



(12) 发明专利申请

(10) 申请公布号 CN 112109727 A

(43) 申请公布日 2020. 12. 22

(21) 申请号 202010943841.1

B60T 17/22 (2006.01)

(22) 申请日 2020.09.08

G06F 17/11 (2006.01)

(71) 申请人 北京踏歌智行科技有限公司

G06K 9/62 (2006.01)

地址 100176 北京市北京经济技术开发区
科谷一街10号院2号楼9层901

G06N 3/04 (2006.01)

申请人 内蒙古霍林河露天煤业股份有限公司

G06N 3/08 (2006.01)

(72) 发明人 刘丽 张巍 袁广忠 赵明磊
孟祥春 张海福

(74) 专利代理机构 北京航智知识产权代理事务
所(普通合伙) 11668

代理人 黄川 史继颖

(51) Int.Cl.

B60W 50/02 (2012.01)

B60W 40/105 (2012.01)

权利要求书3页 说明书8页 附图7页

(54) 发明名称

一种面向露天矿区无人驾驶车辆的制动力
标定方法

(57) 摘要

本发明公开了一种面向露天矿区无人驾驶车辆的制动力标定方法,通过获取不同制动开度、车速以及制动产生的制动力的数据集,采用神经网络离线训练数据集的方式来获得三者之间非线性的关系,标定出不同制动开度和不同速度下的理想制动力,用于进行无人驾驶车辆的制动失效故障诊断。本发明的方法无需采购其他装置或仪器进行制动力的标定,方便快捷,适用性较强,且获得制动力与制动开度、速度三者的非线性关系,更易于针对无人驾驶车辆进行制动失效故障的诊断。



1. 一种露天矿区无人驾驶车辆的制动力标定方法,其特征在于,包括以下步骤:

S1:在车辆进行无人驾驶前,获取一系列制动开度及其对应的车速,根据车辆动力学公式计算出相应的制动力;

S2:对步骤S1获取制动开度、车速和制动力数据进行预处理,作为神经网络的训练样本数据集;

S3:根据步骤S2预处理后得到的训练样本数据集,将制动开度和车速作为BP神经网络的输入,制动力作为BP神经网络的输出,离线训练,得到三者的非线性关系;

S4:在车辆进行无人驾驶时,获取车辆当前的制动开度和车速信息;

S5:将步骤S4获取的信息输入步骤S3得到的非线性关系,经运算得到理想制动力,作为车辆产生实际制动力的对比标准,进行制动失效故障诊断:

实际制动力与理想制动力的差距大于设定的阈值,说明车辆发生制动失效故障;

实际制动力与理想制动力的差距小于设定的阈值,则进行下一时刻的理想制动力计算和制动失效故障的诊断。

2. 根据权利要求1所述的一种露天矿区无人驾驶车辆的制动力标定方法,其特征在于,所述步骤S1中,通过车辆的线控反馈获取制动开度数据,通过轮速计或GPS获取车速数据;制动力F的计算公式为: $F=G*h*\cos\alpha+G*\sin\alpha+m*a$,其中,G为作用于车上的重力, $G=m*g$,m为汽车质量,单位为kg;g为重力加速度, $g=9.8m/s^2$;h为滚动阻力系数; α 为坡度;a为车辆加速度。

3. 根据权利要求1所述的一种露天矿区无人驾驶车辆的制动力标定方法,其特征在于,所述步骤S2中,采用欠采样或过抽样进行数据处理。

4. 根据权利要求1所述的一种露天矿区无人驾驶车辆的制动力标定方法,其特征在于,所述步骤S3中离线训练BP神经网络的具体步骤为:

S3-1:网络初始化;

将制动开度和车速作为输入X,制动力作为输出U,根据输入输出,确定网络输入层神经元数、隐含层神经元数、输出层神经元数;

初始化输入层、隐含层和输出层神经元之间的连接权,具体为,隐含层为一层,则输入层第h个神经元与隐含层第i个神经元之间的连接权为 W_{hi} ,其中, $h=1,2, \dots, m$;隐含层为至少两层,则第一个隐含层第i个神经元到第二个隐含层第j个神经元之间的连接权为 G_{ij} ,以此类推, $j=1,2, \dots, n$;隐含层为一层,则隐含层第i个神经元到输出层第k个神经元之间的连接权为 F_{ik} ,其中, $k=1$;隐含层为至少两层,则第二个隐含层的第j个神经元到输出层的第k个神经元之间的连接权为 F_{jk} ,其中, $k=1$;

S3-2:计算隐含层输出;

将步骤S2预处理后的样本数据集中制动开度和车速数据输入,隐含层为一层,则第一个隐含层第i个神经元接收到的输入 α_i 为:

$$\alpha_i = f\left(\sum_{h=1}^m W_{hi} X_h\right) \quad (1)$$

隐含层为至少两层,则第二个隐含层第j个神经元接收到的输入 β_j 为:

$$\beta_j = f\left(\sum_{i=1}^m G_{ij} \alpha_i\right) \quad (2)$$

其中, X_h 为输入层的第h个神经元, G_{ij} 为第一个隐含层的第i个神经元到第二个隐含层

的第j个神经元之间的权值；神经网络结构中神经元的传递函数采用双曲正切函数tansig，tansig传递函数f(x)为：

$$f(x) = \frac{2}{1+e^{-2x}} - 1 \quad (3)$$

其中，e为自然对数函数的底数，x为自变量；

S3-3:计算输出层输出；

隐含层为一层，隐含层的最终输出作为输出层的输入，则输出层第k个神经元接收到的输入 U_k 为：

$$U_k = f(\sum_{j=1}^n F_{jk} \alpha_j) \quad (4)$$

隐含层为至少两层，则输出层第k个神经元接收到的输入 U_k 为：

$$U_k = f(\sum_{j=1}^n F_{jk} \beta_j) \quad (5)$$

其中， F_{jk} 为第二个隐含层的第j个神经元到输出层的第k个神经元之间的权值；

S3-4:均方误差函数；

对训练例 $(X^p \ Y^p)$ ，p为训练样本数，神经网络的输出为 U^p ，则经过神经网络的输出与实际输出的均方误差 E_p 为：

$$E_p = \frac{1}{2} (\hat{U}_k^p - U_k^p)^2 \quad (6)$$

其中， U_k^p 为第p个样本输入值输入到神经网络，经过神经网络计算后，第k个输出神经元输出的值， $\hat{U}_k^p$ 为第p个训练样本输出值；

S3-5:判断是否达到目标；

$E_p > \varepsilon$ ，则进行步骤S3-7的权值更新，否则训练结束， ε 为训练要求精度；

S3-6:判断是否达到最大训练次数，达到最大训练次数，训练结束；

S3-7:权值更新；

根据梯度下降法，隐含层为一层，由均方误差函数 E_p 调整神经元的权值如下：

$$\begin{cases} F_{ik}(z) = F_{ik}(z-1) - \eta \frac{\partial E_k}{\partial F_{ik}} \\ W_{hi}(z) = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial W_{hi}} = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial F_{ik}} \frac{\partial F_{ik}}{\partial W_{hi}} \end{cases} \quad (7)$$

隐含层为至少两层，调整神经元的权值如下：

$$\begin{cases} F_{jk}(z) = F_{jk}(z-1) - \eta \frac{\partial E_k}{\partial F_{jk}} \\ G_{ij}(z) = G_{ij}(z-1) - \eta \frac{\partial E_k}{\partial G_{ij}} = G_{ij}(z-1) - \eta \frac{\partial E_k}{\partial F_{jk}} \frac{\partial F_{jk}}{\partial G_{ij}} \\ W_{hi}(z) = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial W_{hi}} = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial F_{jk}} \frac{\partial F_{jk}}{\partial G_{ij}} \frac{\partial G_{ij}}{\partial W_{hi}} \end{cases} \quad (8)$$

其中， $F_{ik}(z)$ 为本次第一个隐含层的第i个神经元到输出层的第k个神经元之间的权值， $F_{ik}(z-1)$ 为上一次第一个隐含层的第i个神经元到输出层的第k个神经元之间的权值；

$W_{hi}(z)$ 为本次输入层的第h个神经元到第一个隐含层的第i个神经元之间的权值， $W_{hi}(z-1)$ 为上一次输入层的第h个神经元到第一个隐含层的第i个神经元之间的权值；

$F_{jk}(z)$ 为本次第二个隐含层的第j个神经元到输出层的第k个神经元之间的权值， $F_{jk}(z-1)$

1) 为上一次第二个隐含层的第j个神经元到输出层的第k个神经元之间的权值；

$G_{ij}(z)$ 为本次第一个隐含层的第i个神经元到第二个隐含层的第j个神经元之间的权值, $G_{ij}(z-1)$ 为上一次第一个隐含层的第i个神经元到第二个隐含层的第j个神经元之间的权值；

E_k 为输出层的第k个神经元均方误差函数； η 为学习率, z 为调整神经元的权值次数。

一种面向露天矿区无人驾驶车辆的制动力标定方法

技术领域

[0001] 本发明属于露天矿区自动驾驶技术领域,尤其涉及一种面向露天矿区无人驾驶车辆的制动力标定方法,解决矿用车辆的制动力标定问题。

背景技术

[0002] 在当代装备制造技术中,露天矿区无人驾驶技术发挥着重要的作用,是未来工业互联网的重要发展对象。露天矿区的无人驾驶相对于传统人工驾驶,不仅有效的保证了矿区作业人员的生命财产安全、有效降低驾驶员的人工成本,还提高了车辆利用率和作业效率。而在无人驾驶车辆中,制动性能好坏直接影响其安全性和平顺性,车辆制动性能包括制动力、制动力的平衡、车轮阻滞力等,制动力则是制动性能好坏的主要参数,当实际制动力小于标定的制动力一定范围时,说明无人驾驶车辆发生了制动失效故障,最终会影响露天矿区以及无人驾驶车辆的安全性,因此如何获取标定的制动力作为制动失效故障的判断标准至关重要。

[0003] 目前针对车辆制动力的标定方法有整车制动台架标定方法、车辆制动系统制动标定方法测试,往往通过昂贵或针对性的仪器进行制动力检测如:

[0004] 现有技术公开了一种滚筒反力式汽车制动检测台动态制动力的检测方法,在检测车辆的制动力时,将动态制动力测量装置安装在被检测车辆,进行整车的制动力测试;还公开了一种汽车制动总成制动力测试台,提供一种汽车总成制动力测试台,进行制动力测试时,只将制动盘和气动制动器进行相对滑动实验,计算所得扭矩数据即为制动盘的制动力。

[0005] 上述现有技术都是需要特制的仪器或装置进行制动力的获取,需要采购该仪器或装置,或者去相应制动力测试机构进行测试标定,这种车辆制动力标定的方法不仅增加了资金成本,还浪费了人力物力,而且进行制动台架测试的对象一般都是乘用车或者商用车,对于露天矿区非道路矿用车辆的制动台架较少,因此目前制动台架应用不是很广泛。而且通过制动台架不能获取到制动开度、速度和制动力之间的非线性关系,对露天矿区无人驾驶车辆的实际制动力对比项就不是很明确,最终会导致制动失效故障的诊断出现误判。

[0006] 目前针对车辆制动力的测量标定方法,采用制动台架进行整车或车辆制动系的制动力标定,需要专业人士使用专业的台架进行制动力测试,增加了人力成本、资金成本和时间成本。现在大部分制动台架是针对乘用车或商用车进行制动力标定测试,对非公路矿用车进行标定测试的较少,而且制动台架不能得到制动力与制动开度、速度三者之间的非线性关系,也就不能进行无人驾驶车辆的制动失效故障的诊断,即使得到三者的线性关系进行故障诊断也会导致出现误判。

发明内容

[0007] 为了解决上述已有技术存在的不足,本发明提出一种面向露天矿区无人驾驶车辆的制动力标定方法,通过获取不同制动开度、车速以及制动产生的制动力的数据集,采用

神经网络离线训练数据集的方式来获得三者之间非线性的关系,标定出不同制动开度和不同速度下的理想制动力,用于进行无人驾驶车辆的制动失效故障诊断。本发明的具体技术方案如下:

[0008] 一种露天矿区无人驾驶车辆的制动力标定方法,其特征在于,包括以下步骤:

[0009] S1:在车辆进行无人驾驶前,获取一系列制动开度及其对应的车速,根据车辆动力学公式计算出相应的制动力;

[0010] S2:对步骤S1获取制动开度、车速和制动力数据进行预处理,作为神经网络的训练样本数据集;

[0011] S3:根据步骤S2预处理后得到的训练样本数据集,将制动开度和车速作为BP神经网络的输入,制动力作为BP神经网络的输出,离线训练,得到三者的非线性关系;

[0012] S4:在车辆进行无人驾驶时,获取车辆当前的制动开度和车速信息;

[0013] S5:将步骤S4获取的信息输入步骤S3得到的非线性关系,经运算得到理想制动力,作为车辆产生实际制动力的对比标准,进行制动失效故障诊断;

[0014] 实际制动力与理想制动力的差距大于设定的阈值,说明车辆发生制动失效故障;

[0015] 实际制动力与理想制动力的差距小于设定的阈值,则进行下一时刻的理想制动力计算和制动失效故障的诊断。

[0016] 进一步地,所述步骤S1中,通过车辆的线控反馈获取制动开度数据,通过轮速计或GPS获取车速数据;制动力F的计算公式为: $F = G \cdot h \cdot \cos \alpha + G \cdot \sin \alpha + m \cdot a$,其中,G为作用于车上的重力, $G = m \cdot g$,m为汽车质量,单位为kg;g为重力加速度, $g = 9.8 \text{ m/s}^2$;h为滚动阻力系数; α 为坡度;a为车辆加速度。

[0017] 进一步地,所述步骤S2中,采用欠采样或过抽样进行数据处理。

[0018] 进一步地,所述步骤S3中离线训练BP神经网络的具体步骤为:

[0019] S3-1:网络初始化;

[0020] 将制动开度和车速作为输入X,制动力作为输出U,根据输入输出,确定网络输入层神经元数、隐含层神经元数、输出层神经元数;

[0021] 初始化输入层、隐含层和输出层神经元之间的连接权,具体为,隐含层为一层,则输入层第h个神经元与隐含层第i个神经元之间的连接权为 W_{hi} ,其中, $h = 1, 2, \dots, m$;隐含层为至少两层,则第一个隐含层第i个神经元到第二个隐含层第j个神经元之间的连接权为 G_{ij} ,以此类推, $j = 1, 2, \dots, n$;隐含层为一层,则隐含层第i个神经元到输出层第k个神经元之间的连接权为 F_{ik} ,其中, $k = 1$;隐含层为至少两层,则第二个隐含层的第j个神经元到输出层的第k个神经元之间的连接权为 F_{jk} ,其中, $k = 1$;

[0022] S3-2:计算隐含层输出;

[0023] 将步骤S2预处理后的样本数据集中制动开度和车速数据输入,隐含层为一层,则第一个隐含层第i个神经元接收到的输入 α_i 为:

$$[0024] \quad \alpha_i = f \left(\sum_{h=1}^m W_{hi} X_h \right) \quad (1)$$

[0025] 隐含层为至少两层,则第二个隐含层第j个神经元接收到的输入 β_j 为:

$$[0026] \quad \beta_j = f \left(\sum_{i=1}^m G_{ij} \alpha_i \right) \quad (2)$$

[0027] 其中, X_h 为输入层的第h个神经元, G_{ij} 为第一个隐含层的第i个神经元到第二个隐

含层的第j个神经元之间的权值；神经网络结构中神经元的传递函数采用双曲正切函数tansig,tansig传递函数f(x)为：

$$[0028] \quad f(x) = \frac{2}{1+e^{-2x}} - 1 \quad (3)$$

[0029] 其中,e为自然对数函数的底数,x为自变量；

[0030] S3-3:计算输出层输出；

[0031] 隐含层为一层,隐含层的最终输出作为输出层的输入,则输出层第k个神经元接收到的输入 U_k 为：

$$[0032] \quad U_k = f\left(\sum_{j=1}^n F_{ik}\alpha_j\right) \quad (4)$$

[0033] 隐含层为至少两层,则输出层第k个神经元接收到的输入 U_k 为：

$$[0034] \quad U_k = f\left(\sum_{j=1}^n F_{jk}\beta_j\right) \quad (5)$$

[0035] 其中, F_{jk} 为第二个隐含层的第j个神经元到输出层的第k个神经元之间的权值；

[0036] S3-4:均方误差函数；

[0037] 对训练例 (X^p, U^p) ,p为训练样本数,神经网络的输出为 U^p ,则经过神经网络的输出与实际输出的均方误差 E_p 为：

$$[0038] \quad E_p = \frac{1}{2}(\hat{U}_k^p - U_k^p)^2 \quad (6)$$

[0039] 其中, U_k^p 为第p个样本输入值输入到神经网络,经过神经网络计算后,第k个输出神经元输出的值, $\hat{U}_k^p$ 为第p个训练样本输出值；

[0040] S3-5:判断是否达到目标；

[0041] $E_p > \varepsilon$,则进行步骤S3-7的权值更新,否则训练结束, ε 为训练要求精度；

[0042] S3-6:判断是否达到最大训练次数,达到最大训练次数,训练结束；

[0043] S3-7:权值更新；

[0044] 根据梯度下降法,隐含层为一层,由均方误差函数 E_p 调整神经元的权值如下：

$$[0045] \quad \begin{cases} F_{ik}(z) = F_{ik}(z-1) - \eta \frac{\partial E_k}{\partial F_{ik}} \\ W_{hi}(z) = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial W_{hi}} = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial F_{ik}} \frac{\partial F_{ik}}{\partial W_{hi}} \end{cases} \quad (7)$$

[0046] 隐含层为至少两层,调整神经元的权值如下：

$$[0047] \quad \begin{cases} F_{jk}(z) = F_{jk}(z-1) - \eta \frac{\partial E_k}{\partial F_{jk}} \\ G_{ij}(z) = G_{ij}(z-1) - \eta \frac{\partial E_k}{\partial G_{ij}} = G_{ij}(z-1) - \eta \frac{\partial E_k}{\partial F_{jk}} \frac{\partial F_{jk}}{\partial G_{ij}} \\ W_{hi}(z) = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial W_{hi}} = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial F_{jk}} \frac{\partial F_{jk}}{\partial G_{ij}} \frac{\partial G_{ij}}{\partial W_{hi}} \end{cases} \quad (8)$$

[0048] 其中, $F_{ik}(z)$ 为本次第一个隐含层的第i个神经元到输出层的第k个神经元之间的权值, $F_{ik}(z-1)$ 为上一次第一个隐含层的第i个神经元到输出层的第k个神经元之间的权值；

[0049] $W_{hi}(z)$ 为本次输入层的第h个神经元到第一个隐含层的第i个神经元之间的权值, $W_{hi}(z-1)$ 为上一次输入层的第h个神经元到第一个隐含层的第i个神经元之间的权值；

[0050] $F_{jk}(z)$ 为本次第二个隐含层的第 j 个神经元到输出层的第 k 个神经元之间的权值, $F_{jk}(z-1)$ 为上一次第二个隐含层的第 j 个神经元到输出层的第 k 个神经元之间的权值;

[0051] $G_{ij}(z)$ 为本次第一个隐含层的第 i 个神经元到第二个隐含层的第 j 个神经元之间的权值, $G_{ij}(z-1)$ 为上一次第一个隐含层的第 i 个神经元到第二个隐含层的第 j 个神经元之间的权值;

[0052] E_k 为输出层的第 k 个神经元均方误差函数; η 为学习率, z 为调整神经元的权值次数。

[0053] 本发明的有益效果在于:

[0054] 1. 本发明在不需额外的仪器和装置进行制动力标定, 将采集的数据通过神经网络训练的方法进行制动力标定, 并有效的获取到制动开度、速度和制动力的非线性关系;

[0055] 2. 本发明的方法中对制动力标定的原始数据进行预处理, 考虑到采集的数据具有样本不均衡性, 采用BP神经网络算法的方式来获取制动开度、速度和制动力三者之间的非线性关系, BP神经网络算法不仅易于实现, 而且方便操作;

[0056] 3. 本发明的方法获取的非线性关系能够应用在无人驾驶车辆的制动失效故障的诊断中, 可以进行实时检测。

附图说明

[0057] 为了更清楚地说明本发明实施例或现有技术中的技术方案, 下面将对实施例中所需要使用的附图作简单地介绍, 通过参考附图会更加清楚的理解本发明的特征和优点, 附图是示意性的而不应理解为对本发明进行任何限制, 对于本领域普通技术人员来讲, 在不付出创造性劳动的前提下, 可以根据这些附图获得其他的附图。其中:

[0058] 图1为本发明的制动力标定方法简要流程图;

[0059] 图2为本发明的BP算法网络训练流程图;

[0060] 图3为本发明的神经网络结构图;

[0061] 图4为本发明的制动力标定方法详细流程图;

[0062] 图5为本发明的方法应用流程图;

[0063] 图6为本发明一个实施例的实测路线图;

[0064] 图7为本发明实施例的跑车结果图, 其中, (a) 为惯导输出的车辆速度, (b) 为惯导输出的车辆俯仰角, (c) 为车辆的制动开度百分比, (d) 为实际制动力与理想制动力的比值。

具体实施方式

[0065] 为了能够更清楚地理解本发明的上述目的、特征和优点, 下面结合附图和具体实施方式对本发明进行进一步的详细描述。需要说明的是, 在不冲突的情况下, 本发明的实施例及实施例中的特征可以相互组合。

[0066] 在下面的描述中阐述了很多具体细节以便于充分理解本发明, 但是, 本发明还可以采用其他不同于在此描述的其他方式来实施, 因此, 本发明的保护范围并不受下面公开的具体实施例的限制。

[0067] 本发明通过获取不同制动开度、车速以及制动产生的制动力的数据集, 采用神经

网络离线训练数据集的方式来获得三者之间非线性的关系,标定出不同制动开度和不同速度下的理想制动力,用于进行无人驾驶车辆的制动失效故障诊断。该制动力标定方法无需采购其他装置或仪器进行制动力的标定,方便快捷,适用性较强,且获得制动力与制动开度、速度三者的非线性关系,更易于针对无人驾驶车辆进行制动失效故障的诊断。

[0068] 具体地,如图1-3所示,一种露天矿区无人驾驶车辆的制动力标定方法,包括以下步骤:

[0069] S1:在车辆进行无人驾驶前,获取一系列制动开度及其对应的车速,根据车辆动力学公式计算出相应的制动力;

[0070] S2:对步骤S1获取制动开度、车速和制动力数据进行预处理,作为神经网络的训练样本数据集;

[0071] S3:根据步骤S2预处理后得到的训练样本数据集,将制动开度和车速作为BP神经网络的输入,制动力作为BP神经网络的输出,离线训练,得到三者的非线性关系;

[0072] S4:在车辆进行无人驾驶时,获取车辆当前的制动开度和车速信息;

[0073] S5:将步骤S4获取的信息输入步骤S3得到的非线性关系,经运算得到理想制动力,作为车辆产生实际制动力的对比标准,进行制动失效故障诊断;

[0074] 实际制动力与理想制动力的差距大于设定的阈值,说明车辆发生制动失效故障;

[0075] 实际制动力与理想制动力的差距小于设定的阈值,则进行下一时刻的理想制动力计算和制动失效故障的诊断。

[0076] 步骤S1中,通过车辆的线控反馈获取制动开度数据,通过轮速计或GPS获取车速数据;制动力F的计算公式为: $F=G*h*\cos\alpha+G*\sin\alpha+m*a$,其中,G为作用于车上的重力, $G=m*g$,m为汽车质量,单位为kg;g为重力加速度, $g=9.8m/s^2$;h为滚动阻力系数,由试验确定; α 为坡度;a为车辆加速度。

[0077] 步骤S2中,采用欠采样或过抽样进行数据处理。步骤S1中获取的数据集不能直接进行神经网络训练,原始数据集中是含有异常或错误的数据,最终会导致训练出的结果不具有特征性或应用性差,因此在步骤S2中需要对数据进行预处理。由于获取样本不均衡,会导致训练器在多数类精度较高,少数类的训练器精度很低,甚至造成训练结果不良。本发明采用过抽样方法——通过改变训练数据的分布来消除或减小数据的不平衡。

[0078] 如图2所示,步骤S3中离线训练BP神经网络的具体步骤为:

[0079] S3-1:网络初始化;

[0080] 将制动开度和车速作为输入X,制动力作为输出U,根据输入输出,确定网络输入层神经元数、隐含层神经元数、输出层神经元数;

[0081] 初始化输入层、隐含层和输出层神经元之间的连接权,具体为,隐含层为一层,则输入层第h个神经元与隐含层第i个神经元之间的连接权为 W_{hi} ,其中, $h=1,2, \dots, m$;隐含层为至少两层,则第一个隐含层第i个神经元到第二个隐含层第j个神经元之间的连接权为 G_{ij} ,以此类推, $j=1,2, \dots, n$;隐含层为一层,则隐含层第i个神经元到输出层第k个神经元之间的连接权为 F_{ik} ,其中, $k=1$;隐含层为至少两层,则第二个隐含层的第j个神经元到输出层的第k个神经元之间的连接权为 F_{jk} ,其中, $k=1$;

[0082] S3-2:计算隐含层输出;

[0083] 将步骤S2预处理后的样本数据集中制动开度和车速数据输入,隐含层为一层,则

第一个隐含层第i个神经元接收到的输入 α_i 为:

$$[0084] \quad \alpha_i = f\left(\sum_{h=1}^2 W_{hi} X_h\right) \quad (1)$$

[0085] 隐含层为至少两层,则第二个隐含层第j个神经元接收到的输入 β_j 为:

$$[0086] \quad \beta_j = f\left(\sum_{i=1}^m G_{ij} \alpha_i\right) \quad (2)$$

[0087] 其中, X_h 为输入层的第h个神经元, G_{ij} 为第一个隐含层的第i个神经元到第二个隐含层的第j个神经元之间的权值;神经网络结构中神经元的传递函数采用双曲正切函数 tansig, tansig传递函数 $f(x)$ 为:

$$[0088] \quad f(x) = \frac{2}{1+e^{-2x}} - 1 \quad (3)$$

[0089] 其中,e为自然对数函数的底数,x为自变量;

[0090] S3-3:计算输出层输出;

[0091] 隐含层为一层,隐含层的最终输出作为输出层的输入,则输出层第k个神经元接收到的输入 U_k 为:

$$[0092] \quad U_k = f\left(\sum_{j=1}^n F_{jk} \alpha_j\right) \quad (4)$$

[0093] 隐含层为至少两层,则输出层第k个神经元接收到的输入 U_k 为:

$$[0094] \quad U_k = f\left(\sum_{j=1}^n F_{jk} \beta_j\right) \quad (5)$$

[0095] 其中, F_{jk} 为第二个隐含层的第j个神经元到输出层的第k个神经元之间的权值;

[0096] S3-4:均方误差函数;

[0097] 对训练例 (X^p, U^p) ,p为训练样本数,神经网络的输出为 U^p ,则经过神经网络的输出与实际输出的均方误差 E_p 为:

$$[0098] \quad E_p = \frac{1}{2} (\hat{U}_k^p - U_k^p)^2 \quad (6)$$

[0099] 其中, U_k^p 为第p个样本输入值输入到神经网络,经过神经网络计算后,第k个输出神经元输出的值, $\hat{U}_k^p$ 为第p个训练样本输出值;

[0100] S3-5:判断是否达到目标;

[0101] $E_p > \varepsilon$,则进行步骤S3-7的权值更新,否则训练结束, ε 为训练要求精度;

[0102] S3-6:判断是否达到最大训练次数,达到最大训练次数,训练结束;

[0103] S3-7:权值更新;

[0104] 根据梯度下降法,隐含层为一层,由均方误差函数 E_p 调整神经元的权值如下:

$$[0105] \quad \begin{cases} F_{ik}(z) = F_{ik}(z-1) - \eta \frac{\partial E_k}{\partial F_{ik}} \\ W_{hi}(z) = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial W_{hi}} = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial F_{ik}} \frac{\partial F_{ik}}{\partial W_{hi}} \end{cases} \quad (7)$$

[0106] 隐含层为至少两层,调整神经元的权值如下:

$$[0107] \quad \begin{cases} F_{jk}(z) = F_{jk}(z-1) - \eta \frac{\partial E_k}{\partial F_{jk}} \\ G_{ij}(z) = G_{ij}(z-1) - \eta \frac{\partial E_k}{\partial G_{ij}} = G_{ij}(z-1) - \eta \frac{\partial E_k}{\partial F_{jk}} \frac{\partial F_{jk}}{\partial G_{ij}} \\ W_{hi}(z) = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial W_{hi}} = W_{hi}(z-1) - \eta \frac{\partial E_k}{\partial F_{jk}} \frac{\partial F_{jk}}{\partial G_{ij}} \frac{\partial G_{ij}}{\partial W_{hi}} \end{cases} \quad (8)$$

[0108] 其中, $F_{ik}(z)$ 为本次第一个隐含层的第 i 个神经元到输出层的第 k 个神经元之间的权值, $F_{ik}(z-1)$ 为上一次第一个隐含层的第 i 个神经元到输出层的第 k 个神经元之间的权值;

[0109] $W_{hi}(z)$ 为本次输入层的第 h 个神经元到第一个隐含层的第 i 个神经元之间的权值, $W_{hi}(z-1)$ 为上一次输入层的第 h 个神经元到第一个隐含层的第 i 个神经元之间的权值;

[0110] $F_{jk}(z)$ 为本次第二个隐含层的第 j 个神经元到输出层的第 k 个神经元之间的权值, $F_{jk}(z-1)$ 为上一次第二个隐含层的第 j 个神经元到输出层的第 k 个神经元之间的权值;

[0111] $G_{ij}(z)$ 为本次第一个隐含层的第 i 个神经元到第二个隐含层的第 j 个神经元之间的权值, $G_{ij}(z-1)$ 为上一次第一个隐含层的第 i 个神经元到第二个隐含层的第 j 个神经元之间的权值;

[0112] E_k 为输出层的第 k 个神经元均方误差函数; η 为学习率, z 为调整神经元的权值次数。

[0113] 将步骤S3中得到制动开度、速度和制动力的非线性关系,应用在无人驾驶车辆系统中,实时监测制动是否发生失效故障。如图5所示,在无人驾驶车辆在行驶的过程中,当前时刻获得制动开度和车速信息后,经过非线性关系的运算得到理想制动力,若实际制动力与理想制动力差距大于阈值,说明车辆发生了制动失效故障,若实际制动力与理想制动力差距小于阈值,则进行下一时刻的理想制动力计算和制动失效故障的诊断。

[0114] 下面通过具体实施例说明本发明方法的有效性。

[0115] 实施例1

[0116] 根据上述制动力标定方法进行实车试验,试验环境如表1所示,试验所需参数,如表2所示。测试场地在白云鄂博露天矿区,试验车辆为北方重型汽车集团中MT3600型矿用车辆,其车身质量为120000kg,在无人驾驶试验过程中通过传感器获取车辆的俯仰角和车辆速度,本实施例通过华测中CGI-610型惯导来获取车辆的俯仰角和车辆速度,滚动阻力系数 h 设置为0.02。

[0117] 表1试验条件

[0118]	试验场地	白云鄂博露天矿区
	试验车辆	北方重型汽车MT3600型矿用车辆
	传感器	华测惯导CGI-610

[0119] 表2试验参数

[0120]	车身质量	120000kg
	滚动阻力系数	0.02

[0121] 图6为本发明一个实施例的实测路线图,具体为白云鄂博露天矿区,图中右上角为测试的起始点,左下角为测试终止点,测试长度约为1178m。图7为本发明一个实施例的跑车结果图,图7(a)为惯导输出的车辆速度,图7(b)为惯导输出的车辆俯仰角,图7(c)为

车辆的制动开度百分比,图7(d)为实际制动力与理想制动力 的比值,图中虚线为实际制动力,实线为理想制动力。

[0122] 本实施例中选为下坡路段,主要为了在无人驾驶的过程中实施制动。具体坡度参考图7(b)中的惯导俯仰角(图中的惯导俯仰角是负值为下坡,正值为上坡)。

[0123] 通过图7(a)-图7(d)可以看出,在露天矿区,无人驾驶模式下,矿用车辆在 不同速度下,不同的制动开度下,理想制动力曲线与实际制动力曲线的趋势基本吻合,特别是在时间在150s-200s时,理想制动力的趋势和幅值与实际制动力大致相同。说明本发明的制动力标定方法得到的理想制动力与实际制动力基本相符。若矿用车辆在 行驶过程中产生的实际制动力在一段时间内与理想制动力差值较大时,车辆出现制动 失效的故障,便于后续对制动系的维修和监控。

[0124] 以上所述仅为本发明的优选实施例而已,并不用于限制本发明,对于本领域的技术人员来说,本发明可以有各种更改和变化。凡在本发明的精神和原则之内,所作的 任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

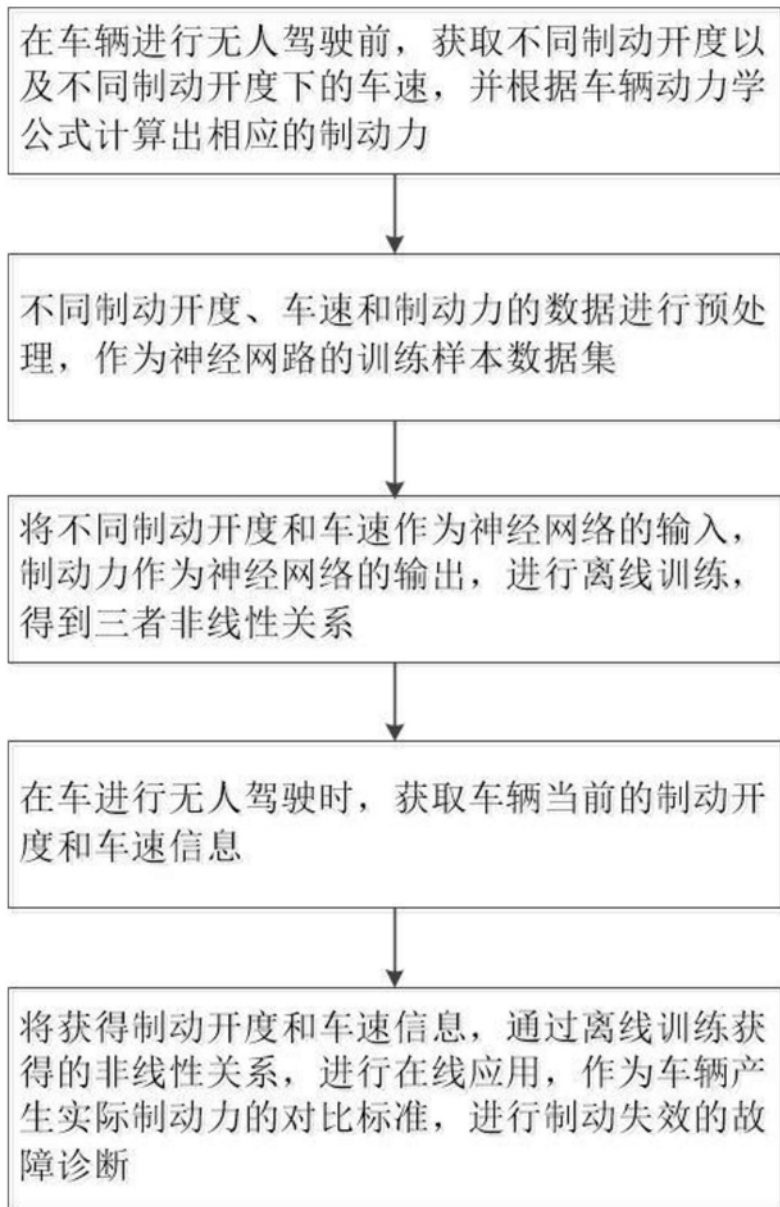


图1

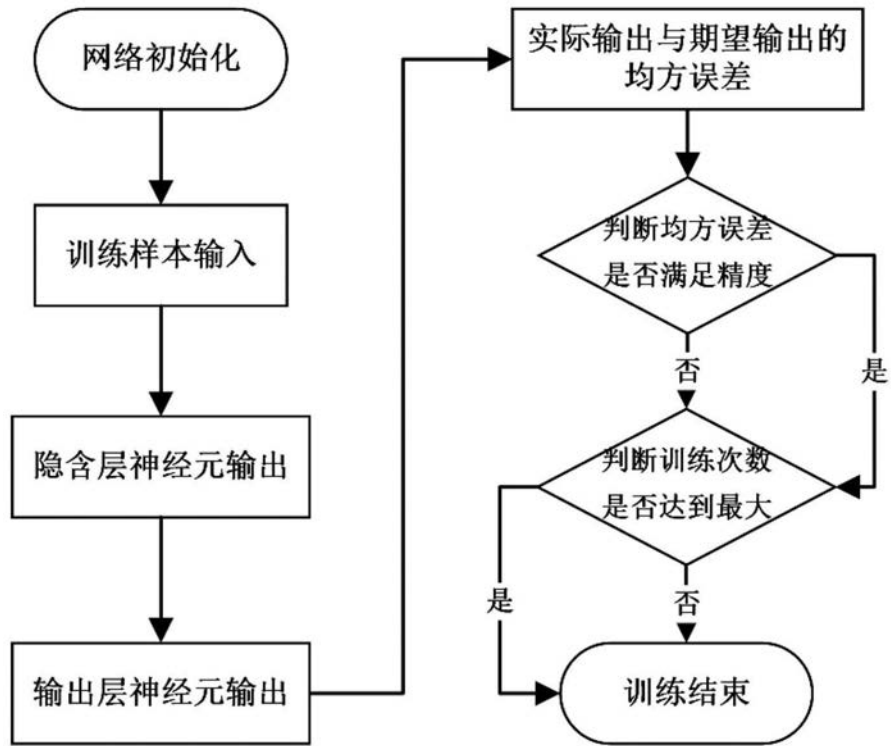


图2

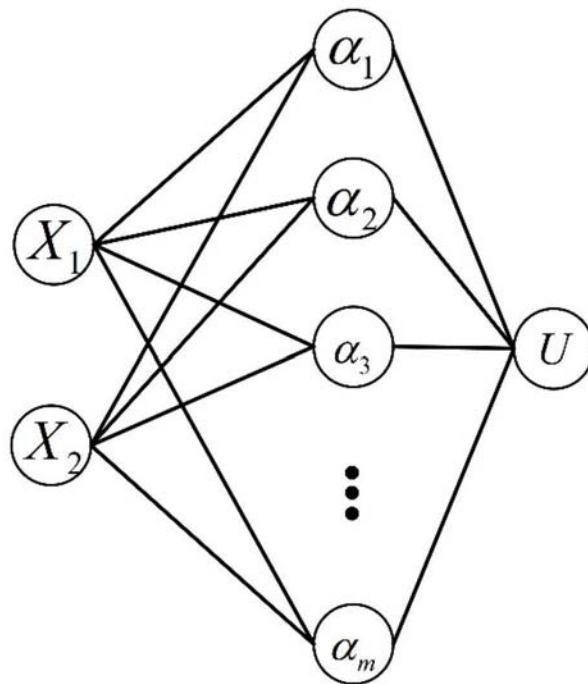


图3

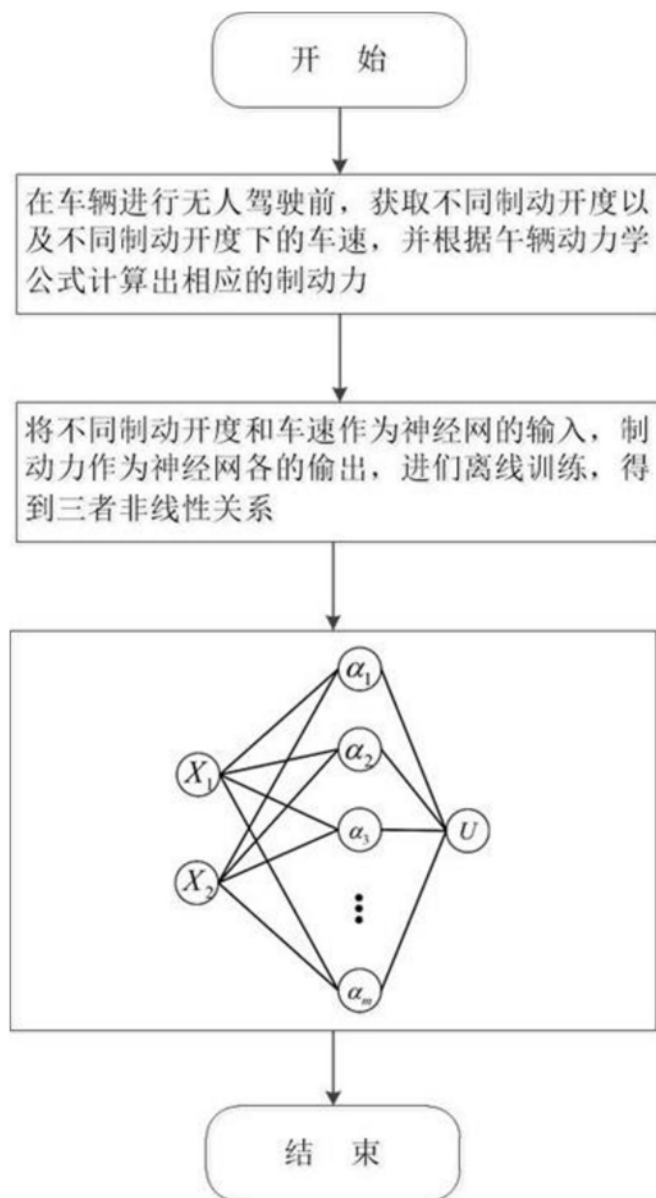


图4

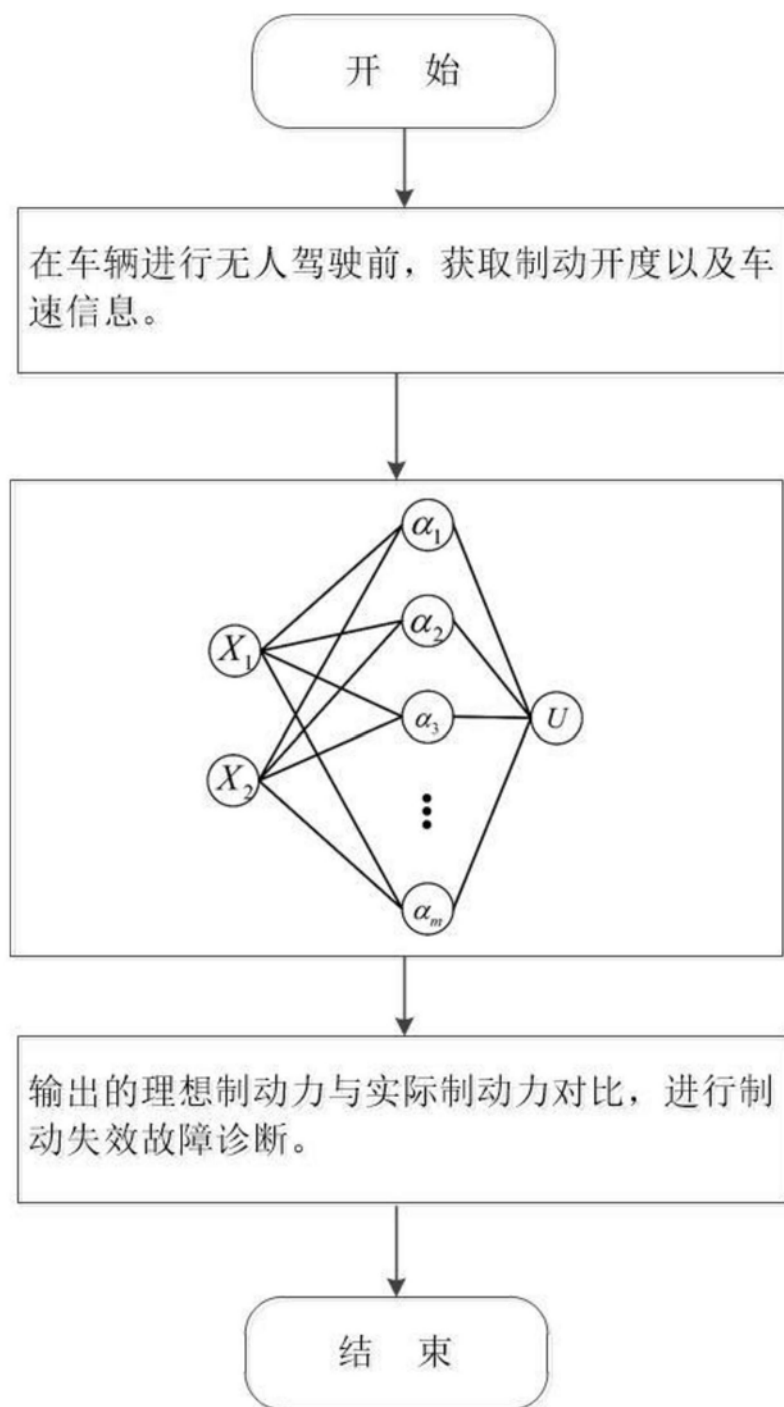
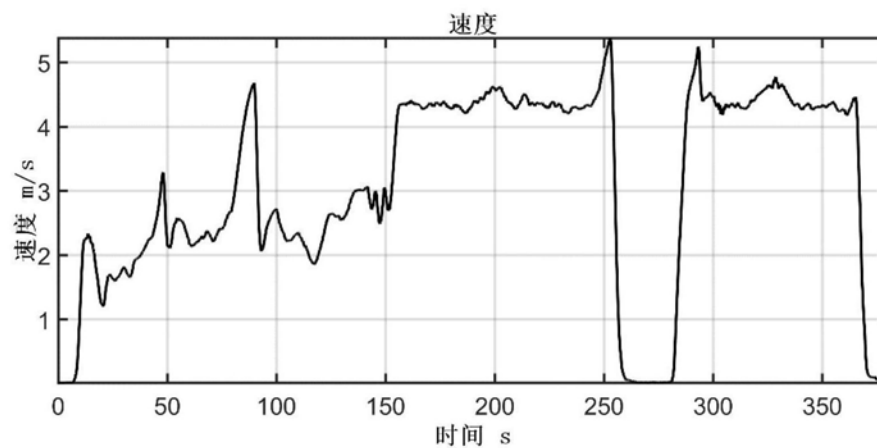


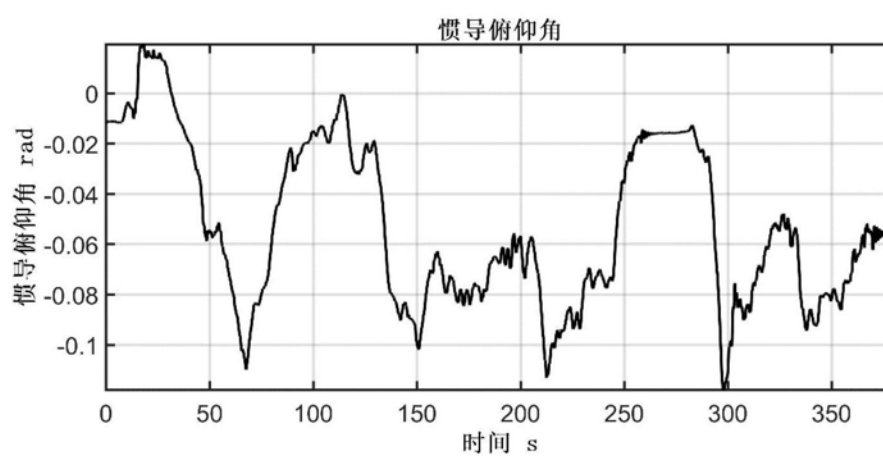
图5



图6



(a)



(b)

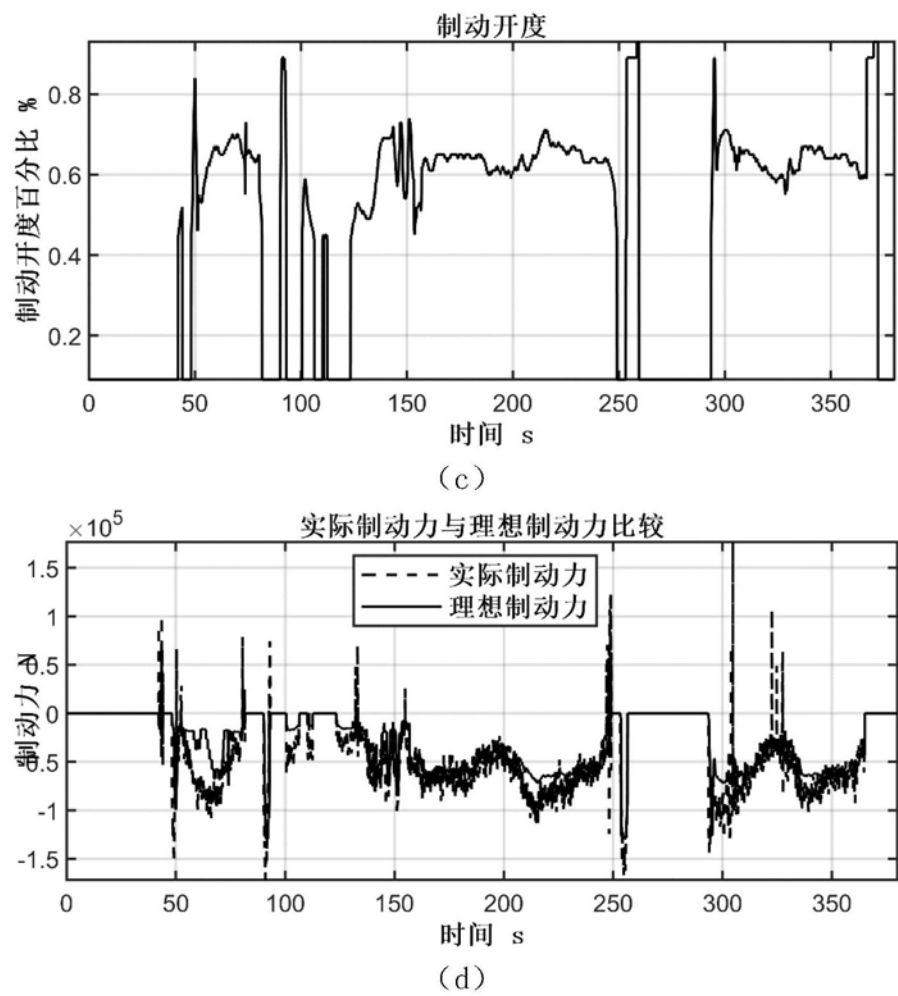


图7