光滑的曲线。采用曲线光滑性和计算稳定性均较好的三次样条插值法替代原有理论计算的生存核中心投影折线。

网络的输入向量为车辆的速度、道路宽度以及11个特征中心点 X 坐标,输入层神经元数目为13个;输出层神经元数目为11个径向偏置数据;输出向量即为生存核中心线投影相对几何中心线的偏移数据。考虑到模型泛化能力测试,等间距地在4~11m/s之间选取15种车辆速度,在4~6m之间选取5种道路宽度,共计75种组合的理论计算生存核中线作为训练数据。RBF神经网络权重参数的训练使用梯度下降法。径向基函数中心、宽度和调节权重参数均通过学习自适应调节到最佳值,使用高斯函数作为径向基函数,均方误差设为0.01,径向基函数的扩散速度设为0.9,通过Matlab的newrb函数创建一个径向基函数网络,该网络可以自动增加网络隐含层的神经元个数,直到均方误差满足精度或神经元个数达到最大值,最终的隐含层节点数为74。

2 机器学习性能结果分析

图6对比了理论计算轨迹与神经网络输出,三角标记虚线表示的网络输出轨迹与实线的理论计算轨迹基本重合,由于训练前使用了数据清洗与插值方法,神经网络输出更加光滑,有利于高速智能车辆转弯控制,同时其安全性具有理论保障。

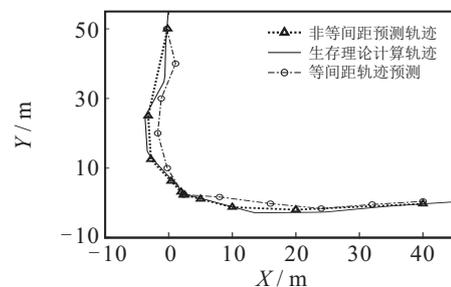


图6 均匀/非均匀特征中心点法训练效果对比

2.1 均匀/非均匀特征中心点法对模型精度的影响

特征中心点位置的选取关系到模型的输入与输出,考虑到弯道具有非线性,图4转弯位置的生存核变动明显,所以采用对弯道位置密集取点、直线道路

稀疏取点的非均匀特征中心点输入方法,该方法比采用均匀分布的特征中心点训练效果更好.图6显示了7.5 m/s,路宽5 m情况下,两种训练方法的效果对比,可见,在相同网络规模下,采用均匀特征中心点法建立的模型在转弯处的输出明显与理论偏差较大.

由于转弯道路处于事故高发区域,图4表明安全区域在转弯处显著缩小,非线性也显著增强,一方面在转弯处密集选取特征点有利于提高输出精度,另一方面,RBF的自组织算法也会对密集的特征中心点位置进行局部优化,使得输出结果偏差小,稳定性高,有利于车辆的智能驾驶控制.

2.2 非均匀特征中心点法的参数敏感性分析

选择道路宽度分别为4 m、4.5 m、5 m、5.5 m、6 m,不同速度状态下机器学习和理论计算的误差如图7所示.车辆速度分别为4.5 m/s、5.5 m/s、6.5 m/s、8.5 m/s、9.5 m/s时,不同道路宽度情况下的误差如图8所示.误差计算公式为

$$e = \frac{1}{k} \sum_{i=1}^k |y_i - r_i|. \tag{21}$$

其中: k 为特征中心点的个数, y_i 和 r_i 分别为理论计算与模型预测在特征点处的径向偏差.

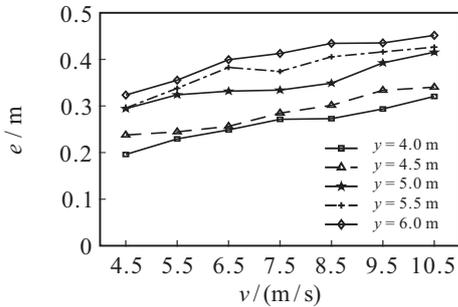


图 7 不同速度下的误差

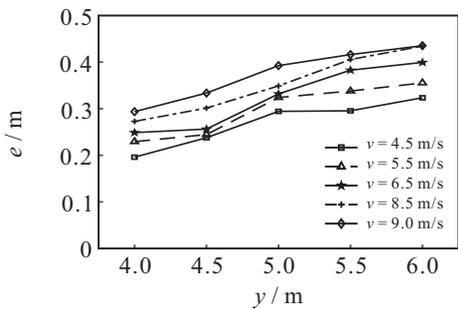


图 8 不同道路宽度下的误差

图7表明,随着速度增大,模型输出误差会随之增大,但是随着道路宽度增加,误差扩大趋势逐渐放缓;图8表明,当道路变宽时,模型输出误差会增大,但总体误差可控,由于生存理论中线能保证最大的动力学控制裕度,输出的模型精度能够保证直角弯道智能驾驶安全.

2.3 模型计算时间与理论计算时间对比

由于生存理论停止迭代要求集合收敛,当迭代次数较多时,迭代时间将呈指数增长.表2对比了模型离散化时间为1.5 s,不同速度和宽度情况下理论计算与网络输出时间,测试软件为Matlab 2017bata,测试环境为CORE i7处理器、8 GB内存、Windows10操作系统.结果表明,神经网络模型能大幅提升生存路径输出速度.

表 2 计算速度对比

	数值				
速度/(m/s)	4.5	6.5	8.5	9.5	10.5
宽度/m	4.5	4.5	5.5	5.5	5.5
生存理论/s	2.303 0	2.167 4	2.195 4	2.008 2	2.053 7
网络输出/s	0.006 9	0.006 4	0.006 6	0.006 2	0.667 1

2.4 数据驱动模型的泛化能力分析

泛化能力是衡量神经网络模型性能的重要指标之一,对上述RBF模型进行泛化能力分析,用于模型训练的数据来自于速度范围为4~11 m/s,道路宽度范围为4~6 m的生存理论计算.

在训练数据范围外选取更高车辆速度观察模型的泛化能力,假设车辆速度12 m/s,道路宽度8 m,结果如图9所示.虽然网络输出结果跟踪上了理论计算,但在弯道处偏差较大,主要原因在于非线性特征导致模型过拟合,使得模型在输入数据超出训练集范围的情况下泛化能力下降.但是,模型训练的数据来自理论计算而非驾驶员实地驾驶记录,所以可以轻松获取更大范围的理论数据应用于模型训练,从而保证模型在极端状态下(如高速度、窄路面)输出合理精度.

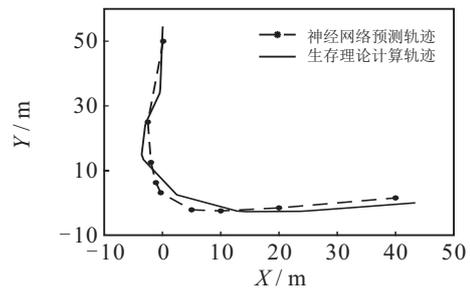


图 9 车速12 m/s,路宽8 m的神经网络与理论计算输出

3 高速自动驾驶转弯控制实验

图10为基于simulink的自动驾驶转弯控制仿真程序, x 为车辆模型(5)的5个状态.为展示神经网络生成的生存路径信息具有驾驶控制的简易性,采用基于单纯跟踪(pure pursuit)的几何导航控制器(geometric controller)进行仿真实验^[25].其中: K 为比例系数, θ 为当前位置前方1 s行进距离的参考轨迹中心点的期望航向角.为使车辆沿神经网络输出的路

径行驶,控制函数模块中有神经网络输出的路径数据,将其转换为汽车行驶期望航向角 θ ,然后通过控制器作用,使车辆航向角 ψ 跟踪期望航向角 θ .

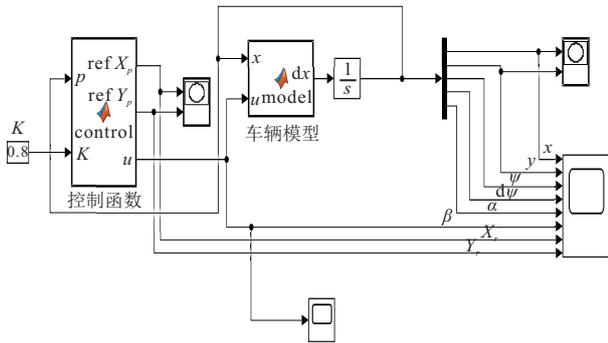


图10 simulink 仿真

车辆模型模块中包含车辆的原始非线性动力学模型^[16]

$$\begin{bmatrix} \ddot{\psi} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} -\frac{D_1\alpha_1^2 + D_2\alpha_2^2}{VI_{zz}} & -\frac{D_1\alpha_1 + D_2\alpha_2}{I_{zz}} \\ -\left(1 + \frac{D_1\alpha_1 - D_2\alpha_2}{MV^2}\right) & -\frac{D_1 + D_2}{MV} \end{bmatrix} \begin{bmatrix} \dot{\psi} \\ \alpha \end{bmatrix} + \begin{bmatrix} \frac{D_1\alpha_1}{I_{zz}} \\ -\frac{D_1}{MV} \end{bmatrix} \beta, \quad (21)$$

$$\dot{V} = \gamma,$$

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} \cos(\psi + \alpha) \\ \sin(\psi + \alpha) \end{bmatrix} V. \quad (22)$$

将式(22)线性化即为生存核理论计算所使用的系统模型(5),模型参数见表1.为增加控制难度,将模型速度设为恒定9.2 m/s,道路宽度设定为4.5 m,90°弯道,则该模块的控制输入 u 仅限于使用车辆的转向控制 β , $\beta = K(\theta - \psi)$.仿真结果如图11所示.结果表明,通过简单控制器得出的转向操作即可保证车辆在恒定高速状态下安全大曲率转弯,并基本跟踪上了规划路径.表明所生成的路径控制裕度较大,这得益于理论计算中心轨迹远离高维极端状态和极限控制,所以容易实现大曲率的高速转向控制.由于控制变量受表1的约束限制($-0.2 \leq \beta \leq 0.2$),在比例控制器后端配有 β 的饱和输出限制,有

$$\beta = \begin{cases} 0.2, & u > 0.2; \\ u, & -0.2 \leq u \leq 0.2; \\ -0.2, & u < -0.2. \end{cases} \quad (23)$$

控制输入 β 的变化曲线如图12所示,其中 β 在5.5 s和7.5 s左右发生控制饱和,此时的转弯曲率达到最大.上述仿真模型的建立使用了真实非线性车辆模型(22),而生存理论计算采用线性化模型,并以此

训练了神经网络模型.图11结果表明,神经网络模型给出的规划路径符合原始车辆模型的高速转弯,这是由于线性化后的模型(5)相比原始模型(22),在相同控制范围内($-0.2 \leq \beta \leq 0.2$)转弯的幅度有所降低,因此计算出的生存核空间相比真实车辆的性能保守,安全性更高.

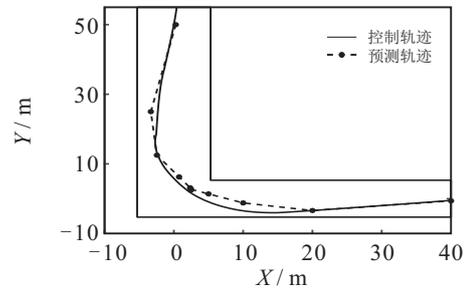


图11 轨迹控制曲线

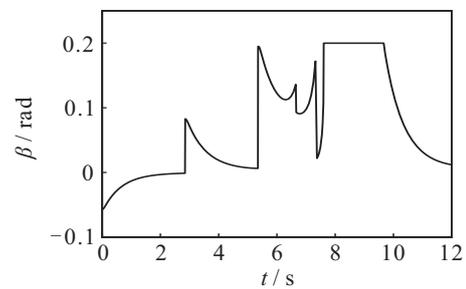


图12 控制输入 β 变化曲线

4 结论

得益于生存理论提供的训练数据,基于RBF神经网络可以较高精度地输出规划路径,解决生存理论计算速度过慢的问题,控制效果快速、精确、稳定.通过两种数据训练方法(均匀与非均匀)的对比分析表明,采用非均匀数据训练方法具有更高的模型输出精度,同时也具有一定的模型泛化能力.综上,基于生存理论与机器学习技术提出的智能车辆高速控制方法,可以提高车辆控制的实时性,并有效提升通行效率,减少交通事故,在智能交通领域具有广阔的应用前景.作为生存理论的机器学习初探,目前仅以直角弯道、车辆质点模型作为典型应用,下阶段将继续开展车辆刚体在多弯道复杂道路环境下的机器学习生存路径研究,使自动驾驶车辆的生存控制更加贴合实际.

参考文献(References)

- [1] Krzysztof K. Anti-lock braking system modelling and parameters identification[C]. International Conference on Methods and Models in Automation and Robotics(MMAR). Brazilian: IEEE Press, 2014: 342-346.
- [2] Lie A, Tingvall C, Krafft M. The effectiveness of ESP(electronic stability program) in reducing real life

- accidents[J]. *Journal of Crash Prevention & Injury Control*, 2004, 5(1): 37-41.
- [3] Abid O, Cabannes Q, Senouci B. Supervisor and control investigation in smart/autonomous vehicles: Environment recognition and objects detection ADAS application case study[C]. *International Symposium on Mechatronics and its Application (ISMA)*. Brazilian: IEEE Press, 2018: 1-7.
- [4] Gordon T J, Lidberg M. Automated driving and autonomous functions on road vehicles[J]. *Vehicle System Dynamics*, 2015, 53(5): 958-994.
- [5] Katrakazas C, Quddus M, Chen W H, et al. Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions[J]. *Transportation Research, Part C: Emerging Technologies*, 2015, 60(11): 416-442.
- [6] 张照生. 交通约束下的行车最优路径规划[D]. 北京: 清华大学汽车工程系, 2013.
(Zhang Z S. Optimum vehicular path planning under traffic restriction[D]. Beijing: Department of Automotive Engineering, Tsinghua University, 2013.)
- [7] Jiang J, Zhang H C. Research on robot path planning based on improved potential field ant colony algorithm[J]. *Computer Simulation*, 2016, 33(9): 329-334.
- [8] Gao Y. Determining the viability for a class of nonlinear control systems on a region with nonsmooth boundary[J]. *Control and Decision*, 2006, 21(8): 923-926.
- [9] Xu X, Lu S F, Chen L. Trajectory tracking of distributed-drive self-driving vehicle based on coordination between autonomous steering and differential steering[J]. *Automotive Engineer*, 2018, 40(4): 475-481.
- [10] Xiu C J, Chen H. A research on local path planning for autonomous vehicles based on improved APF method[J]. *Automotive Engineer*, 2013, 35(9): 808-811.
- [11] Ma L, Xue J, Kawabata K, et al. A fast RRT algorithm for motion planning of autonomous road vehicles[C]. *The 17th International IEEE Conference on Intelligent Transportation Systems*. Qingdao: IEEE, 2014: 1033-1038.
- [12] Gray A, Gao Y, Lin T, et al. Predictive control for agile semi-autonomous ground vehicles using motion primitives[C]. *2012 American Control Conference (ACC)*. Montreal: IEEE, 2012: 27-29.
- [13] Ji J, Tang Z R, Wu M Y, et al. Path planning and tracking for lane changing based on model predictive control[J]. *China Journal of Highway and Transport*, 2018, 31(4): 172-179.
- [14] Vandanjon P O, Coiret A, Lorino T. Viability theory and road safety[J]. *Vehicle System Dynamics*, 2013, 51(4): 602-623.
- [15] Aubin J P. *Viability theory*[M]. Boston: Birkhauser, 1991: 1-42.
- [16] Liu L, Gao Y, Wang C. Road safety analysis for high-speed vehicle in complex environments based on the viability kernel[J]. *IET Intelligent Transport Systems*, 2018, 12(6): 495-503.
- [17] Liniger A, Lygeros J. Real-time control for autonomous racing based on viability theory[J]. *IEEE Transactions on Control Systems Technology*, 2019, 27(2): 464-478.
- [18] Tom Mitchell. *Machine learning*[M]. 北京: 机械工业出版社, 2013: 81-126.
- [19] Dimitra P, Kyriakopoulos K J. Viability control for a class of under actuated systems[J]. *Automatica*, 2013, 49(1): 17-29.
- [20] Liu L, Gao Y, Wu Y P. High speed obstacle avoidance control of wheeled mobile robots with non-homonymic constraint based on viability theory[J]. *Control and Decision*, 2014, 29(9): 1623-1627.
- [21] Gao Y. Viability criteria for differential inclusion[J]. *Journal of Systems Science and Complexity*, 2011, 24(5): 825-834.
- [22] Soudbakhsh D, Eskandarian A. Steering control collision avoidance system and verification through subject study[J]. *IET Intelligent Transportation System*, 2015, 10(9): 907-915.
- [23] Saint-Pierre P. Approximation of the viability kernel[J]. *Applied Mathematics and Optimization*, 1994, 29(2): 187-209.
- [24] Guernic C. Le. 'Reachability analysis of hybrid systems with linear continuous dynamics'[D]. Grenoble: Université Joseph-Fourier-Grenoble I, 2009.
- [25] Amer N H, Zamzuri H, Hudha K, et al. Modelling and control strategies in path tracking control for autonomous ground vehicles: A review of state of the art and challenges[J]. *Journal of Intelligent & Robotic Systems*, 2017, 86(2): 225-254.

作者简介

刘磊(1982—), 男, 讲师, 博士, 从事机器人自动控制、智能交通系统等研究, E-mail: liulei@usst.edu.cn;

杨晔(1995—), 男, 硕士生, 从事智能交通、数据挖掘的研究, E-mail: 739470357@qq.com;

刘赛(1994—), 男, 硕士生, 从事机器人控制的研究, E-mail: 2201635701@qq.com;

高岩(1962—), 男, 教授, 博士生导师, 从事非光滑优化以及生存理论等研究, E-mail: gaoyan@usst.edu.cn;

王富正(1968—), 男, 教授, 博士, 从事新能源汽车等研究, E-mail: ffw@ntu.edu.tw;

王亚刚(1967—), 男, 教授, 博士, 从事自动控制系统等研究, E-mail: ygwang@usst.edu.cn.

(责任编辑: 郑晓蕾)