

整车控制软件平台化设计及仿真

魏冰 陈添 刘震辉 闫岗 董彬

长城汽车股份有限公司

摘要: 本文依托某纯电车型的整车控制软件开发项目，研究整车控制软件平台化设计及仿真关键技术，通过 MATLAB/Simulink 仿真软件搭建基于整车控制软件平台化的仿真系统，对系统进行仿真研究，并对结果进行验证，为后续开发以及同领域整车控制开发提供参考。

关键词: 整车控制；软件平台化；仿真系统；设计

0 前言

随着汽车工业电动化和智能化的飞速发展，整车控制软件的重要性日益凸显，而软件平台化的开发更是实现汽车技术创新的关键路径。整车控制软件经历了从分散到集成、从固定功能到可升级的变革，MIL 测试是整车控制软件“V”型开发流程中的重要一环，因此，有必要对 MIL 测试的平台化进行研究。

整车控制软件平台化仿真的优势如下：允许在硬件实现之前进行功能和控制算法的验证，减少后续开发中可能出现的问题；虚拟仿真环境灵活性，在模拟环境中，对各种边界条件、异常工况及极端场景进行模拟，减少成本；平台化仿真，统筹软件开发，缩短开发周期，提高研发效率。

目前对于整车控制的软件平台化仿真实现方法主要有：模型驱动开发与验证、集成环境搭建以及跨域协同仿真等。本文以某纯电车型的整车控制软件平台化的设计开发和仿真为例，通过整车控制软件平台化设计及仿真系统，旨在为整车控制平台化领域提供理论依据和技术参考。

1 整车控制软件开发流程

整车控制软件开发经历了从代码开发到基于 MBD 设计开发的过程，基于 AUTOSAR 规范的整车控制软件通常要严格遵循其开发流程，从而确保软件质量、功能安全符合行业规范。一个典型的软件“V”型开发流程如图 1 所示。

在软件开发的整个生命周期过程中，整车控制软件的开发是一个严谨的闭环过程，涵盖从需求分析、软

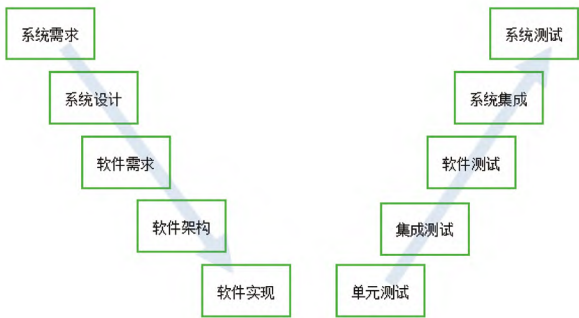


图 1 软件“V”型开发流程

件架构设计到量产的全过程。软件单元测试阶段一般是独立地检验每个模块，但在实际运行时，模块之间可能存在复杂的依赖关系，单元测试可能无法发现因模块间交互而产生的问题。软件集成测试阶段，一般指 HIL 测试，将独立开发的各个模块生成的代码按照接口规范进行集成编译，从而生成可执行 A2L 和 Hex 文件，下载到真实的硬件平台环境中进行下一步测试验证，但是 HIL 测试系统的建立和维护成本较高，需要专业的技术人员进行持续支持，并且由于测试环境的复杂性，调试和故障排查过程可能耗时较长。为了有效弥补单元测试和 HIL 测试的不足和缺陷，需要进行 MIL 测试，以获得更全面可靠的系统验证结果。

基于以上研究背景，在整车控制开发领域，软件平台化仿真系统基于 MIL 测试，考虑在独立开发各个模块初期进行仿真平台的搭建，与单模块的单元测试相比，进行整车全功能的仿真验证，从而利用软件平台化的仿真设计来匹配尽量多的车型。

2 软件平台化仿真架构

本文基于 MATLAB/Simulink 软件平台化仿真架构，

如图2所示,主要包含平台仿真模型 PlatModel,整车控制软件集成模型以及两者之间的交互 I/O 接口模块。

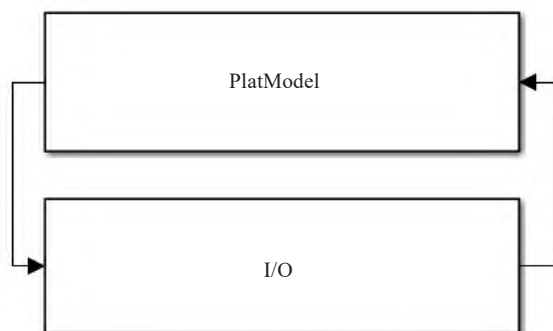


图2 软件平台化仿真架构

2.1 PlatModel 平台仿真模型

PlatModel 是被控对象的数学模型,它模拟了整车控制器所控制的实际物理系统或子系统的行为,整车控制器仿真模型主要包含五个仿真系统。

电池管理系统仿真模型,包括信号处理、输入/输出信号模块。电池管理系统不仅接收来自VCU的信号,同时向VCU反馈电池管理系统的状态信息,进行电池电压电流计算,单体电池的电压电流和温度等关键参数,充放电管理,并准确评估每个电池单元的荷电状态和健康状态,检测电池过充电、过放电、过温和过电流等异常情况,及时切断充放电回路,确保电池控制系统的安全。在高压系统启动过程中,在主正继电器进行闭合前,先进行预充控制,在预充阶段完成后,再进行主正继电器闭合,主负继电器闭合,保证高压回路的正常建立,实现整车高压上电控制。

车辆动力学仿真模型是该平台仿真模型的核心之一,基于整车动力学方程构建车辆纵向动力学仿真模型,并考虑滚动阻力、空气阻力、坡度阻力和路面的影响,根据整车接收到的转矩请求反馈,再结合路面参数,计算出电机转速信息。其中,总的环境阻力计算公式如下:

$$F_f = mgf \cos \theta \quad (1)$$

$$F_g = mg \sin \theta \quad (2)$$

$$F_a = 1/2 \rho v^2 C_d A \quad (3)$$

$$F = F_f + F_g + F_a \quad (4)$$

式中, F 为总的环境阻力; F_f 为滚动阻力; F_g 为坡道阻力; F_a 为空气阻力; m 为整车质量; g 为重力加速度; θ 为路面坡度角; ρ 为空气密度; v 为车辆车速; C_d 为空气阻力系数; A 为迎风面积。

整车驱动转矩计算公式如下:

$$F_{trq} = \frac{T_{in}}{r} \quad (5)$$

式中, T 为整车转矩请求; i 为传动比; η 为传动效率; r 为车轮半径。

电机转速计算公式如下:

$$n = \frac{1000v}{60 \times 2\pi r i} \quad (6)$$

式中, n 为电机转速; v 为车辆车速; r 为车轮半径; i 为传动比。

电机控制系统仿真模型,包括主要信号计算和电机控制系统输入/输出信号。电机控制系统根据整车控制转矩请求计算出电机控制转矩,电机控制模式判断,电机转速计算,母线电压和电机功率计算,电机最大可用驱动转矩和最大可用回收转矩计算,电机失效状态等。电机控制系统能够控制电机进行工作模式的切换。转矩模式下,电机控制系统响应整车控制系统的转矩请求;速度模式下,将根据目标转速和实际转速进行PI控制,输出电机的实际转矩。

车身控制器仿真模型,包括接收输入信号和输出控制信号,用于感知车辆状态和当前环境条件。车身控制系统接收VCU的信号,进行车门、车灯、车窗和刮水器控制,上下电时序控制,整车驾驶员意图输入的模拟,网络管理报文控制,空调热管理系统控制等的模拟,输出控制信号反馈给VCU。

电子稳定控制系统仿真模型,主要包含底盘稳定性(牵引力控制系统/防抱死制动系统/动态稳定系统等)功能的控制,底盘协同制动系统功能控制,响应整车控制器的控制指令,反馈各功能状态,底盘辅助驾驶类功能激活时与整车控制器的交互信号的计算。

2.2 整车控制软件模型的集成

本文所设计的整车控制系统架构如图3所示,主要包括整车上下电控制系统、接口信号处理系统、硬线信号控制系统、转矩控制系统、热管理系统、显示控制系统、能量管理系统及故障管理系统。

整车上下电控制系统在整车状态机进行车辆状态



图3 整车控制系统架构

判断或者充电意图判断后,检测直流一直流变换器、电机控制器、充电控制器和电池管理系统的状态是否满足驾驶员的操作意图,再发送控制指令到相应的控制器。在充电枪连接时,VCU 执行相应的充电流程,根据充电连接状态,将交流电请求发送给充电控制器进行车辆充电,将直流电请求发送给电池管理系统进行车辆充电并和充电桩进行握手交互,根据不同的充电状态控制 LED 灯点亮。

接口信号处理系统,VCU 接受 CAN/LIN 线各控制器总线信号输入,对总线信号进行合理性检测和超范围处理,实现控制系统的高内聚和低耦合。

硬线信号控制系统,通过传感器及相关硬线检测,包括充电口信号、供电电压、碰撞信号及高压互锁信号检测等。

转矩控制系统,在车辆处于高压系统闭合状态下,VCU 通过采集加速踏板、制动踏板、驾驶模式和回收等级等信号,判断驾驶员的需求转矩,并将整车转矩请求发给电机控制器,从而控制车辆行驶。

热管理控制系统,通过电桥智能温度控制系统、电池智能温度控制系统、预测热管理和多模热泵系统等功能,实现热管理控制系统功能。

能量管理系统,计算和分配车辆在行驶过程和充电过程中,当前可用功率限制和转矩限制,协调各部件实现能量管理系统控制功能。

显示控制系统,包括人机交互界面、加速踏板信号处理、计算车速和加速度,识别驾驶员意图并显示车辆的状态信息,进行指示灯、文字、能量流的显示和回收等级显示等,在车辆发生故障时发出提示和警告,驾驶员通过仪表信息判断整车状态。

故障管理系统,包括故障管理监控硬线故障、控制器内部故障和外部控制器故障等,VCU 检测和记录故障信息,激活故障反应,进行降级或者关闭相关功能。

各个功能在软件开发过程中按照功能需求分为不同的 SWC 进行开发,因此需要将所有 SWC 集成到一个模型环境中才能进行系统级别测试,主要分为整车控制模型组合、信号流的定义和连接、模型调度的设置和仿真模型配置与设定几个步骤。

整车控制模型的组合,将全部的整车控制子系统模型按照设计要求连接起来,形成一个完整的系统模型,可以通过拖拽方式将子系统模块嵌入到主系统模型中,或者通过引用模型的方式将外部模型链接进来。本方案采用引用模型的方式将所有子系统模型组合到一起,该

方法具有显著的优点:子系统模型内部逻辑可以更新,而主系统模型框架可以保持不变,可以使整个系统工程更易去维护。

整车控制器的所有交互信号一般分为以下几类:CAN/LIN 总线信号、硬件 I/O 采样输入信号、底层控制信号和子系统之间的交互信号,如图 4 所示的信号流连接和交互流程,首先将每个子系统模型的所有输出信号分层组合为一个 O_BUS,再将其和其他外部信号组合到 VCU_BUS 中,保证所有子系统模型的输入信号都能直接从 BUS 中获取,按照此方法连接可以实现系统模型的高效运行。

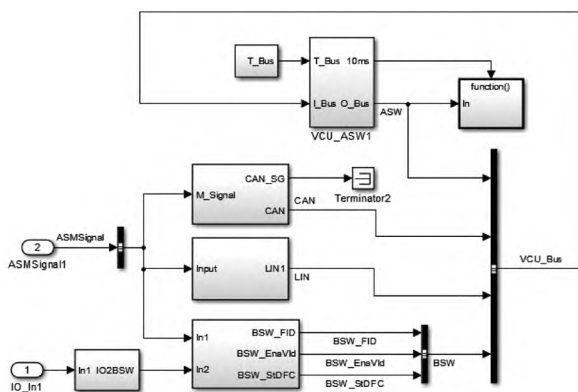


图 4 信号流连接和交互

调度周期指的是控制器内各个功能模块执行控制算法的时间间隔或者频率,在整车控制软件的开发中,调度周期的设置是至关重要的,因为它决定了控制器对车辆状态监测、控制指令发出、故障检测和处理等方面的实时性。整车控制器的各个子系统模型按照功能定义,有不同的调度周期,遵循的原则是:任务或者功能优先级越高,调度周期越短。例如:5 ms 任务调度周期主要包含一些硬线信号、加速踏板信号、12 V 电源信号和碰撞信号等;10 ms 任务调度周期包含大多数功能的调度周期;50 ms 任务调度周期一般用于显示模块,优先级相对较低;100 ms 任务调度周期主要用于循环故障码和历史故障码发送等。目前项目中存在 5 ms、10 ms、50 ms 和 100 ms 几种任务调度周期,为了保证每个子系统模型在系统模型仿真时的运行周期和实际代码在控制器内部运行周期一致,可以利用事件触发的逻辑或者 Stateflow 图表,利用 1ms 的基准步长,实现每隔 5 ms、10 ms、50 ms 和 100 ms 触发相应的调度计算和系统状态更新,如图 5 所示为控制系统任务调度图,该调度方案实现了各个功能模块协同工作,合理分配资源,保证系统高效稳定地运行。

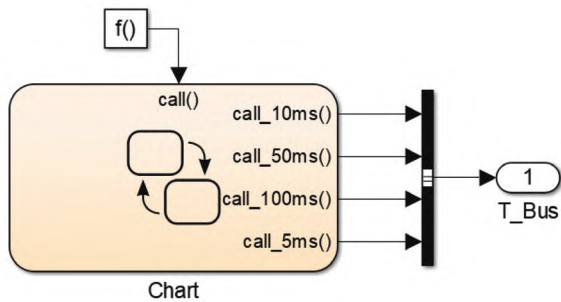


图5 控制系统任务调度

仿真模型配置与设定，所有子系统模型需要有统一的模型配置参数才能进行控制系统的仿真，具体参数设置需要根据具体项目特点来决定，例如，车辆动力学参数包括：整车质量、滚动半径、风阻系数和转动惯量等；动力系统参数包括：电池容量、电压平台、减速比、电机峰值扭矩和峰值功率等；仿真平台参数包括：求解器配置等。

2.3 整车模型和 PlatModel 的接口信号交互

整车控制模型与 PlatModel 之间的信号交互主要通过 CAN/LIN 等通信矩阵来实现，使用自动化的脚本，将整车控制的输入信号和输出信号与总线信号进行自动化的连接和配对，从而实现控制系统仿真模型的信号流连接，主要分为以下几个步骤：

(1) 通信矩阵的解析，通过 DBC 文件提取 ECU 节点、报文和信号定义。如图 6 所示为 DBC 文件提取的自动化脚本。

(2) 自动生成总线对象，根据通信矩阵动态生成 Simulink 总线，为每个 ECU 节点的发送报文创建总线。如图 7 所示为自动生成总线脚本。

(3) 模型接口配置，在模型中添加 CAN Pack 模块，绑定 ECU 节点与信号。如图 8 所示为模型接口配置脚本。

(4) 信号映射与自动化连接，通过名称映射或者规则匹配，跨模型连接信号，如图 9 所示，自动化脚本控制逻辑是：首先，在 DBC 中查找从 VCU 发送的信号，匹配发送端信号；在 DBC 中查找 PlatModel 接收的信号，匹配接收端信号；在 Simulink 模型中连接发送模块到接收模块。

3 整车控制系统需求和通信矩阵的平台化

为了实现整车控制软件平台化仿真，系统需求和通信矩阵的平台化是必不可少的，只有将整车功能需求和通信矩阵进行平台化的设计开发，PlatModel 中各

```
% 示例：加载DBC文件并提取信号
db = canDatabase('vehicle_signals.dbc');
ecuNodes = (db.ECUs.Name); % 获取所有ECU节点（如VCU, PlatModel）
messages = db.Messages; % 获取所有CAN报文

% 提取信号列表
signalList = [];
for msg = messages
    for sig = msg.Signals
        signalList = [signalList; struct(...
            'Name', sig.Name, ...
            'ECU', sig.ECU, ... % 发送/接收ECU节点
            'StartBit', sig.StartBit, ...
            'Length', sig.Length, ...
            'DataType', sig.DataType ...
        )];
    end
end
```

图6 DBC文件提取脚本

```
% 为ECU节点“VCU”生成发送报文的总线对象
vcuMessages = messages(strcmp(messages.ECU, 'VCU'));
for msg = vcuMessages
    busElements = [];
    for sig = msg.Signals
        elem = Simulink.BusElement;
        elem.Name = sig.Name;
        elem.DataType = sig.DataType; % 如'uint8', 'single'
        elem.Dimensions = 1;
        elem.Description = sprintf('StartBit:%d, Length:%d', ...
            sig.StartBit, sig.Length);
        busElements = [busElements; elem];
    end
    % 创建总线对象VCU_Bus
    busObj = Simulink.Bus;
    busObj.Elements = busElements;
    assignin('base', ['Bus_', msg.Name], busObj); % 例如Bus_VCU_Msg1
end
```

图7 自动生成总线脚本

```
% 在整车模型（VehicleControl）中添加CAN报文发送模块（VCU节点）
canTxBlockPath = 'VehicleControl/CAN_Tx_VCU';
add_block('canlib/CAN Pack', canTxBlockPath);
set_param(canTxBlockPath, 'Database', 'vehicle_signals.dbc', ...
    'Message', 'VCU_Msg1');

% 将VCU输出信号绑定到CAN Pack模块
signalsToPack = ['ThrottleCmd', 'BrakePressure'];
% 需要发送的信号名
for i = 1:length(signalsToPack)
    sigName = signalsToPack(i);
    add_line('VehicleControl', [sigName '/1'], ['CAN_Tx_VCU/' num2str(i)]);
end
```

图8 模型接口配置脚本

```
% 连接整车模型与PlatModel的总线信号
for sig = signalList
    if strcmp(sig.ECU.Sender, 'VCU') && strcmp(sig.ECU.Receiver, ...
        'PlatModel')
        % 在VehicleControl中找到发送信号块
        txBlock = find_system('VehicleControl', 'Name', sig.Name);
        % 在PlatModel中添加CAN Unpack模块并连接
        canRxBlockPath = 'PlatModel/CAN_Rx_Plant';
        if isempty(find_system('PlatModel', 'Name', 'CAN_Rx_Plant'))
            add_block('canlib/CAN Unpack', canRxBlockPath);
            set_param(canRxBlockPath, 'Database', 'vehicle_signals.dbc', ...
                'Message', sig.Message);
        end
        % 连接CAN解包信号到PlatModel输入端口
        add_line('PlatModel', [canRxBlockPath '/' sig.Name], ...
            [sig.Name '_in/1']);
    end
end
```

图9 信号连接脚本

控制系统仿真模型才有可能设计为标准化的功能模块，否则，不同车型项目则需要对 PlatModel 进行适应性开发迭代，这样会花费大量的调试和测试时间，而这与当前主流的快节奏的敏捷开发流程是相违背的，因此整车控制系统需求和通信矩阵的平台化是软件平台化仿真设计的前提。

基于降低开发成本、提升跨车型的复用性的目标，并确保控制系统的可扩展性与兼容性，实现整车控制系统需求和通信矩阵平台化的一般思路需进行系统需求分层和通信矩阵平台化设计。

3.1 系统需求分层平台化控制

将整车控制需求分为平台层（车型无关）和非平台层（车型相关），平台层定义跨车型需求，如基础诊断协议，非平台层定义配置车型特有参数，如电池容量、电机功率等参数或者选装功能，利用预编译宏或者配置文件动态加载功能模块，通过变体管理定义平台功能的必选，可选和互斥关系，例如：两驱车型不启用转矩分配功能，四驱车型启用转矩分配功能；高性能车型启用弹射起步功能，基础车型禁用弹射起步功能。

3.2 通信矩阵的平台化设计

基于行业标准，使用统一的通信矩阵格式，定义信号的编码方式、刷新周期和物理量纲，并将信号分为平台基础信号和可扩展信号，例如：车速、挡位、加速踏板开度和电池电量等属于平台基础信号；转矩分配、弹射起步状态等是可扩展信号，同时将信号发送分为周期型和事件型，例如：转矩、车速信号定义为周期型循环发送，而故障灯点亮定义为事件型发送，从而减少总线的负载率。

3.3 分层架构设计

基于 AUTOSAR 分层模型，分为应用层、RTE 层和基础软件层，其中，应用层主要是车型无关的控制算法与车型相关的我参数进行分离，例如：最大车速限制 PI 控制算法；RTE 层通过虚拟功能总线屏蔽相关硬件差异；基础软件层统一驱动抽象，例如：CAN 收发器驱动适配不同供应商硬件。

3.4 模块化组件开发

将通用模块独立为 lib 库文件，例如：转矩控制核心算法，硬件接口模块通过适配器兼容不同的 ECU，例如：电机控制接口、电池管理控制系统接口和定义统一的接口，如：ADC 读取、PWM 接口等，便于支持硬件快速切换，支持新功能添加和技术升级。

4 整车控制系统仿真验证

基于整车控制系统仿真平台，在不同工况下对此系统进行模型仿真，以此验证仿真系统的有效性。

4.1 整车上电时序仿真

整车上电时序仿真结果如图 10 所示。上电时，整车从休眠到唤醒后，整车状态进入 Power up；主继电器控制状态，预充继电器先进行闭合，到主正继电器闭合，主负继电器闭合，预充继电器断开，BMS 状态从 Standby 进入 Ready 状态；VCU 状态从 Precharge 到 Standby，再跳到请求转矩控制模式；MCU 状态从初始状态到 Precharge 再到 Standby，再执行整车请求的转矩模式，至此，整车整个上电仿真过程完成。

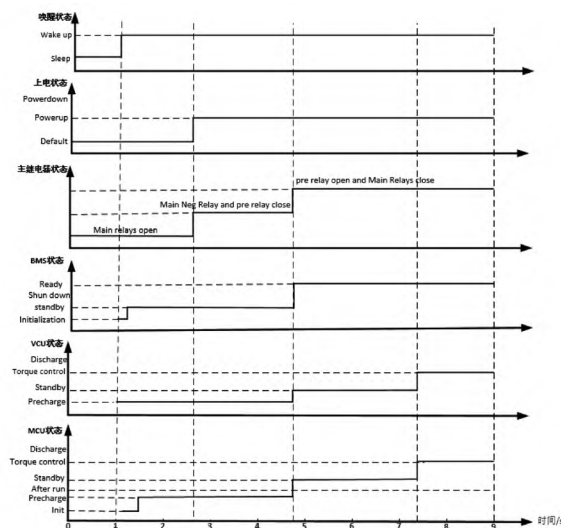


图 10 整车上电时序仿真结果

4.2 整车蠕行转矩控制仿真

整车蠕行转矩仿真结果如图 11 所示，以前进挡 D 位滑行进入蠕行为例，在 8% 上坡，无制动力请求，在车速降到蠕行激活的车速阈值时，则蠕行使能标志位置位，蠕行转矩会根据蠕行目标车速和实际车速进行 PI 调节进行加载，在规定时间内达到蠕行的目标车速。

5 结语

本文基于模型开发，研究基于整车控制软件平台化仿真技术，在 MATLAB/Simulink 环境中搭建了整车控制平台化仿真系统，包括平台仿真模型、整车控制软件模型的集成、整车模型和 PlatModel 的接口信号交互以及整车控制系统需求和通信矩阵的平台化要求，在此仿真环境中分别模拟车辆上电，蠕行工况下进行

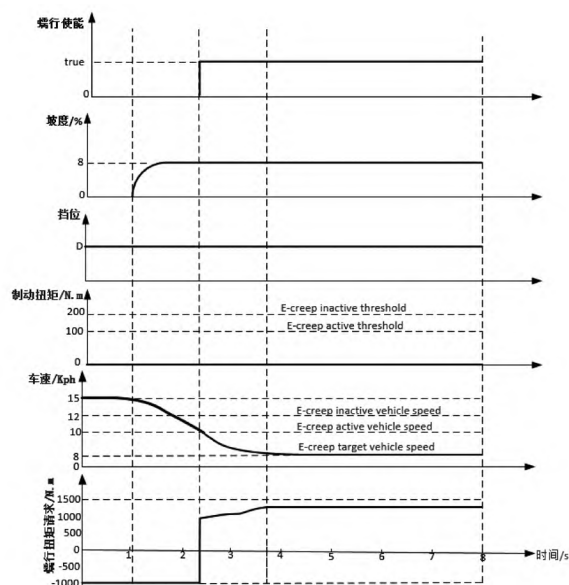


图 11 整车蠕行转矩仿真结果

整车上电时序, 蠕行转矩控制的仿真实验验证, 仿真结果表明, 所搭建的整车控制平台化仿真设计系统具有可行性和有效性, 为后续整车平台化设计开发提供基础的参考和依据。

综上所述, 整车控制软件平台化仿真对整车策略和算法的研究优化意义深远, 节省资源和研发时间,

大大地缩短开发周期, 在实际工程中具有很强的实用价值。

参考文献

- [1] 张晓先. 基于 AUTOSAR 规范的车用电机控制器软件开发 [M]. 北京: 科学出版社, 2020.
- [2] 魏学哲. 汽车电子与软件架构 [M]. 北京: 机械工业出版社, 2023.
- [3] 孙忠潇. Simulink 仿真及代码生成技术入门到精通 [M]. 北京: 北京航空航天大学出版社, 2015.
- [4] 薛定宇, 陈阳泉. 基于 Matlab/Simulink 的系统仿真技术与应用 [M]. 北京: 清华大学出版社, 2008.
- [5] 许同盟. 纯电动汽车整车控制器设计与控制策略研究 [D]. 济南: 山东大学, 2019.
- [6] 张海燕. 电动汽车关键技术研究 [J]. 汽车工程师, 2021(4): 19-24.
- [7] 黄其, 薛利昆, 罗玲. 电动汽车整车控制器设计试 [J]. 自动化与仪表, 2019, 34(3): 14-18.
- [8] 王胜达, 海涛, 董利莹. 基于 Simulink 的纯电动汽车整车控制系统设计 [J]. 科学技术与工程, 2020(7): 2885-2891.

(上接第 45 页)

燃油车水平; 最小转弯直径降至 16.7 m, 降幅 14.4%, 电池布置紧凑度指数提升 51%。同时, 多目标综合评价模型的 P 值从 0.15 升至 0.28, 突破了电动重卡机动性与长续驶的矛盾壁垒。

未来, 相关研究应聚焦于固态电池量产成本控制和电驱桥可靠性提升, 加速该方案工程化落地。同时, 探索电池底盘一体化 (CTC) 技术与线控转向系统融合应用, 以实现空间优化基础上的转向稳定性与紧凑化双重突破, 推动电动重卡向零碳高效物流核心装备发展。

参考文献

- [1] 中国汽车工程学会. 节能与新能源汽车技术路线图 2.0 [M]. 北京: 机械工业出版社, 2021.
- [2] 李江, 陈奇锋, 张伟. 纯电动重卡动力总成结构研究 [J]. 汽车实用技术, 2021(19): 17-19.
- [3] 国家标准化管理委员会, GB/T 1589-2016 道路车辆外廓尺寸、轴荷及质量限值 [S]. 北京: 中国标准, 2016.
- [4] 国家标准化管理委员会, GB/T 35782-2017 道路甩挂运输车辆技术条件 [S]. 北京: 出版社, 2017.