论文题目：CAN-bus 协议逆向分析系统的设计与实现
学生姓名：陈秋言
指导教师：马小博

摘 要

随着汽车上的电子控制系统的大量增加，整个车辆系统更加冗杂。为了解决这一问题，目前应用最为广泛的解决方法是采用控制器局域网络，即所谓的 CAN-bus 系统。而对 CAN-bus 协议进行逆向分析，获取协议具体含义信息并在 PC 上进行可视化显示是值得研究的问题。

本文采取的逆向分析方法是对采集的数据手动进行指令执行序列分析，进行滤波处理后逐一分析各帧数据的具体含义，实现对以车窗、车门窗锁等部件为代表的 CAN-bus 舒适总线进行协议逆向分析，得到舒适总线部分协议的具体指令含义。之后根据这些结果设计出 CAN-bus 协议逆向分析系统，自动对采集的数据进行逆向分析，并且对分析结果进行可视化显示。而为了保证系统实时分析的功能及加快逆向分析效率，之后又对系统添加了直接从分析仪中读取数据的功能，同时增加数据发送功能并重新设计了系统的界面。最后，对该系统的数据接收、数据发送和逆向分析三个部分的功能进行了测试，各项结果表明系统已达到预期目标，基本能实现对 CAN-bus 协议的逆向分析。

关 键 词：协议逆向；CAN-bus；数据统计分析

Title: Design and implementation of CAN-bus protocol reverse-engineering system
Name: Qiuyan Chen
Supervisor: Xiaobo Ma

ABSTRACT

With the increase of electronic control systems in automobiles, the vehicle system is more and more complex. Among these solutions to the problem of excessive body wiring harness, CAN-bus system is advertised as the one most widely taken. Now how to reverse-engineer the CAN-bus protocol and get the specific meaning of the protocol, is an issue that is worth studying.

In this thesis, to realize protocol reverse-engineer of CAN comfort bus and get the specific meaning of part protocol of comfort bus, the data collected and instruction execution sequence are analyzed manualy frame by frame after filtered. Then, according to the analysis results, a CAN-bus protocol reverse analysis system would be designed, which reverses the collected data and visualizes the analysis results.In order to ensure the real-time analysis function of the system, there would be two functions added to the system, which are data receiving and data sending. By the way, the system interface would have also been redesigned. Finally, the data receiving, data sending and reverse analysis of the system would be tested, and the results have been achieved expected goal. The system can achieve reverse analysis of CAN-bus protocol basically.

KEY WORDS: Protocol reverse engineering; CAN-bus; Statistic analysis of data

目 录

1 绪论

本章首先说明 CAN 总线应用广泛，从而引出本文的研究 CAN-bus 协议逆向的意义。之后简要介绍了一些协议逆向的方法以说明业界的研究现状和研究成果。最后提出本文的研究内容和组织结构

1.1 研究背景及意义

如今汽车技术不断发展，人们在追求车辆动力性和操控性能的同时，也开始舒适度和安全性能也提出了更高的要求。这导致车内越来越多的集成电路，所以需要更多的电子控制系统来完善了汽车的各项功能，提升汽车的智能化程度，如安全气囊装置、车载电网控制装置、电动门窗装置、防抱死制动装置等。然而，同时电子控制模块的数量也以惊人的速度增加。这导致这些电子控制模块之间的数据交换也随之增加。而在传统的数据交换形式下这会导致车身线束的增加，不但增加了制造成本，还占用空间并且增加了整个车身重量，此外还会提高因线束老化而引起电气故障的几率，从而降低系统的可靠性。[1]

解决这个问题的关键主要是通过计算机网络技术，将车载控制模块与车载网络连接起来，从而实现数据的高效传输。采用控制器局域网络（即 CAN-bus 系统）等总线技术，车载网络控制系统能够同时处理大量来自集成电路的信息和执行其各种功能以及不断增加的数据交换。

而这次设计的主要目的就是对 CAN-bus 总线上的一些实时数据进行捕获，同时立即对其进行协议逆向分析，并在 PC 端以简明的形式直接展示出来，实现在 PC 端对汽车上相关指令操作的监控分析，同时也能在 PC 端向总线发送数据，以实现对 PC 端对汽车的一些相关操作。

1.2 国内外研究现状

本文的核心研究内容是实现协议逆向。协议逆向指的是在没有具体协议相关文件的情况下，通过各种方法由传输的数据入手，最终推算出协议具体含义的方法。[10]

而如今国内外的研究学者对于协议逆向的方法主要分为两类，报文序列分析和指令执行序列分析。[2]

1.2.1 报文序列分析

报文序列分析技术的原理是报文格式样本之间的具有一定相似性的特征，采用比对、递归聚类、关联度映射等措施，可以挖掘推测协议的格式、语义等信息。这种方法的理论基础是每个报文样本是报文格式的一个实例，而同一报文格式的多个样本具有一定的相似性. [3]

报文序列分析技术的主要流程是，首先先确定进行比对的序列集合，并将这些成对的序列分别进行比对从而得到初始距离矩阵；之后根据矩阵生成系统进化树；最后根据生成的进化树将序列与之进行比对并输出结果。[4]

1.2.2 指令执行序列分析

指令执行序列分析技术是指用于数据解析过程中对指令执行序列进行分析的一种技术。这种类型的研究理论基础是协议实体是否按照格式规范解析报文，即通过监视协议实体对报文的处理过程和各个报文片段的使用方式，可以获得报文的结构以及语义信息。[5]

指令执行序列分析技术目前基于动态污点分析技术。[9]它是一种在指令级别上跟踪和分析数据处理过程的技术。这种技术主要方法是通过识别出应用程序接收到的协议信息来获取程序执行轨迹，结合协议各个字段的分析策略，从而得出协议的结构，达到协议逆向的目的。[6]

1.3 本文工作及论文结构

本文以上述背景为基础，对 CAN-bus 协议进行研究。通过相关仪器对 CAN-bus 网络进行数据采集，通过滤波手动对数据帧与示教板进行协议逆向分析。最后，先后提出基于文本文档对 CAN-bus 协议逆向分析的可视化页面展示和实时对 CAN-bus 协议逆向分析的可视化页面展示。

该系统的优势其在于对数据进行逆向分析的工作都是采用控制变量法对指令执行序列手动进行分析，准确率较高。而在最后的可视化页面中，系统读取后进行分析的速度也是比较快的，基本能做到实时分析。不足之处在于在一开始的分析过程中效率比较低。

本文主要围绕 CAN-bus 协议逆向分析系统展开，总共分为六章，其中第三、四、五三章为本文的核心内容。文章的逻辑组织结构如图 1-1 所示。

第一章 绪论 本章阐述了相关研究背景，列举了国内外研究逆向分析的主要方法，最后总结了本文工作并给出了论文的总体结构。

第二章 相关背景概述 本章主要介绍了逆向分析的对象 CAN-bus 和 J1939 协议。

第三章 数据采集与分析 本章首先介绍了数据采集的工具，之后对数据的采集和分析方法做了详细地描述，并列出分析结果。

第四章 基于文本文档的 CAN-bus 协议逆向分析系统 本章主要根据分析结果设计并实现了逆向分析系统，并进行测试。

第五章 实时 CAN-bus 协议逆向分析系统 本章主要根据第四章的不足重新完善了系统，并对各个部分功能进行测试。

第六章 结论与展望 本章主要总结了这次研究的优缺点，并指出接下来系统改进的方向。

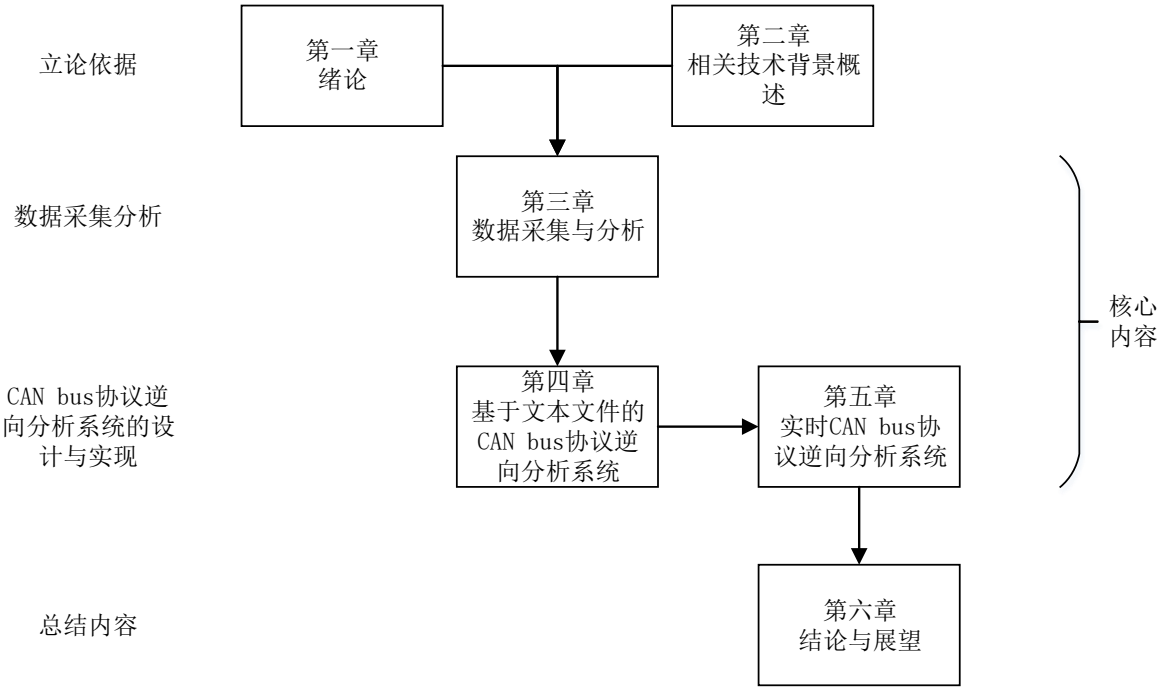

图 1-1 论文逻辑组织结构图

2 相关技术背景概述

本章对论文研究中涉及的相关理论和技术进行介绍，首先介绍了 CAN-bus 线路相关的特点、优势以及一些应用举例，之后引出应用于车辆的 J1939 协议，简单介绍了其报文格式和分类。

2.1 CAN-bus

控制器局域网络，简称 CAN-bus，是世界上应用最广泛的现场总线之一，也是欧美地区汽车智能控制系统和嵌入式工业控制局域网的标准总线，有着自动化领域的计算机局域网的美称，为实现分布式控制系统各个节点间高效的通信提供技术支持。同时，它也可以廉价地运用于汽车电气系统中，如照明、电动车窗等，可以替代所需要的硬件连接。[7]

2.1.1 CAN-bus 系统的组成

如图 2-1 所示，CAN-bus 系统主要包含了两个数据传输终端、两条数据线、收发器以及控制器几个部分。

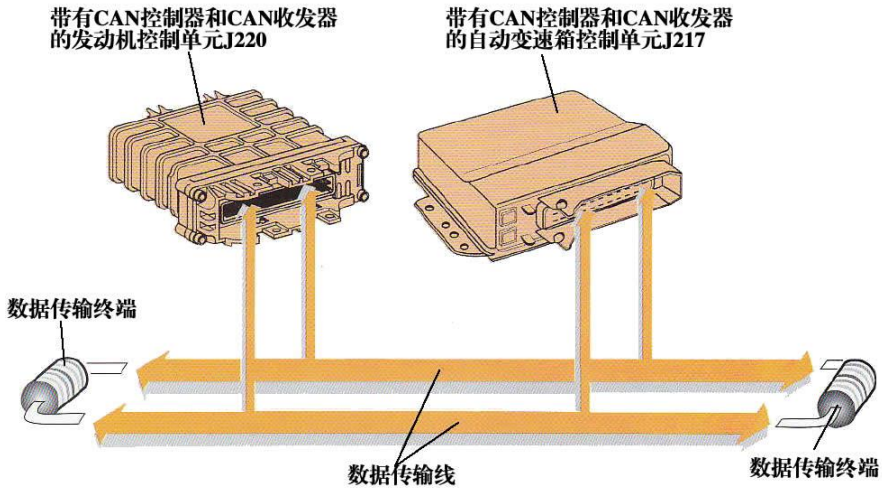

图 2-1 CAN-bus 系统的组成

CAN 控制器用于接收和处理由控制单元中的微型计算机发送来的数据，并将这些数据传送给 CAN 收发器。同样地，CAN 控制器能将来自收发器的数据处理后传输给控制单元中的微型计算机。

CAN 收发器有接收数据和发送数据两种功能。它将来自 CAN 控制器的数据转换为电信号并将其发送到数据传输线；同样,它也为 CAN 控制器接收和转换数据。

数据传输终端是一个电阻，主要功能是防止数据在线端返回，从而导致信号失真、影响数据传输。

数据传输线是双向传输数据的两条线，分别被称为高线和低线。为了防止外部电磁干扰和向外福射，CAN 总线使用双绞线，两条线上的电位相反，即若一条线电压为 5V，则另一条线为 0V，从而保证电压总和与常值相等。

2.1.2 CAN-bus 系统数据传输过程

如图 2-2 所示，CAN-bus 系统数据传输主要有以下五个过程：

（1） 准备数据——控制单元 1 准备数据，并提供给 CAN 控制器。

（2） 发送数据——CAN 收发器接收数据，并将其转换为电信号发出。

（3） 接收数据——网络内其他控制单元都成为接收器，接收数据。

（4） 检测数据——每个控制单元检测接收到的数据以判断其是否是功能所需。

（5） 认可数据——控制单元如果判断数据是被需求的，则认可并处理数据，反之则忽略。

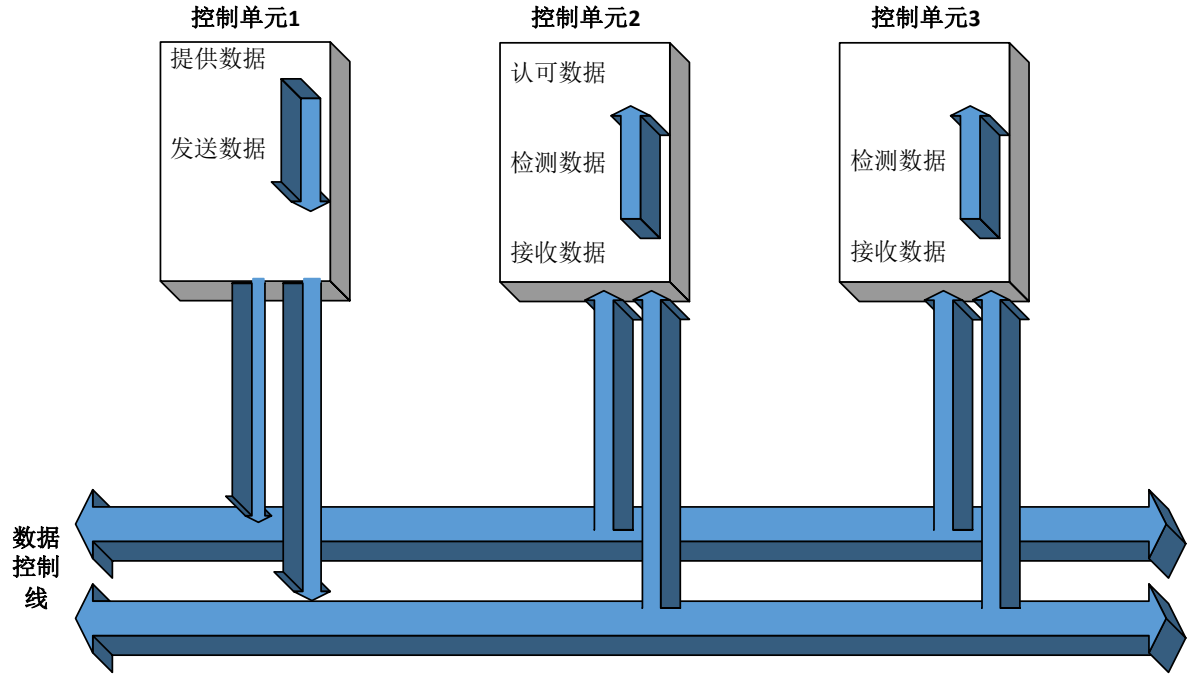

图 2-2 CAN-bus 系统数据传输的过程

2.2 J1939 报文

J1939 基于 CAN-bus 系统，主要应用在车辆的电子部件之间的数据传输，其中包括对 CAN 网络物理层、数据链路层、应用层、网络层四个层的定义以及故障诊断和网络管理，最高可达到 250Kbps 的通讯速率。在 J1939 协议中，不仅仅规定了传输类型、分组结果、构、流量检测等，还定义了报文的具体内容。[8]

2.2.1 J1939 报文格式

J1939 中定义了参数的具体格式，例如标识符、优先级、数据长度、参数范围等。而参数又分两类，状态参数和测量参数。状态参数表示多态信号的某一状态，如巡航

控制激活/关闭、发动机刹车使能/禁能、错误代码等。测量参数则表示信号的具体值，例如油缸温度、发动机转速等。

表 2-1 为 CAN2.0 的标准帧和扩展帧，J1939 协议定义的帧，三者的格式及关系。

J1939 协议报文单元主要包括优先权位、保留位、数据页位、协议数据单元、扩展单元，源地址以及数据场，如表 2-2 所示

表 2-1 CAN2.0 的标准和扩展格式及 J1939 协议所定义的格式

CAN 扩展帧格式	SOF	11 位标识符				SRR	IDE	18 位扩展标识符		
J1939 帧格式	帧起始位	优先权	R 位	数据页	PF 格式	SRR 位	扩展标识	PF	PS 格式	原地址
CAN 帧位置	1	2~4 28~26	5 25	6 24	7~12 23~18	13	14	15,16 17,16	17~24 15~8	25~32 7~0

表 2-2 J1939 协议报文单元格式

优先权位	保留位	数据页位	协议数据单元	扩展单元	源地址	数据场
3	1	1	8	8	8	0~64

2.2.2 J1939 报文分类

J1939 报文可分为三种类型，参数类，命令类和警报类。

参数类主要包括发动机的主要运行参数。 如转速，油压，油温等，这些参数有的是控制板直接就能够读取的，有些参数控制板必须发送请求命令才能读取。

命令类参数主要是是控制命令，如开机命令、关机命令，加减速命令，还有请求命令等。

报警类是当 ECU 检测到异常时发送的命令 。

而在此次系统设计中，涉及到的相关 J1939 报文以命令类报文为主。

2.3 本章小结

本章围绕论文的研究中心，对 CAN-bus 的组成以及传输方式进行了简要介绍，并且将 CAN-bus 与传统的数据方式进行比较，之后引出相关的 J1939 协议，介绍了 J1939 协议的格式和分类，为设计 CAN-bus 协议逆向分析系统提供基础。

3　数据采集与分析

本章介绍了论文研究所涉及的数据采集与分析部分。首先介绍了采集数据时使用的工具 CANalyst-II 分析仪和 CAN-bus 网络传输示教板，之后描述了数据采集的具体方法，最后对数据帧的分析方法及分析结果进行了详细的描述。

3.1.1 CANalyst-Ⅱ分析仪

CANalyst-II 分析仪是带有 USB2.0 接口和 2 路 CAN 接口的 CAN 分析仪，具有 CAN-bus 协议分析功能、工业级电压隔离、透明传输中继功等能，支持多种协议分析功能，此外还兼容周立功的 CANPro 软件。

如图 3-1 所示，CANalyst-II 分析仪有三个部分接口，分别是 USB、CAN1 和 CAN2，三端之间完全隔离。

图 3-1 CANalyst-II 分析仪外形

CANalyst-II 工作方式主要有两种，发送和接收两种。

CAN 发送：分析仪将从 PC 机的 USB 接口发来的数据解析后保存在 T-Buffer 缓冲区。同时，不断地读 T-Buffer 缓冲区，组成一个 CAN 消息帧，发送到 CAN 总线接口。

CAN 接收：分析仪将从 CAN 网络接收到的数据保存在 R-Buffer 缓冲区，当接受到请求查询接收的指令时，分析仪将缓冲区中的数据发送到 USB 接口。

3.1.2 CAN-bus 网络传输示教板

CAN-bus 网络传输示教板的原型是大众帕萨特 B7。这里主要是还原了它的

CAN-bus 系统同时配备了相关的辅助控制系统、执行器和传感器。充分展示 CAN 数据传输网络系统的结构和工作过程。

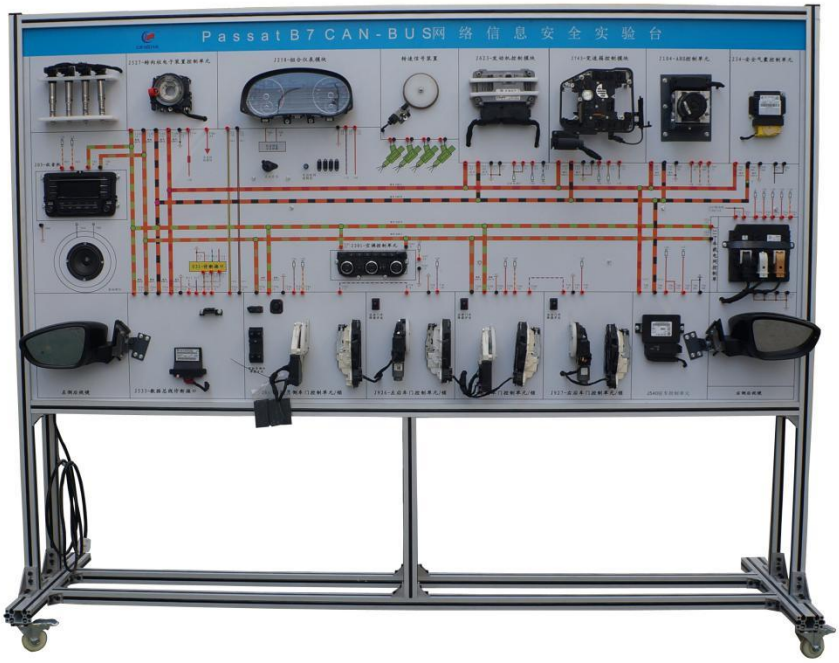

图 3-2 CAN-bus 网络传输示教板外形部件图

以下是此次数据采集分析中主要涉及得到的部件：

（1）点火开关

接通 220V 电源后，将点火开关打到 ON 位置，曲轴信号盘开始转动，示教板真实模拟发动机启动工作过程。

（2）门锁开关

按下门锁开关的左侧，车门上锁，不能从车外打开。驾驶员车门打开时，按钮将不起作用。解锁时按下开关的右侧，车门解锁。

但是示教板并没有安装车门部分，因此门锁开关的相关测试仅能通过门锁开关上的指示灯完成。

图 3-3 门锁开关、后视镜开关以及驾驶员侧升降器开关

（3）电动车窗升降器

电动车窗升降器有两类，一个是驾驶员侧升降器开关，另一类是另外三个车门侧的升降器开关。

驾驶员侧升降器开关附带有车窗锁，可以控制其他三个开关是否有效，只有当车窗未上锁时才可以通过其他三个开关调节其对应车窗升降

另外，在熄火后约 10 分钟内，只要前面两个车门没有打开，车窗升降器还可以继续工作。然而由于该示教板并没有车门部分，这项功能并不能进行测试。

（4）后视镜开关

后视镜采用电气控制，驾驶员可通过操纵电动后视镜开关，轻松调节后视镜的位置，获得理想的后视镜位置。后视镜调节旋钮调到 L 位置可调整左侧后视镜，调到 R 位置则可调整右侧后视镜；调到中间则不进行调整。

（5）空调控制单元

该示教板附带空调控制单元，可以调节空调的工作模式、风向以及设定温度等，但是并没有空调的其他部分，所以仅能够通过控制单元控制，其具体结果显示只能从指示灯上判定。

图 3-4 空调控制单元

（6）其他部件

此外，示教板还有转速控制装置、发动机控制装置以及收音机等模块。但是由于可以操纵按钮的限制，这些部件不能明显地从示教板上看出对其调整后的具体反映，因此在本次实验中没有涉及。即，本次实验主要是分析 CAN-bus 上舒适总线部分。

3.2 数据采集

将 CANalyst-II 分析仪的 CAN1 部分 H、L 两个接口与舒适总线的接口的 H、L 端相连，另一侧通过转接线接入电脑的 USB 端口。

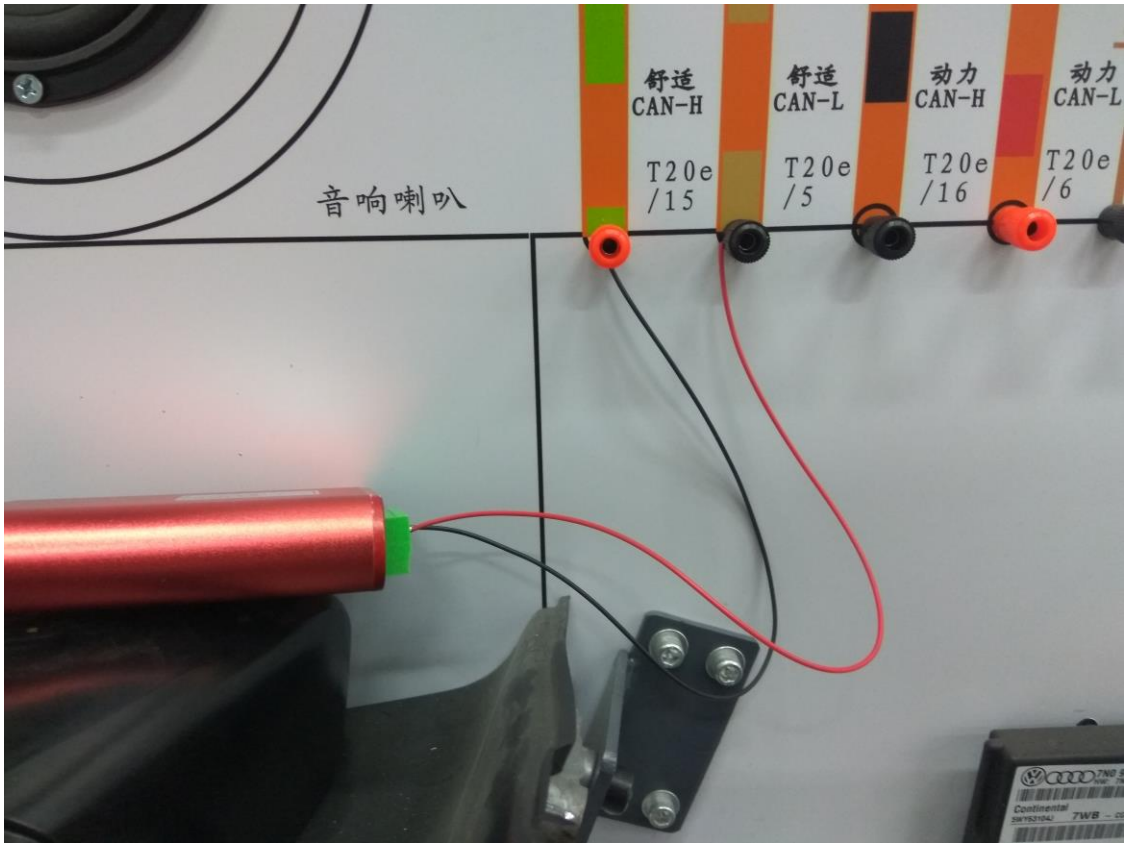

图 3-5 采集数据接线

为了采集数据的方便，使用周立功 CANpro 采集数据。

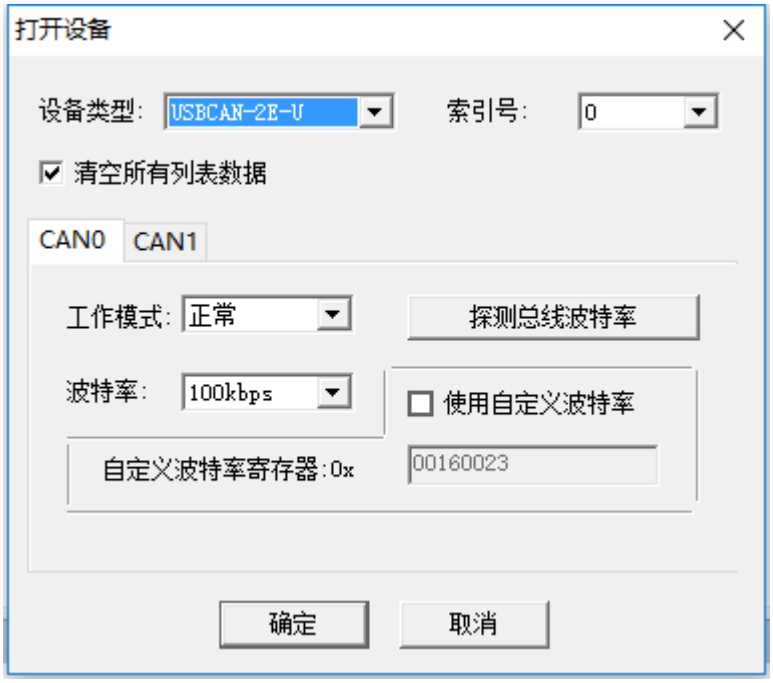

图 3-6 周立功 CANpro 打开设置

打开后启动设置如图所示，根据软件的说明书，选择设备类型为 USBCAN-2E-U，索引号为 0。而分析仪的接线用的是 CAN1 部分，对应软件 CAN0，其中工作模式正常，但波特率仍然未知，只能对常用波特率一个个尝试。当列表显示没有读取到的任何信

号时，则表明波特率不对。经过多次测试，波特率为 100kbps。此时在列表中不断读取到大量数据。

图 3-7CANpro 读取到的大量数据

序号	传输方向	时间标识	状态	名称	帧ID	格式	类型	DLC	数据
2986	接收	18:25:04.211			0x00000359	数据帧	标准帧	0x08	80 CD FF 00 90 6B C0 00
2987	接收	18:25:04.214			0x00000367	数据帧	标准帧	0x08	E0 00 00 00 7D FF 0B 00
2988	接收	18:25:04.218			0x00000527	数据帧	标准帧	0x08	10 FF FF 90 7F FF FF 19
2989	接收	18:25:04.218			0x000004BD	数据帧	标准帧	0x06	06 00 FF 00 00 00
2990	接收	18:25:04.218			0x00000369	数据帧	标准帧	0x08	FC 00 1F 00 00 00 00 00
2991	接收	18:25:04.226			0x0000040C	数据帧	标准帧	0x08	00 01 04 01 00 44 00 00
2992	接收	18:25:04.229			0x00000470	数据帧	标准帧	0x08	00 00 C6 C6 50 00 00 1F
2993	接收	18:25:04.232			0x000005DD	数据帧	标准帧	0x05	46 00 90 00 00
2994	接收	18:25:04.235			0x000004B9	数据帧	标准帧	0x06	06 00 FF 00 00 00
2995	接收	18:25:04.247			0x0000062F	数据帧	标准帧	0x04	00 00 00 00
2996	接收	18:25:04.250			0x00000381	数据帧	标准帧	0x06	80 0C 00 00 00 00
2997	接收	18:25:04.253			0x00000383	数据帧	标准帧	0x04	00 00 00 00
2998	接收	18:25:04.253			0x00000555	数据帧	标准帧	0x08	E8 00 79 00 00 00 00 00
2999	接收	18:25:04.256			0x00000561	数据帧	标准帧	0x08	F0 00 00 00 00 00 00 F0

读取到一条数据主要包含以下内容：序号、传输方向、时间标识、状态、名称、帧 ID、格式、类型、DLC 以及数据内容等十项内容。

其中序号应该是由 CANpro 软件在读取数据的同时添加的部分，其余九项为从分析仪中读取到数据帧的具体内容。

而状态、名称两项空缺，格式和类型两项内容基本统一不变，即正常情况读取的数据这四项没有太大变化。

传输方向如其名所示，确定数据的传输方向。

时间标识表明数据收发的时间。

即读取到的数据帧中，与其具体含义关联性最大的主要是帧 ID、DLC 和数据内容这三项。其中 DLC 用于修饰数据内容的长度。观察得知，对于大部分相同帧 ID 的数据帧，数据内容的长度基本是相同的，少部分例外。

根据 J1939 协议的报文格式可以确定，每个帧 ID 会对应对汽车上的一些参数，如车门锁打开/关闭，点火开关打开/关闭等。

经过多次采集，发现在静态时，采集到的数据的帧 ID 主要是以下 53 个：0x040B，0x03E1，00x5E1，0x0151，0x05B5，0x0658，0x0470，0x04B9，0x0531，0x0555，0x0561，0x040C，0x0359，0x035B，0x0651，0x03B5，0x0381，0x0383，0x058C，0x0369，0x05DD，0x02C1，0x02C3，0x04BD，0x0400，0x0397，0x0457，0x0621，0x05E8，0x05F7，0x0151，0x0658，0x05F7，0x0151，0x0402，0x0601，0x062F，0x0367，0x0527，0x0403，0x0551，0x0571，0x0575，0x0591，0x05D1，0x065F，0x05F5，0x0631，0x0635，0x06DF，0x06DE，0x040A，0x03B5。

此外，在操作示教板上可以操作的各个按钮后，发现采集到的帧 ID 多了一个 0x0181。

3.3 数据分析

在使用 CANpro 采集数据的同时，可以对数据进行滤波处理，仅在列表中显示出某一特定的帧 ID 的数据。

数据分析针对指令执行序列分析，采用控制变量法，单独筛选出相同帧 ID 的数据，调节示教板上各个按钮，每次向示教板发送单一的指令，观察采集到数据内容的变化。

以下逐一分析各数据帧。

1）帧 ID 为 0x0151 的数据帧

当采用滤波处理，筛选出帧 ID 为 0x0151 的数据帧时，发现在静态时采集到的数据如下表所示：

表 3-1 静态时帧 ID 为 0x0151 的数据

x| 00 F9 08 F1
x| 00 F9 18 E1
x| 00 F9 28 D1
x| 00 F9 38 C1
x| 00 F9 48 B1
x| 00 F9 58 A1
x| 00 F9 68 91
x| 00 F9 78 81
x| 00 F9 88 71
x| 00 F9 98 61
x| 00 F9 A8 51
x| 00 F9 B8 41
x| 00 F9 C8 31
x| 00 F9 D8 21
x| 00 F9 E8 11
x| 00 F9 F8 01

即前两个字节始终为 00 F9，而第三四字节的后一位为 8、1，而前一位之和为 F。以此不断循环。

而当依次操作示教板上的各个按钮时，采集到的数据和上面的规律一样，没有变化。即目前并不能分析出帧 ID 为 0x0151 的数据帧与示教板某个部件之间的关系。

2）帧 ID 为 0x0181 的数据帧

在之前采集到帧 ID 为 0x0181 的数据主要如下表所示：

表 3-2 帧 ID 为 0x0181 的具体数据

x| 00 00

x| 00 01
x| 00 04
x| 00 10
x| 00 40
x| 01 00
x| 04 00
x| 10 00

当筛选 0x0181 时，发现在静态时，并没有采集到任何数据。即在静态时，不会有帧 ID 为 0x0181 的数据帧产生。

当依次操作示教板上的按钮，都没有采集到数据直到操作驾驶员侧升降器开关后 1-2 秒内，会产生 10 条左右数据，其形式与上表类似，数据长度为两个字节，四个位，每个位都有可能是 0、1、2、4、8 这五个数字。

继续操作升降器开关，同时观察经过滤波后采集到的帧 ID 为 0x0181 的数据，发现这四个位分别对应着驾驶员侧升降器开关的四个按钮，依次是:右前窗，左前窗，右后窗和左后窗。而 0、1、2、4、8 这五个数字代表的是五种车窗的状态，4 表示上升，1 表示下降，8 表示持续上升，2 表示持续下降，而 0 表示不对车窗进行任何操作。

而内容为 x|00 00 的数据仅在结束操作驾驶员侧升降器开关后 1-2 秒内产生，而在静态时不会采集到这帧数据。

以上即为帧 ID 为 0x0181 的数据帧与驾驶员侧升降器开关的关系。

3）帧 ID 为 0x0291 的数据帧

当采用滤波处理，筛选出帧 ID 为 0x0291 的数据帧时，发现在静态时，其具体的数据为：x| 00 00 00 00 00 00 00 00 。

即帧 ID 为 0x0291 的数据共八个字节，静态时均为 0。

依次操作示教板上的按钮，发现当操作门锁开关时，采集到的数据会发生变化，如下表所示：

表 3-3 帧 ID 为 0x0291 的具体数据

x| 00 00 00 00 00 00 00 00
x| 00 55 00 00 00 00 00 00
x| 00 AA 00 00 00 00 00 00

即只有第二字节会发生变化，可变为 11，55,或者 AA，其他字节数据均为 00.

进一步操作门锁开关发现，55 对应锁门，AA 对应打开门锁，而 00 表示没有对门锁进行任何操作。这就是帧 ID 为 0x0291 的数据帧与门锁开关之间的关系。

以上三个数据帧是所有数据帧中的典型示例，帧 ID 为 0x0151 的数据帧代表尚未发现和示教板关系的数据帧，帧 ID 为 0x0291 的数据帧代表静态时能采集到数据且已

分析出与示教板某部件的关系的数据帧，而帧 ID 为 0x0181 的数据帧代表静态时没有采集到数据但已分析出与示教板某部件的关系数据帧。

其他数据帧的分析过程和以上三种数据帧类似，由于篇幅原因，不再列出。以下几个小部分为其他数据帧与示教板之间的关系。

4）帧 ID 为 0x0381 的数据帧

帧 ID 为 0x0381 的数据帧在静态时数据为 x| 80 00 00 00 00 00。

其与示教板之间的关系如下:

第一字节第一位：静态为 8，C 表示上锁，A 表示解锁；第二位：0 表示车门未锁，2 表示已锁；

第二字节第二位：C 表示车窗未锁，0 表示已锁；

后四个字节：为 00 00 00 00 时表示左前车窗没有变化，为 00 02 00 01 时表示左前车窗在升降变化；

而第二字节第一位：4 表示车窗上升，8 表示车窗持续上升，1 表示车窗下降，2 表示车窗持续下降，而 0 表示车窗没有变化。

5）帧 ID 为 0x03B5 的数据帧

帧 ID 为 0x03B5 的数据帧在静态时数据为 x| 80 00 00 00 00 00。

其与示教板之间的关系如下:

第一字节第二位：0 表示车门未锁，2 表示已锁；

后五字节 00 00 02 00 01，驾驶员侧操作右前车窗下降；

后五字节 00 00 01 00 00，驾驶员侧操作右前车窗上升；

后五字节 40 00 02 00 01，右前门操作右前车窗下降；

后五字节 40 00 01 00 00，右前门操作右前车窗上升。

6）帧 ID 为 0x03E1 的数据帧

帧 ID 为 0x03E1 的数据帧在静态时数据为 x| 42 00 FF 00 00 00 80 00。

其与示教板之间的关系如下:

第五字节为 02，1A，39，59，71，分别对应空调自动挡的 1、2、3、4、5 档；

第七字节为表示 84 空调未开；

第七为 80 空调已开，第五字节增加，升温到某个值，受左旋钮控制，最大 2F，最小 4E。

7）帧 ID 为 0x03E3 的数据帧

帧 ID 为 0x03E3 的数据帧在静态时数据为 x| FF FF FF 00 00 FE 00 FF。

其与示教板之间的关系如下:

第四个字节值为 00 08 10 18 分别对应空调控制单元第一行第六个按钮三个档，00

01 02 03 对应第一行第一个按钮三个档，第四字节为两个值之和

第五个字节 80 为第一行第五个按钮左灯亮，00 为未亮或右灯亮。

8）帧 ID 为 0x04B9 的数据帧

帧 ID 为 0x04B9 的数据帧在静态时数据为 x| 06 00 FF 00 00 00。

其与示教板之间的关系如下:

第二字节为表示左后门操作左后窗，为 10 时表示按住下降，20 时表示持续下降，40 是表示上升，80 时表示持续上升；

第五个字节：00 为车窗未锁，08 为车窗已锁。

9）帧 ID 为 0x04BD 的数据帧

帧 ID 为 0x04BD 的数据帧在静态时数据为 x| 06 00 FF 00 00 00。

其与示教板之间的关系如下:

第二字节为表示右后门操作右后窗，为 10 时表示按住下降，20 时表示持续下降，40 是表示上升，80 时表示持续上升；

第五个字节：00 为车窗未锁，08 为车窗已锁。

10）帧 ID 为 0x0591 的数据帧

帧 ID 为 0x0591 的数据帧在静态时数据为 x| 83 01 0F 00 40 BC 02 00。

其与示教板之间的关系如下:

第二字节，第五字节第七字节为 10 40 02 时车门未锁，为 01 48 52 时车门锁上。

11）帧 ID 为 0x0601 的数据帧

帧 ID 为 0x0601 的数据帧在静态时数据为 x| 00。

其与示教板之间的关系如下:

这个数据帧表示对后视镜开关的操作。

第一位表示选择后视镜，0 为中间不选择后视镜，1 为两侧车窗均操作，2 为仅操作右侧车窗；

第二位 0 为未动，4 为后视镜靠近车，8 为后视镜远离车。

12）帧 ID 为 0x064F 的数据帧

帧 ID 为 0x064F 的数据帧在静态时数据为 x| 03 00 30 00 00 FF FF 00。

其与示教板之间的关系如下:

第三字节第二位：0 时车门未锁，C 时车门已锁。

以上为分析后得到的各帧与示教板的之间的关系，尚未发现与示教板部件间关系的数据帧没有列出来。

3.4 本章小结

本章围绕数据采集分析，首先介绍了采集数据的工具，简单涉及到了示教板上与研究相关的部件；然后介绍了数据采集的方法以及在采集过程中可以对数据分析进行滤波的预处理；最后给出以帧 ID 为 0x0151、0x0181 和 0x0291 三个数据帧为代表的数据帧的分析过程，以及其他能分析得出其与 CAN-bus 网络传输示教板某些部件关系的数据帧。

4 基于文本文档的 CAN-bus 协议逆向分析系统

本章主要介绍了对基于文本文档的 CAN-bus 协议逆向分析系统的一些设计，其中包含有设计思路和部分关键代码以及最后对该系统进行了评估。

4.1 设计思路

由以上的数据采集和分析的结果可以得知，CAN-bus 协议的具体含义主要在数据帧的帧 ID 和数据这两部分。而设计 CAN-bus 协议逆向分析系统，可以首先将采集的数据导入一个结构体，其中主要包含数据帧的帧 ID、数据长度、数据内容以及其他内容。其中帧 ID 和数据用于对协议的分析；而数据的格式设置成数组比较合适，而每一数据帧的数据长度不一，因此使用数据长度来便于将采集到的数据导入设置的结构体中。

先前，实施数据采集的 CANpro 工具可以将采集到的数据以文本文档的格式保存到电脑上。

所以，根据以上准备，基于文本文档的 CAN-bus 协议逆向分析系统的主要工作流程如流程图所示：

首先，提前通过 CANpro 在一段时间内采集数据并将其保存为文本文档；

第一步，打开系统后，读取文本文档，将一条数据导入系统设定的结构体中；

第二步，根据结构体里的帧 ID 进行条件匹配，若匹配成功则进入第三步，否则进入第四步；

第三步，根据结构体数据部分的内容判断汽车进行了什么操作，并输出结果；

第四步，如果这是最后一条数据则结束程序，否则，返回第二步，读取下一条数据。

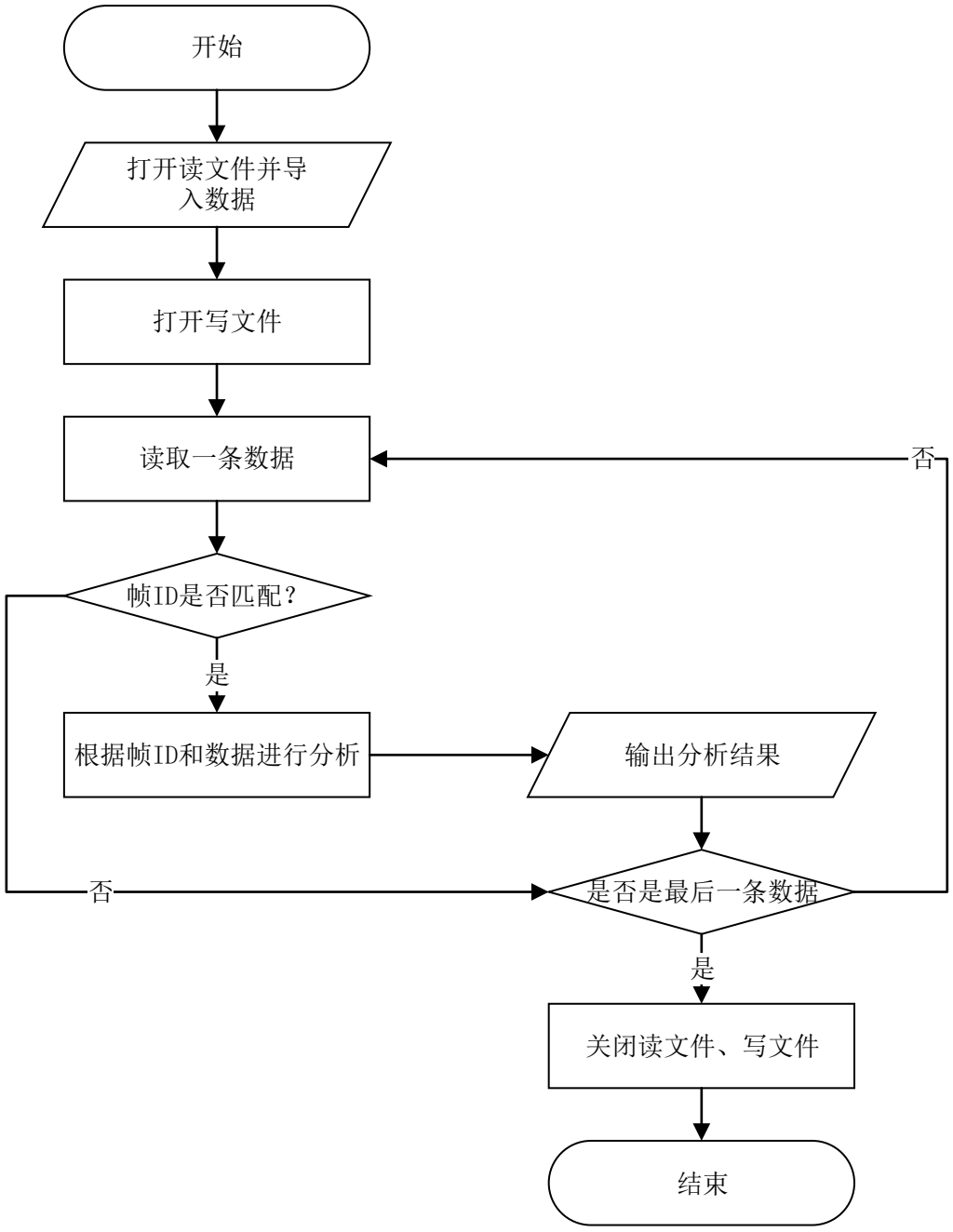

图 4-1 基于文本文档的 CAN-bus 协议逆向分析系统的流程图

4.2 系统实现

4.2.1 数据读写部分

下图为保存得到的文本文档格式的数据：

12 - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

序号	传输方向	时间标识	状态	名称	帧ID	格式	类型	DLC	数据
0	接收	16:30:42.750			0x00000397	数据帧	标准帧	0x08	03 03 00 00 00 00 00 00
1	接收	16:30:42.750			0x00000457	数据帧	标准帧	0x03	01 40 00
2	接收	16:30:42.750			0x00000470	数据帧	标准帧	0x08	00 00 C6 C6 50 00 00 1F
3	接收	16:30:42.750			0x00000531	数据帧	标准帧	0x04	03 00 70 73
4	接收	16:30:42.750			0x00000369	数据帧	标准帧	0x08	FC 00 16 00 00 00 00 00
5	接收	16:30:42.750			0x000005DD	数据帧	标准帧	0x05	46 00 90 00 00
6	接收	16:30:42.760			0x0000040C	数据帧	标准帧	0x08	00 01 04 01 00 44 00 00
7	接收	16:30:42.760			0x000004B9	数据帧	标准帧	0x06	06 00 FF 00 00 00
8	接收	16:30:42.760			0x00000601	数据帧	标准帧	0x01	00
9	接收	16:30:42.770			0x000003E1	数据帧	标准帧	0x08	43 00 FF 00 00 00 80 00
10	接收	16:30:42.770			0x00000555	数据帧	标准帧	0x08	E8 00 7B 00 00 00 00 00
11	接收	16:30:42.770			0x00000561	数据帧	标准帧	0x08	E0 00 00 00 00 00 00 E0
12	接收	16:30:42.780			0x00000381	数据帧	标准帧	0x06	82 0C 00 00 00 00
13	接收	16:30:42.780			0x00000383	数据帧	标准帧	0x04	00 00 00 00
14	接收	16:30:42.780			0x0000062F	数据帧	标准帧	0x04	00 00 00 00
15	接收	16:30:42.780			0x000002C1	数据帧	标准帧	0x06	00 00 81 00 14 00
16	接收	16:30:42.780			0x000002C3	数据帧	标准帧	0x01	07
17	接收	16:30:42.800			0x00000470	数据帧	标准帧	0x08	00 00 C6 C6 50 00 00 1F
18	接收	16:30:42.800			0x00000531	数据帧	标准帧	0x04	03 00 80 83
19	接收	16:30:42.800			0x00000621	数据帧	标准帧	0x07	04 00 80 FF 11 AA 14
20	接收	16:30:42.800			0x000003B5	数据帧	标准帧	0x06	82 00 00 00 00 00
21	接收	16:30:42.810			0x00000400	数据帧	标准帧	0x08	02 01 04 05 28 44 00 00
22	接收	16:30:42.820			0x00000658	数据帧	标准帧	0x08	70 00 00 0E 40 00 00 00
23	接收	16:30:42.830			0x0000065F	数据帧	标准帧	0x08	01 44 36 32 41 34 36 48
24	接收	16:30:42.830			0x00000151	数据帧	标准帧	0x04	00 F9 98 61
25	接收	16:30:42.830			0x0000058C	数据帧	标准帧	0x08	80 00 3F 00 FF 03 00 00
26	接收	16:30:42.830			0x000004BD	数据帧	标准帧	0x06	06 00 FF 00 00 00

图 4-2CANpro 保存的数据

其中，数据最多有八个字节，则数据的数组声明 8 位，通过 DLC 确定调用的数组位数。根据上述格式，构造的结构体如下：

```
struct OBJ    //
{
  int xuhao;                    //序号
  const char* fangxiang = "接收";          //传输方向
  char TimeStamp[13];//时间标识
  char zhuangtai = 0;           //状态
  char mingcheng = 0; ;         //名称
  long int id;                    //帧 id
  const char* geshi = "数据帧";//格式
  const char* leixing = "标准帧";//类型
  int DataLen;                  //数据长度
  int Data[8];                  //数据
};
```

而之后打开写文件和读文件，同时进行读写，部分函数如下：

```
FILE *fp;
  ofstream fout("D:\dout.txt", ios::app);//打开写文档
  if ((err = fopen_s(&fp, "D:\date.txt", "r")) != 0)//打开读文件
  {
        printf("The file was not opened\n");
```

```
6.          return -1;
7.    }
8.    while (err == 0)
9.  {
10.  /**************************************************************************/
11.  /************************导入、分析数据并输出结果************************/
12.  /**************************************************************************/
13. if (obj1.xuhao == obj[i].xuhao)//当连续读到两条序号相同的数据时，关闭读写文件结束
    循环
14.         {
15.         fclose(fp);
16.         fout.close();
17.         break;
18.         }
19.  system("CLS");//清理屏幕
20. }
```

数据输出的内容主要有三个部分，首先输出了当前读取这一帧数据的帧 ID 和数据，之后经过分析后输出其具体含义，最后输出了当前数据帧所有部件的状态。而最后主要是在界面上输出以上信息，同时会将当前帧的各项信息以及具体指令含义保存到文本文档中。

4.2.2 数据分析部分

对于数据的分析主要是通过 switch 函数与帧 ID 进行匹配，之后对能够匹配到的数据帧与之前数据分析的结果进行对比，得出这帧数据表示的指令含义。

以下为其中对帧 ID 为 0x0551 的数据帧的分析判断：

```
1.    switch (obj[i].id)
2.    {
3.  //……
4.    case 0x00000551:
5.          aa = obj[i].Data[0] | 0xF0;
6.          if (aa == 0xF0)
7.                dianhuo1 = 0;
8.          if (aa == 0xF1)
9.                dianhuo1 = 1;
10.         break;
11. //……
```

```
12.  default:
13.          break;
14.  }
```

4.3 功能测试

对采集到的一个数据集导入系统后，系统会依次对这些数据进行处理分析，以下某一时刻处理的截图和处理后输出的文件截图：

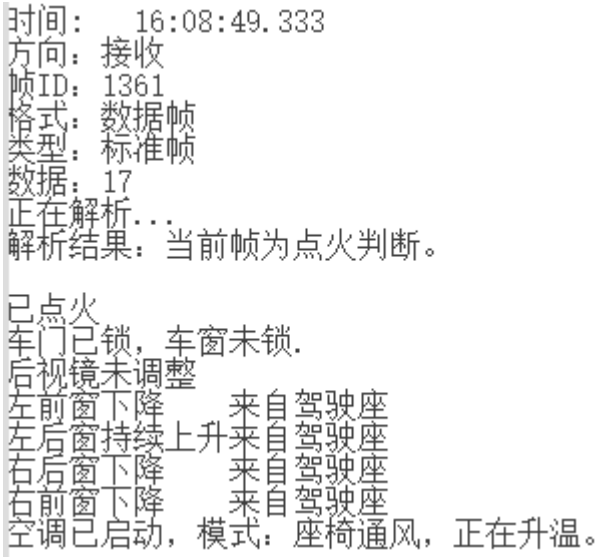

图 4-3 某一时刻对数据集进行处理

```
0       16:08:40.523接收  1136  数据帧  标准帧  0,0,198,198,80,0,0,31,   解析结果：当前帧为
1       16:08:40.523接收  1034  数据帧  标准帧  11,1,1,1,0,4,0,0,   解析结果：当前帧为
2       16:08:40.523接收  1329  数据帧  标准帧  3,0,160,163,   解析结果：当前帧为
3       16:08:40.533接收  1361  数据帧  标准帧  17,   解析结果：当前帧为点火判断。
4       16:08:40.533接收  1569  数据帧  标准帧  4,1,128,255,17,171,21,   解析结果：当前帧为
5       16:08:40.533接收  1420  数据帧  标准帧  128,0,63,0,255,3,0,0,   解析结果：当前帧为
6       16:08:40.543接收  949  数据帧  标准帧  130,0,0,0,0,0,   解析结果：当前帧为门锁及右前窗判断。
7       16:08:40.543接收  1209  数据帧  标准帧  6,0,255,0,0,0,   解析结果：当前帧为窗锁及左后窗判断。
8       16:08:40.553接收  1755  数据帧  标准帧  48,90,5,   解析结果：当前帧为
9       16:08:40.553接收  705  数据帧  标准帧  0,0,129,0,20,0,   解析结果：当前帧为
10      16:08:40.553接收  707  数据帧  标准帧  7,   解析结果：当前帧为
11      16:08:40.553接收  337  数据帧  标准帧  0,249,8,241,   解析结果：当前帧为
12      16:08:40.553接收  1624  数据帧  标准帧  112,0,0,14,64,0,0,0,   解析结果：当前帧为
13      16:08:40.563接收  897  数据帧  标准帧  130,12,0,0,0,0,   解析结果：当前帧为门窗锁及左前窗判断。
14      16:08:40.563接收  899  数据帧  标准帧  0,0,0,0,   解析结果：当前帧为
15      16:08:40.563接收  1035  数据帧  标准帧  12,1,0,0,0,0,   解析结果：当前帧为
16      16:08:40.573接收  1631  数据帧  标准帧  1,68,54,50,65,52,54,72,   解析结果：当前帧为
17      16:08:40.573接收  919  数据帧  标准帧  13,13,0,0,0,0,0,0,   解析结果：当前帧为
18      16:08:40.573接收  1213  数据帧  标准帧  6,0,255,0,0,0,   解析结果：当前帧为窗锁及右后窗判断。
19      16:08:40.573接收  1111  数据帧  标准帧  1,64,0,   解析结果：当前帧为
20      16:08:40.573接收  1136  数据帧  标准帧  0,0,198,198,80,0,0,31,   解析结果：当前帧为
21      16:08:40.573接收  857  数据帧  标准帧  128,205,255,0,144,107,192,0,   解析结果：当前帧为
22      16:08:40.583接收  859  数据帧  标准帧  8,0,0,255,35,255,1,0,   解析结果：当前帧为
23      16:08:40.583接收  1329  数据帧  标准帧  3,0,176,179,   解析结果：当前帧为
24      16:08:40.583接收  1617  数据帧  标准帧  208,3,80,143,57,89,64,0,   解析结果：当前帧为
25      16:08:40.583接收  1501  数据帧  标准帧  70,0,144,0,0,   解析结果：当前帧为
26      16:08:40.583接收  873  数据帧  标准帧  252,0,15,0,0,0,0,0,   解析结果：当前帧为
```

4-4 处理数据集后输出的文本文档

4.4 本章小结

本章主要介绍了基于文本文档的 CAN-bus 协议逆向分析系统的设计思路，之后对导入数据和分析数据的关键部分代码进行了说明，最后测试了系统的相关功能。

该系统基本能够实现对采集到的数据进行逆向的功能。由于逆向的分析方法比较简单，具体的逻辑规律由手动测试得到，该系统在准确率上没有问题。在处理效率上，任意取一个样本数为 500 的数据集，导入系统分析，用时约 29 秒，而观察这些数据，发现这些数据采集时所用的时间约 5.2 秒，相对来说处理效率还是慢了。此外该系统仅能适用于从汽车上采集后保存下来的数据集，可以用来事后对汽车进行一些分析，但是不能实时的反应汽车上一些变化。

5 实时 CAN-bus 协议逆向分析系统

本章主要介绍了对实时 CAN-bus 协议逆向分析系统的一些设计，其中首先介绍了分析仪相关的库函数，根据这些函数的调用流程确定了系统的整体框架，之后通过对交互界面的设计对系统进行了完善，然后对各部分关键代码进行了说明，最后对系统各项功能测试并做出评估。

为了改善基于文本文档的 CAN-bus 协议逆向分析系统的不足，新的系统需要直接从分析仪上读取数据。而在采集数据的时候，可以发现操作示教板某个部件时，能够获取几条相同的数据，此外未进行滤波的数据每秒钟能采集到 350 条左右。这意味着不需要对读取到的每一帧数据都进行分析。即可以通过对读取的数据随机采样再进行分析的方式提高系统的效率，同时保证分析的准确率。

5.1 相关库函数

由于 CANpro 程序不能修改，也不能接入之前编写的数据分析函数，实时 CAN-bus 协议逆向分析系统需要重新写数据接收模块，直接从分析仪读取数据。

分析仪附带二次开发函数库，其中部分函数如下：

1） VCI_CAN_OBJ

与之前基于文本文档的 CAN-bus 协议逆向分析系统中自行设置的 CAN 帧结构体不同，直接与分析仪连接需要 CAN 帧的格式严格地与分析仪一致。

```
1.  typedef struct _VCI_CAN_OBJ {
2.  UINT ID; //帧 ID。
3.  UINT TimeStamp; //设备接收到某一帧的时间标识，计时单位为 0.1ms
4.  BYTE TimeFlag; //是否使用时间标识
5.  BYTE SendType; //发送帧类型。=0 时为正常发送，=1 时为单次发送
6.  BYTE RemoteFlag; //是否是远程帧
7.  BYTE ExternFlag; //是否是扩展帧
8.  BYTE DataLen; //
9.  BYTE Data[8]; //
10. BYTE Reserved[3];//系统保留
11. }
```

2） VCI_OpenDevice

这个函数的主要作用是打开设备，一个设备只能打开一次。

3） VCI_CloseDevice

这个函数的主要作用关闭设备。

4） VCI_InitCan

这个函数的主要作用是初始化指定的 CAN 通道。有多个 CAN 通道时，需要多次调用，在该系统中不需要使用多个 CAN 通道，只需要调用一次。

在该函数中，需要设置滤波方式和波特率。

Filter 为滤波方式，设置为 0，接收所有滤波。

而 Timing0 和 Timing1 为波特率定时器，之前确定波特率为 100kbps，查表得，设置 Timing0 = 0x04，Timing1 = 0x1C。

5） VCI_StartCAN

这个函数的主要作用是启动 CAN 通道。有多个 CAN 通道时，需要多次调用，在该系统中不需要使用多个 CAN 通道，只需要调用一次。

6） VCI_ClearBuffer

这个函数的主要作用是清空 CAN 通道的缓冲区。主要用来清除接收缓冲区数据，但同时发送缓冲区的数据也会一并被清除。

7） VCI_Transmit

这个函数的主要作用是发送数据，返回值为实际发送成功的帧数。

8） VCI_Receive

这个函数的主要作用是从指定的设备 CAN 通道的接收缓冲区中读取数据，返回值为接收到数据的帧数。

5.2 设计思路

根据 5.1 中的相关库函数之间的关系以之前第四章的系统及，实时 CAN-bus 协议逆向分析系统的结构框图如图 5-1 所示：

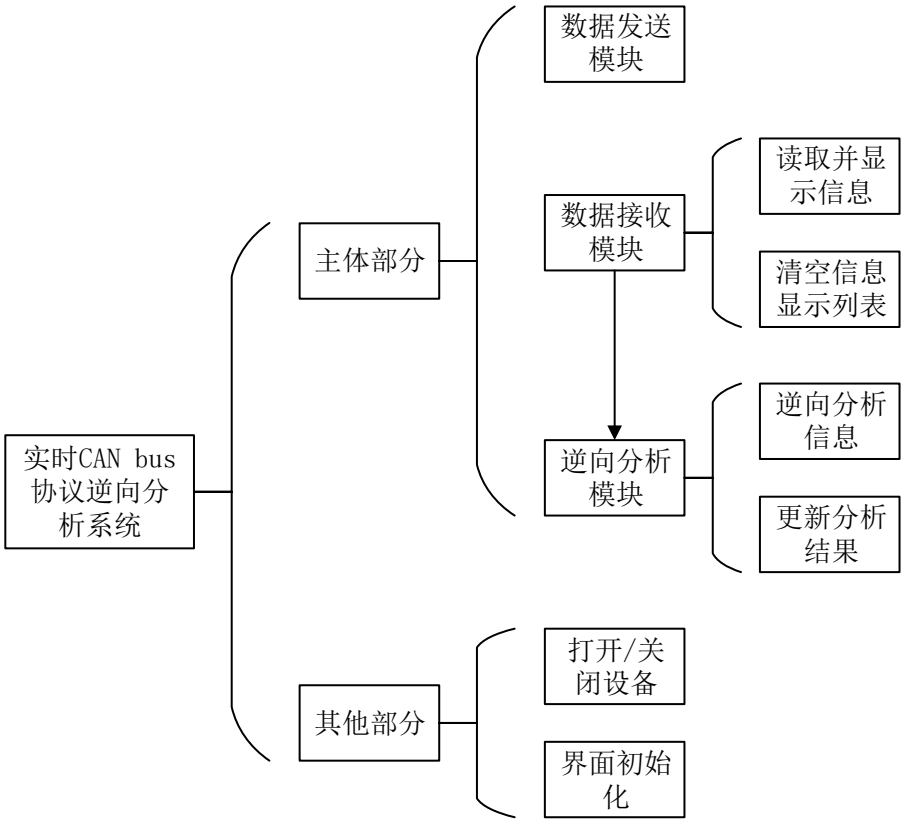

图 5-1 实时 CAN-bus 协议逆向分析系统的结构框图

其中数据发送、数据接收和逆向分析为三个独立的模块，但逆向分析模块的源数据依赖于数据接收模块。

5.3 交互界面设计

如图 5-1 所示，实时 CAN-bus 协议逆向分析系统主要有数据发送、数据接收和逆向分析三个部分，此外还需有打开设备和关闭设备两个按钮按钮。

为了读取到的数据能够同时显示当前读取的数据和之前读取的数据，数据接收模块的显示部分可以用 List Control 通过编写一个列表函数实现。此部分需要的其他控件主要有一个清空信息显示列表的按钮和打开或关闭数据接收功能的 check-box（此处不使用按钮是由于每次读取数据都需要调用函数，设为 check-box 便于判断）此外对于数据帧不同形式应该设置一个选项确定显示的帧 ID 形式

对于数据发送模块，根据数据帧的格式，数据发送部分的界面需要有类型、格式、CAN 线路索引、设备类型、帧 ID 以及数据几个部分，以及一个发送数据的按钮。而采集数据时每次对示教板的按钮进行操作会得到几条相同的相关数据，可以推测得知发送数据时，若只发送一帧数据，仅能短暂地改变示教板上某部件的状态。以车窗为例，正常情况完成车窗的升降需要按住开关达到一定时长，这不是发送一条数据可以实现的。为了使发送数据后汽车的变化更加明显，此部分界面应该有发送次数和间隔时间两个部分，以保证该系统能够连续发送相同的数据。在这些部分中，数据类型、

格式、CAN 线路索引、设备类型这四个部分需要输入的内容都比较单一，可设置为 combo-box，仅进行选择即可。而剩下的帧 ID、数据、次数、间隔四个部分由于可以输入的内容选择范围太广，使用 Edit box 较为合适。

而逆向分析模块这一部分，主要有点火开关、车门锁、车窗锁、空调、后视镜以及各个车窗几个部分，都只需要具有输出功能即可，且输出为文字，则使用 Edit box 控件禁用可改写功能比较合适。由于之后代码实现的原因，此处需要添加一个更新状态的按钮。

实时 CAN-bus 协议逆向分析系统的交互界面如下：

5-2 实时 CAN-bus 协议逆向分析系统界面

5.4 系统实现

5.4.1 数据发送部分

在交互界面中，与数据发送部分相关的按钮只有一个发送数据，其关联函数 OnButtonSend()。

而为了保证该系统能够连续发送相同的数据，发送数据部分设置了发送次数和时间间隔。因此，发送信息部分的代码在读取发送信息后，需要根据读取的发送次数多次调用发送函数 VCI_Transmit。部分代码如下：

```
1.  void CDemoCANDlg::OnButtonSend()
```

```
2.  {
3.    VCI_CAN_OBJ sendbuf[1];
4.    UpdateData(TRUE);//将数据从界面更新到变量
5.    /*************************************************************************/
6.    /************************从界面获取发送信息*****************************/
7.    /*************************************************************************/
8.    int flag;
9.    while (cishu2 > 0)
10.   {
11.         flag = VCI_Transmit(m_DevType, m_DevIndex, m_nCanIndex, sendbuf, 1);// 调用
      动态链接库发送函数
12. /*************************************************************************/
13. /*****************************发送信息列表显示*****************************/
14. /*************************************************************************/
15.         cishu2--;
16.         Sleep(jiange2);
17.   }
18. }
```

5.4.2 数据接收部分

在界面设计里，与数据接收部分相关的控件有两个，一个是清屏按钮，另一个是接收数据的选择框。

清屏按钮关联的函数是一个清空信息显示列表的函数 m_list.DeleteAllItems()。

而当接收数据的选择框被勾上是表示开始接收数据，未被选择的时候表示关闭停止接收数据，因此，在 OnCheckCanrxEn()中需要重复调用 UpdateData(TRUE)函数重复查询开启/停止接收数据的状态是否改变，相关代码如下：

```
1.  void CDemoCANDlg::OnCheckCanrxEn()
2.  {
3.    UpdateData(TRUE);
4.    if (m_bCanRxEn)
5.    {
6.          StopFlag = 0;
7.          AfxBeginThread(ReceiveThread, 0); //开启接收线程
8.    }
9.    else
10.         StopFlag = 1;
```

```
11. }
```

其中 m_bCanRxEn 为选择框对应的变量。而 ReceiveThread 包含接收数据（通过调用接收函数 VCI_Receive 实现）以及将接收到的数据输出到信息显示列表显示两项功能。在实现将信息显示到列表的同时，会将一个缓冲区大小（200 条）的数据赋到结构体数组 pCanObj[200]，供逆向分析部分使用，当时下一次读取缓冲区数据时，新的数据会覆盖掉旧的数据。

5.4.3 逆向分析部分

在逆向分析部分中，分析函数 analyze(VCI_CAN_OBJ a)的分析依据也是来源于第三章中对数据分析的结果，因此其设计思路与第四章基于文本文档的 CAN-bus 协议逆向分析系统的分析部分思路相同，仅需要修改部分变量名。这部分函数调用在更新信息信息列表的代码后面。而其中的参数 pCanObj[i]为全局变量，在数据接收并输出到列表的同时会更新。

analyze(VCI_CAN_OBJ a)函数分析得到的结果会首先保存在与界面中控件绑定的变量中，但由于接收函数中一直调用 UpdateData(TRUE)，这使得控件中的内容会首先覆盖变量中的信息。

因此，需要设计一个更新状态的按钮，点击时会发生中断，此时函数 OnCheckCanrxEn()暂停执行，而是调用函数 void OnButtonshow()，进行逆向分析，并将分析结果更新到界面中，从而达到更新汽车状态的目的。整个过程完成后，系统继续接收数据。

其中在选择数据进行逆向分析的时，不需要对所有数据逐帧进行分析，只需要随机采样分析其中的部分数据，以此提高系统的分析效率。这里采用每 10 条数据中抽取一条数据进行分析。相关代码如下：

```
1.  void CDemoCANDlg::OnButtonshow()
2.  {
3.    StopFlag = 0;
4.    for (int i = 0; i < 200; i++)
5.    {
6.          if (i % 10 == 0)
7.          {
8.                analyze(pCanObj[i]);//开始分析数据
9.                UpdateData(FALSE);//更新列表
10.         }
11.   }
12.   StopFlag = 0;
13. }
```

5.4.4 其他部分

除了以上三个部分外，该系统的代码还应该包括界面初始化、打开设备、关闭设备。

界面初始化主要是对交互界面中显示逆向分析结果的相关控件赋上初始状态，以及使信息显示列表第一行显示各列的含义。这部分只需要修改相关变量，最后调用UpdateData(FALSE)函数将变量更新到界面中。

而打开设备部分除了要调用函数 VCI_OpenDevice 打开设备外，还需要调用 VCI_InitCan 函数，设置滤波方式、波特率定时器、工作方式，从而实现 CAN 通道初始化。

至于关闭设备部分只需要调用函数 VCI_CloseDevice 关闭设备即可。

5.5 功能测试

5.5.1 数据接收模块

这部分主要测试的内容是开启或关闭数据接收，观察列表是否开始或停止显示读取的数据。

数据接收模块测试结果如图所示：

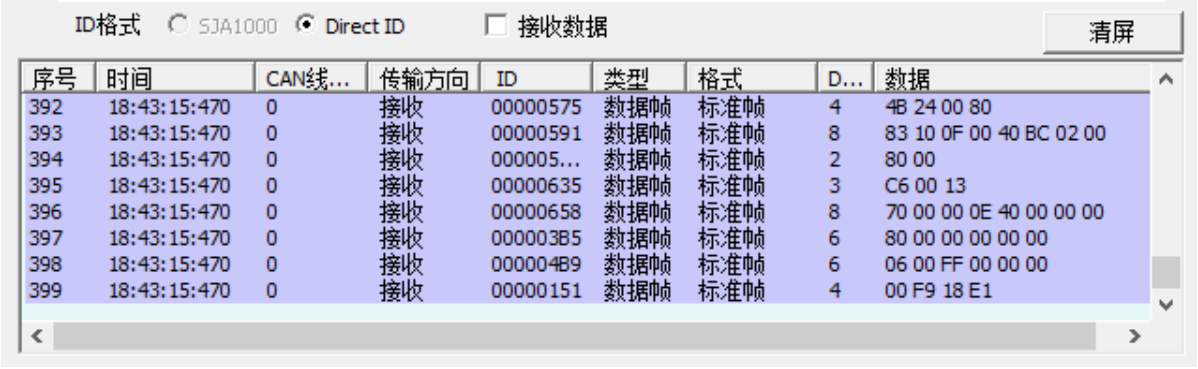

图 5-3 数据接收模块测试结果

该系统能够成功读取数据，并且在读取数据过程中选择关闭数据接收，列表中仍会显示继续部分数据，显示结束后，数据量为 200 的倍数。其原因为读取数据方式为一次性读取缓冲区中所有数据，而缓冲区大小为 200。

5.5.2 数据发送模块

这部分主要测试的内容是在开启和关闭数据接收两种情况下，分别能否通过分析仪，向示教板发送数据，观察列表输出结果和示教板是否有反应。

数据发送模块测试结果如图所示：

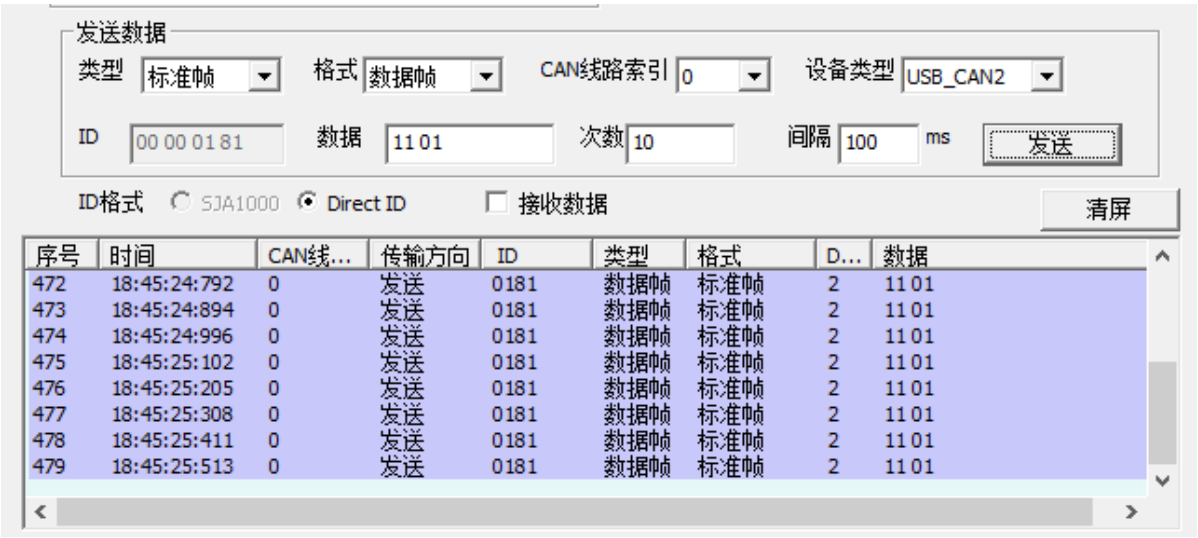

图 5-4 数据发送模块测试结果

该系统能够成功发送数据，发送的数据内容会在信息显示列表中显示出来，但示教板上是否出现变化与发送的数据内容有关系。

5.5.3 逆向分析模块

这部分主要测试的内容是在开启数据接收情况下，改变前门车窗的状态，同时点击更新状态按钮观察界面显示部分发生的变化。

逆向分析模块测试结果如图所示：

该系统在开启数据接收的情况下，点击更新状态按钮，列表更新读取的数据会发生短暂地停滞，此时显示部分会更新示教板各部件的状态。

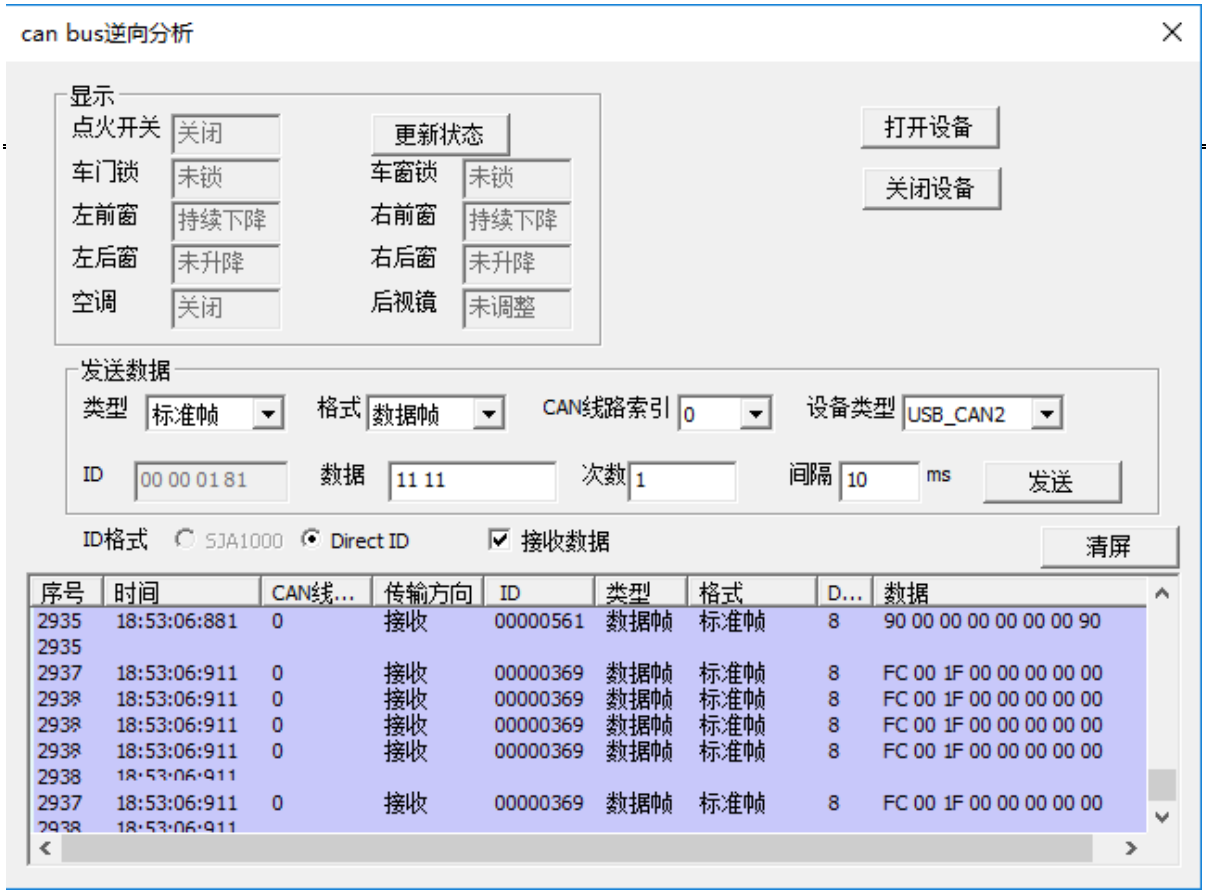

图 5-5 逆向分析模块测试结果

5.6 本章小结

本章首先列出了 CANalyst-II 分析仪的相关接口函数，并给出了整体的调用流程，使得对实时 CAN-bus 协议逆向分析系统有一个全局的认识。之后通过交互界面的设计，介绍了系统的三个模块，包括数据接收模块、数据发送模块和逆向分析模块。然后对系统的重要部分代码进行了简单介绍。实时 CAN-bus 协议逆向分析系统基本形成。最后，测试了三个模块的各项功能，并对结果进行展示和详细说明。

6 结论与展望

本文主要是以 CAN-bus 系统在车辆上的广泛应用为背景，基于协议逆向分析技术提出了实时捕获 CAN-bus 总线上的数据、对其进行分析并在 PC 端直观展示出来的需求。主要包括以下内容:

（1） 系统调研。

通过阅读相关文献，对协议逆向技术、CAN-bus 总线系统以及 J1939 报文做了深入的了解并进行总结概括。

（2） 数据采集与分析。

由于能力的限制，在数据分析的方向选择上并没有选择报文序列分析或者指令执行序列分析中比较自动化的方法，同时因为数据采集的同时可以对采集到的数据进行预处理，因此采用了手动对指令执行序列进行分析的方法。

（3） 系统设计与实现

本文首先根据之前的分析结果，设计了基于文本文档的 CAN-bus 协议逆向分析系统，但在结果测试时发现其处理输出效率低下且不能实时对数据进行处理。因此，以该系统作为核心的分析部分，重新设计了实时 CAN-bus 协议逆向分析系统。新的系统继承了前一个系统较高的准确率，同时具有实时对数据进行分析的功能，此外还增加了数据发送的功能。

但是该系统还有很大的发展空间，在实时性方面，由于代码的缺陷，目前不能做到长期处于分析状态自动进行分析，而需要手动点击按钮才会对读取到的数据进行逆向分析。在逆向分析的数据方面，仅对 CAN-bus 的舒适总线进行了分析，而对动力总线和控制总线中还有大量的数据没有进行分析。而在分析的方法上也较为复杂。

在未来的工作中，我们会根据这些问题去探索新的解决方案去优化 CAN-bus 协议逆向分析系统。

致　谢

光阴荏苒，四年的大学生活转瞬即逝，烈日炎炎下怀着紧张期待的心情踏进这所学校仿佛不过是昨天。而如今，又要在这个季节离开这里。我要感谢所有从毕业设计开题到论文的顺利完成这期间给予我指导和帮助的老师以及学长学姐们。

首先，我要诚挚感谢我的论文指导老师马小博老师，在撰写这篇论文的时候遇到很多问题与疑惑，马老师不仅提供了很多理论和技术上的指导，同时还认真负责的帮助我反复修改论文，审阅格式，规范了论文的排版，更清楚的表达了每个章节要说明的内容。而且介绍了很多优秀的学长学姐帮助我，百忙之中依旧坚持和我一起探讨问题解决问题，使得我能够完成自己的毕业设计论文。

其次感谢我的同学魏仁柱，在毕设项目一开始没有头绪时，给我指点方向，给予了我很大的帮助，以及完成毕设的过程中与我分享经验，使我少走了很多弯路。

再者，要感谢马老师实验室的同学们，他们在我本科毕设阶段给予了我很大的帮助，在毕设项目遇到困难苦于没有头绪时，他能帮助我共同找出问题关键点所在，并且指导我接触了我很多网络安全领域方面的知识。

然后要感谢的是我亲爱的室友们，我们一起生活了四年，一路走来虽然期间有小吵小闹，但是更多的是理解、包容和相亲相爱，在此过程中我感到温暖和幸福。这些友情岁月将成为我生命中不可缺少的珍贵礼物。

最后要感谢我的母校西安交通大学和所有任课的老师们，在这里，我收获了知识和友谊，也练就了乐观向上的心态和一颗感恩的心。

谨以此文对所有人献上祝福，并真诚的说一声“谢谢”！祝福母校年年桃李，岁岁芬芳！

参考文献

[1] 王荣．基于 CAN 总线的智能车辆数据采集与处理[D]．重庆：重庆交通大学，2016．

[2] 曾令元．面向比特流数据的无人机测控协议逆向[D]．成都：西南交通大学，2017．

[3] W Cui, J Kannan, H Wang. Discover: Automatic Protocol Reverse Engineering from Network Traces[C]. In 16th Usenix Security Symposium, 2007.199-212

[4] 陈佳莹．无人平台测控协议信息逆向分析[D]．成都：西南交通大学，2016．

[5] J. caballero, P. Poosankam, C. Kreibich. Bidirectional protocol reverse engineering[OL]. 2009-5-5. http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-57.html

[6] 钟晓欢．基于文本类型的应用层协议逆向解析技术的研究[D]．北京：北京邮电大学，2014．

[7] 姜玉珺．电喷柴油机应用中的 CANBUS 通信接口设计[D]．广东：华南理工大学，2012．

[8] 高云华．SAE J1939协议在汽车电器通信系统中的应用[J]．河海大学常州分校学报，2005,9

[9] 宋铮，王永剑，金波等．二进制程序动态污点分析技术研究综述[J]． 信息网络安全，2016(3):77-83.

[10] 朱宇．网络协议逆向解析与缺陷测试的关键技术研究[D]．北京：北京邮电大学，2013．

[11] J Peng; ZH Zhang; H He. A Method for Detecting Abnormality of CAN Bus inVehicle[J]. Instrumentation, 2017-06-15

[12] N Borisov, D Brumley, HJ Wang. A generic application-level protocol analyzer and ites language[C]. In Proceedings of the Network and Distributed System Security Symposium. 2008.125-130

[13] W Li, M Ai, B Jin. A Network Protocol Reverse Engineerning Method Based on Dynamic Taint Propagetion Similarity[C]//International Conference on Intelligent Computing. Springer International Publishing, 2016:580-592

[14] W Cui, M Peinado, K Chen. Automatic Reverse Engineering of Input Format: US,US8935677[P]. 2015

[15] WB Wu, TS Hong, ZL Zhang. Experimental System of Vehicle CAN bus Based on J1939 protocol[J]. Instrumentation, 2017-6-15

[16] 刘易. 船用柴油机监控系统 CAN 总线协议设计与研究[D] ．哈尔滨工程大学，2010.

[17] 潘璠，吴礼发，杜有翔等．协议逆向工程研究进展[J]．计算机应用研究，2011,28(8):2801-2806．

[18] 何亮．基于 LabVIEW 的电动汽车整车下线检测系统的研究与开发[D] ．太原：中北大学，2014．

[19] 黄强．基于 SAE J1939车载远程诊断系统设计与实现[J] ．工业控制计算机，2015,28(3):130-131

[20] 邵明朝，徐雪萍，颜传武．现代汽车 CAN 总线系统故障诊断方法[J] ．科技资讯，2017,15(15):37-38

附录 A 外文译文

关于自动化车辆的潜在网络攻击

Jonathan Petit and Steven E. Shladover

摘要：

自从上世纪 80 年代中期智能交通系统开始研究以来，机车自动化已经是其一个重要应用了。大多数时间，它通常被认为只是一个概念，距离有合适部署的准备还很遥远。然而，近年来“自动驾驶”车汽车和汽车制造商对于他们在 2020 前的部署宣告表明了，机车自动化已经开始渐渐实现了。智能交通系统产业已经开始将注意力集中在“联网汽车”（美国）或者“合作智能交通系统”（欧洲）的概念上了。这些概念是基于车辆间(V2V)或者车辆与基础设施(V2I/I2V)之间的数据通信来提供的。实现其应用程序所需的信息。单独的自动车辆和合作社的线程还没有被彻底地编织在一起，但这将是一个必要的步骤，因为在不久的将来数据的合作交换将会提供重要的投入以提高性能我们分析了自动车辆和合作自动化车辆的威胁。这个分析表明，需要比许多人预期的多得多的冗余。在汽车自动化系统发展的早期阶段，我们也提高了对这些威胁的讨论。统发展的早期阶段这些威胁的讨论。

关键词：自动化车辆，自动车辆，合作自动化车辆，网络攻击，安全。

一、引言

自 20 世纪 80 年代中期开始研究以来，汽车自动化一直是智能交通系统领域的基本应用之一。在大多数时候，它被普遍认为是一个未来的概念，并没有准备好部署。各种各样的研究项目已经在环境感知和车辆控制方面取得了有利的技术，并产生了实验性的实现，以显示自动化技术如何应用于道路车辆。这些导致了欧洲、北美和日本的大规模示威活动[1]-[6]，引起了大众媒体和贸易媒体的断断续续的关注。在公众是视野外，学术研究也在进行中[7]-[9]。

由于美国国防高级研究计划局发起的“大挑战”和“城市挑战”，公众对自动化车辆的认识有所增加。这些导致了谷歌在开发一款“自动驾驶”汽车方面的最新工作，该汽车吸引了前所未有的媒体兴趣。媒体的兴趣引发了许多关于自动驾驶对许多社会问题的影响的猜测(道路安全、隐私、交通、能源和环境影响、土地使用、汽车工业的经济和网络安全)。大多数这种猜测都是不明智的，部分原因是自动化车辆系统的操作概念还没有很好地定义。然而，公众的兴趣激发了对汽车原始设备制造商和供应商行业的新兴趣，以及政府机构开始赞助关于自动汽车概念的新研究。

近年来，智能交通系统产业已经把注意力集中在“联网汽车”(美国)或“合作”(欧洲)的概念上。这些概念是基于车辆之间的数据通信(V2V)或者车辆与基础设施(V2I/I2V)之间的数据通信，以提供实现其应用程序所需的信息。汽车工业对这些合作系统中涉及的隐私问题和网络攻击的风险表示关注，特别是对涉及碰撞警告和避碰的安全关键

应用程序。因此，目前正在进行的关于合作系统的研究包括重大的努力，以确定网络威胁和确定需要应用的战略以保护它们。

自动车辆和合作智能运输系统的单独的线程尚未被彻底地结合在一起，但这将是不久的将来的一个必要步骤，因为数据的合作交换将提供重要的投入，以提高自动化系统的性能和安全性。这意味着，开始考虑合作的自动化车辆系统对网络安全的影响，至少是重要的。

然而，对于需要关注的非合作(自治)自动化系统，也存在潜在的网络威胁。这些可能比对非合作的自动智能交通系统的威胁更具有破坏性，因为如果它完全脱离了动态驱动任务，那么驱动程序可能无法提供独立的未损坏信息或击败故障系统。

据我们所知，这是针对自动化车辆的特殊需要和漏洞的潜在网络攻击的首次调查。在汽车自动化系统发展的早期阶段，对这些威胁进行更广泛的思考和讨论是很重要的，这样更多的研究人员就可以从不同的角度来解决这个问题。对车辆自动化系统的全面保护将需要广泛的研究人员参与，他们可以预见到最广泛的威胁;因此，我们并没有声称在这一主题的最初处理中已经确定了它们。

研究意义:在本文中，我们将讨论以下问题。

a)自律自动车辆如何受到攻击?

b)合作自动化车辆如何受到攻击?

c)自律和合作的自动化车辆的安全性和隐私机制有什么不同?

组织:本文的其余部分组织如下。第二节概述了现代汽车系统安全威胁的相关工作。然后，第三节定义了与自动化有关的相关术语，并阐述了本文所考虑的假设。第四节定义攻击者模型。在第五节中，我们描述了用于对攻击表面进行分类的方法。在将自动车辆与合作的自动化车辆进行讨论之前，我们在第六节中提出了交叉挑战。然后，第七节提出并讨论了自动自动车辆的安全性和隐私威胁。同样，第 8 节讨论的是合作的自动化车辆。最后，第九节总结本文。

二、相关工作

现代汽车系统的安全分析是一个很好的研究课题。更具体地说，车载网络的安全性已经引起了关注，因为到目前为止，车辆还没有“联网”。图 1 展示了典型的车载网络结构，其中 Wolf 等[10]研究了汽车总线系统的危害(LIN, CAN, MOST, FlexRay, and Bluetooth)。他们还描述了一些对具有代表性的汽车总线系统的协议层的一些可行的攻击，假设攻击者对相应的车辆网络有物理或逻辑的访问。Hoppe 等[11]演示了实用的控制器是网络(CAN)总线攻击，攻击者可以操纵电动窗升降机、警示灯和安全气囊控制系统。

Koscher 等[13]证明了一个能够潜入几乎任何电子控制单元(ECU)的攻击者可以利用这种能力完全绕过一系列安全关键系统。他们展示了在广泛的汽车功能上施加敌意控

制的能力，完全忽略了驾驶员的输入，包括刹车失灵，有选择地制动单个轮子的需求，并停止引擎。然而，他们的攻击在自动化程度上有一定的限制，不能控制转向或加速。

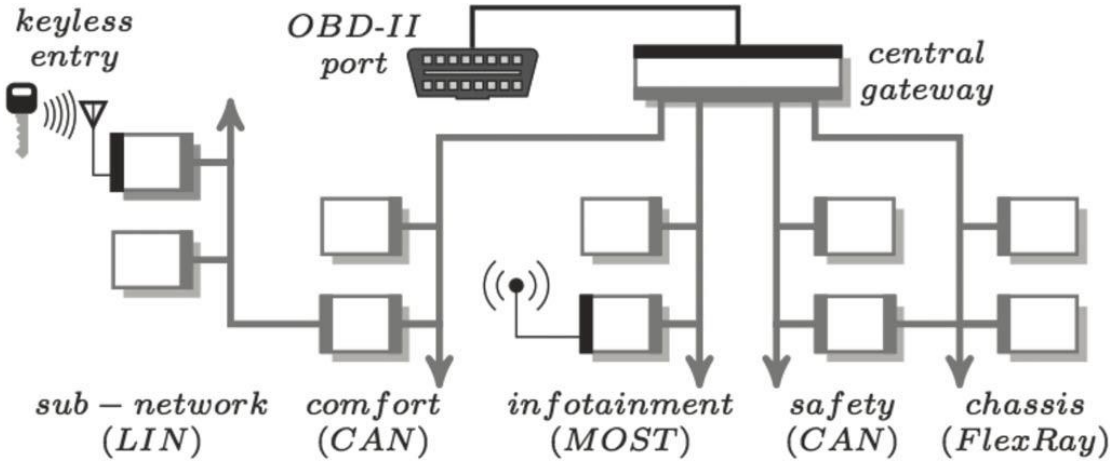

图 1 所示：现代汽车典型的车载网络结构示意图[12]。

Checkoway 等[14]分析了现代汽车的外部攻击表面。他们发现远程开发是可行的，通过广泛的攻击表面(包括机械工具、CD 播放机、蓝牙和蜂窝式无线电)，此外，无线通信通道允许远程车辆控制、位置跟踪、机舱内的音频过滤和盗窃。

我们通过调查自动化车辆系统潜在的网络攻击来区分上述工作。因此，对车载网络的攻击仍然存在，但成功攻击的后果可能更为关键。

三、定义和假设

1. 定义

自动化:使用电子或机械设备来代替人力取代人工驾驶的道路交通工具的方法。

自律自动化:完全基于车辆上传感器获取的信息的车辆自动化，不与其他实体(其他车辆或道路基础设施)进行积极的通信或合作。在本文的剩余部分中，我们用“自律自动化车辆”来表示一个具有自律自动化系统的车辆。

合作自动化:车辆自动化，包含从道路基础设施或其他车辆传达的信息，也可能涉及与其他车辆的机动谈判。在本文的剩余部分中，我们用“合作自动化车辆”来表示一个具有协同自动化系统的车辆。

动态驾驶任务[15]:在道路交通中操作机动车辆所需的所有实时功能，不包括目的地和路点的选择(即航行或路线规划)，包括但不限于:

- 对象和事件检测、识别和分类;
- 对象和事件响应;
- 实时任务规划;
- 转向、转向、车道保持、车道变化;
- 加速和减速;
- 增强醒目性(灯光、信号、手势等)。

最小风险条件[15]:一种低风险的机动车操作条件，自动驾驶系统自动地利用系统故障或驾驶员的故障来响应要求接管动态驾驶任务的请求。一个最小的风险条件可能需要自动地将车辆停在一站，最好是在活跃的交通车道之外(假定可用性)。

条件自动化[15]:利用自动驾驶系统对动态驾驶任务的所有方面进行兼职和驱动模式依赖的性能，并期望当自动驾驶系统达到其驱动模式依赖能力的极限时，驾驶员将会控制动态驾驶任务。

高自动化[15]:一种自动驾驶系统的兼职、驾驶模式依赖或地理上的限制性能的动态驾驶任务的所有方面。包括当它达到其驱动模式依赖能力的极限时，如果驾驶员在提示时不能恢复动态驾驶任务，能够自动把机动车到最小风险条件。

完全自动化[15]:自动驾驶系统在所有道路和环境条件下，由人类司机管理的所有方面的自动驾驶系统的无条件的全职工作。

2. 假设

本文关注的焦点是提供高水平自动化的动态驱动任务的系统，驱动程序不再被要求监视外部威胁的驱动环境。这意味着驾驶员的注意力很可能集中在其他的物体上，而车辆正在被驱动，因此在驾驶员能够重新投入使用任何可能需要的纠正措施之前，很可能会有一些重要的时间(至少是几秒钟)。因此，司机不可能一直被认为是确保安全的最终保障，这与 ISO 26262 功能安全标准的假设形成了对比，后者认为司机确实是安全的最终保护人。还假设司机不需要有任何特殊的培训或许可来操作自动车辆，所以司机的行为应该被认为是典型的当前司机。

在 SAE J3016 关于驱动自动化的定义中[15]，这意味着我们的重点是自动化的三个最高级别:条件自动化、高自动化和完全自动化，就像在 III-A 节中提到的那样。

对于在以往的网络攻击危害研究中考虑过的智能交通系统，车辆的驾驶员总是被假定为完全参与驾驶任务，并注意驾驶环境中的危险。然而，随着我们在这里考虑的自动化程度的规模，这显然不再是必要的情况。在有条件的自动化系统中，司机有望在一个不利事件发生的几秒钟内恢复对车辆的控制，但在这几秒钟内会发生很多事情(当行驶距离达到 100 米时)。在高自动化和完全自动化系统中，车辆自动化系统被要求将车辆带到一个安全的(“最小风险”)状态，即使司机不采取行动，给系统设计者增加了一个更大的负担，以管理网络攻击的任何后果而不影响安全。

网络攻击可能会损害自动车辆用于确定其位置并规划其轨迹的一些信息来源，而不影响其他来源。在这些条件下，车辆的数据融合软件可以通过结合所有来源的数据来确定车辆的真实状态和周围环境。数据融合软件成功识别和补偿攻击的能力取决于其他(未损坏的)信息的数量和质量。在这个基础层面上，从传感器获取的数据与从其他车辆或基础设施传输的数据之间没有根本的区别，但是来自合作系统的通信数据可以代表额外的数据源来增加车载传感器数据。事实上，合作的相邻车辆和基础设施元件(传感器)可以通过一种不受攻击的车辆来证实或驳倒观测，提供独立的方法来核实与攻击

有关的潜在可疑信息。越广泛的数据来源范围，可以利用数据融合确定车辆的真实状态和它的邻居从未被破坏的来源的机会就越多。

对自动化和协同自动化系统的威胁进行了单独分析。自治系统不包括通信来支持其控制功能;因此,基于外部来源的通信的攻击并不直接应用于它们(尽管它们可能会通过它们的信息娱乐系统(比如其他车辆)受到攻击)。除了通过它们的通信渠道进行的另一组攻击之外，合作系统还可以受到类似于自治系统的同样的影响。交流和合作的增加使他们受到更广泛的攻击，但与此同时,通信和合作数据提供额外的信息来源,可用于识别攻击和收购独立信息用来弥补攻击。

四、攻击者模型

在这里，我们定义了可能存在于自动化车辆系统中的攻击者的类型。我们采用了类似的分类[16], [17]。

内部与外部:内部攻击者是网络中的一个经过身份验证的成员，可以与其他成员进行通信；外部攻击者被网络成员视为入侵者，因此在攻击的多样性中受到限制，不过，它可以窃听通讯。

恶意与理性:恶意攻击者不寻求从攻击中获得个人利益，目的是伤害成员或网络的功能，因此，它可以使用任何方法来忽略相应的成本和后果；相反，理性的攻击者寻求个人利益，因此，在攻击手段和攻击目标方面更容易预测。

主动与被动:主动攻击者可以生成包或信号来执行攻击，而被动攻击者只监听通信通道(即:无线或车载有线网络)。

局部与扩展:攻击者可以在范围内受限，即使它控制多个实体(车辆或基站)，这使它成为本地的；扩展攻击者控制分散在网络上的多个实体，从而扩展其范围。

有意与非故意:故意攻击者故意制造攻击，而非故意攻击则是由错误的传感器或设备产生的网络事件。

在本文中，我们考虑了所有类型的攻击者，除了非故意的，因为它的可行性很难评估，因为它依赖于传感器的质量。我们还假设攻击者可以访问它的受害者的车辆(包括车载网络)。

五、方法

为了促进对自动车辆的安全和隐私问题的讨论，我们首先列出自律自动车辆的攻击表面(即攻击的入口点)，以及合作的自动化车辆。对于每次攻击，我们定义如下标准:

a) 方法:描述在攻击表面上执行的攻击。

b) 攻击的可行性(FA):描述执行攻击所需的知识水平。攻击的可行性也代表了发起这种攻击所需的技术专长。一些攻击可能需要技术或硬件方面的高技术专长，从而降低攻击的可行性。例如，攻击者可以通过发起物理攻击来提取车载单元(OBU)或路边单元的程序代码和密钥，这需要很高的技术知识。另一方面，廉价的现成产品可以使攻击变得非常可行。因此，攻击的可行性也取决于攻击者的资源。预算、人力和工具

是三个关键的资源，例如，预算可以是时间(例如，学习技术，编码软件工具)或金钱(购买设备或软件)。

c） 需要对目标车辆进行物理访问(PA):对目标车辆进行物理访问是否需要进行攻击?(是/否)

d） 容易被司机发现:司机能侦测到攻击吗? 附录 A-1 和附录 A-2 所列,象征“*”意味着“如果用户看着显示“(考虑到显示器显示警告和即将到来的操作例如)。这个标准假定司机对感兴趣的对象有一定程度的熟悉。

e） 系统检测方便:系统能检测到攻击吗?

f） 攻击成功率(PAS):根据之前的标准，我们评估攻击成功的概率。例如，一个高度可行的攻击(标准 b)但是很容易检测到(标准 d 或 e)不太可能成功。

g） 车辆的后果:描述车辆的直接后果，如进入最小风险条件。

h） 危险产生:在宏观的角度，描述攻击造成的危险(例如，交通干扰)。

i） 缓解技术:描述可用于减轻此类攻击影响的缓解技术。

标准 b、d、e 和 f 使用风险等级:低/中/高。低的可行性意味着所需的知识/设备不容易获得，需要时间来掌握。检测的低易用性意味着检测是困难的。例如，一个驱动不能在无线信道上检测到攻击，因为她/他无法看到在这个介质上发生了什么。风险等级是根据我们的知识来分配的，它的发展取决于设备的发展和它们的可达性。

标准“缓解技术”提出了一种通用的安全技术，可以用来预防或减轻攻击。我们把它作为一种通用的，作为缓解技术应该遵循的最好的技术概念。

我们不认为附录 A-1 和附录 A-2 是详尽的,而是旨在提高对自动车辆的安全和隐私问题的认识。

可以注意到，我们的方法类似于常见故障模式和效果分析(FMEA)。

FMEA 方法的目的是确定产品或过程的潜在失效模式，评估与这些失效模式相关的风险，将问题按重要性排序，并识别和实施纠正措施以解决最严重的问题[18]。我们将 FMEA 术语与我们的上下文相适应。例如，用“means”代替“failure”，“result For the vehicle”而不是“failure”，“create”而不是“severity”和“FA”，而不是“occurrence”。攻击树是分析系统和子系统安全性的另一种正式方法。例如，在[19]中使用攻击树来将对 V2V 通信的攻击形式化。然而，在我们的背景下，大量的攻击使树变得太大、太笨重。此外，攻击树并没有具体地整合检测部分，这是评估攻击成功概率的必要条件。

六、跨领域议题

自动化程度可能会降低驱动程序的干预能力，从而使攻击的后果更加严重。

事实上，最近通用汽车公司(尚未发表)的研究表明，在完全自动驾驶的连续时间间隔从 5 分钟到 30 分钟后，驾驶员基本上脱离了驾驶任务和对驾驶环境的监控，几乎完全依赖于自动化系统。因此，即使车辆自动化系统能够识别出需要系统脱离的威胁，司机也可能无法在合理的时间间隔内重新控制车辆(例如几秒钟)。对于自主和合作的自

动化车辆来说，技术难题之一是来自不同来源的数据的融合。在[20]和[21]中，作者提出了多传感器数据融合和数据聚类技术，实现了数据分类。然后，根据所考虑的数据类别，开发出不同的保护策略。的确，一个健壮的数据融合系统可能有助于识别网络攻击产生的异常输入，但只有在数据来源足够冗余的情况下，这才可能发生，而且攻击并没有影响到这些数据源中的大多数。如果只有两个信息来源关于车辆的某一特定状态，并且其中一个已经损坏，融合系统可能无法分辨哪个是有效的，哪个不是。然而，有了两个以上的源，就可以更容易地隔离一个异常的数据源(并且通过多个独立的数据源进行攻击需要攻击者的复杂程度大大提高)。对于合作的自动化车辆，从其他车辆或路边传递的数据可以作为另一个传感器输入到融合系统。然而，由于其设计者并不一定知道数据的来源(其他车辆或路边系统)已被充分地保护起来以防止被入侵，所以主机车辆系统对该数据的信任可能比从它自己的传感器获得的数据要少。这对于安全关键的车辆机动决策尤为重要。

附录 A- 1 自动驾驶车辆的攻击表面

目标	方法	攻击可行性	是否物理接触	被司机发现的可能	被系统发现的可能	成功率	车辆的后果	危险产生	缓解技术
基础设施标志	改变符号	低	不可用	高	低	中低	错误反应	交通骚乱	让交通标志变得更坚固；车载地图数据库；司机报告
	使其不可辨认	高	不可用	高	低	中低	错误反应，无反应	交通骚乱	让交通标志变得更坚固；地图数据库；司机报告
	移开符号	高	不可用	高	低	中低	无反应	交通骚乱	让交通标志变得更坚固；地图数据库；司机报告
机器视觉	仅从信息源头致盲	高	不接触	中	高	高	模式退化	司机被干扰	不同角度并联相机
	从其他可获取的信息源头致盲	高	不接触	中	高	高	关闭相机	无	不适用
	仅从信息	低	不接触	中	低	中	错误反	司机被	其他来源的数

	源头伪造紧急刹车信号灯						应	干扰	据
	从其他可获取的信息源头伪造紧急刹车信号灯	低	不接触	中	低	中	错误反应	司机被干扰	不适用
GPS	电子欺骗	高	不接触	低	中	高	错误定位	交通骚乱或者撞车	证明
	干扰	高	不接触	低	中高	高	不能获得精确的定位信息	除非有其他途径获取定位信息否则停车	Anti-Jam GPS 技术；高质量的 IMU
车载设备	植入恶意软件	中	USB 接触;其他不接触	低	中	中	取决于恶意软件的能力	取决于恶意软件的能力	娱乐信息和安全信息分离；侵入侦查系统
	头部装置攻击	中	接触	高	中	中	显示意外的信息	司机被干扰	对显示安全状态信息的保护
声学传感器	干扰	中	不接触	中低	低	低	关闭声学传感器	不适用	滤波器，频率分析
	伪造撞车的声音	高	不接触	中低	低	低	错误反应	交通骚乱	其他数据来源
	伪造超声波反射	中	不接触	低	低	低	错误的正面或负面的侦查障碍	交通骚乱或者低速的撞车	其他数据来源
雷达	无价值的东西	中	不接触	中	高	中	退化模式	交通骚乱	滤波器；其他数据来源

	特殊材料	低	不接触	中	低	中	不探测周围环境	碰撞	其他数据来源
	干扰	高	不接触	低	高	中	关闭雷达	交通骚乱	滤波器；其他数据来源
	幽灵车辆	高	不接触	中	中	中	错误探测	交通骚乱	滤波器；其他数据来源
激光雷达	干扰	高	不接触	低	高	中	关闭雷达，退化模式	车辆对情况认识减少	滤波器；其他数据来源
	特殊材料	高	不接触	中	中	中	错误侦查	交通骚乱	滤波器；其他数据来源
道路	描述修改	低	不适用	中	低	低	错误侦查	交通骚乱	司机报告
	攻击 led 灯	低	不适用	低	低	低	错误侦查	交通骚乱	司机报告
车载传感器	窃听（蓝牙等）	高	不接触	低	低	中	隐私泄露	无	车内安全
	窃听（CAN bus）	高	接触	中	低	中	逆向工程	无	车内安全
	注入CAN信息	中	接触	中	高	中	网上的错误信息	交通骚乱	车内安全
里程碑	磁性攻击	高	接触	低	低	中	错误定位和驾驶	交通骚乱	其他数据来源
传感器	对陀螺仪的热攻击	中	接触	低	低	低	错误定位和驾驶	交通骚乱	包装，车内安全
电子设备	EMP	低	不接触	低	高	中	对电子部件临时或永久损坏	车辆不能自动驾驶	EMP 防卫
地图	地图中毒	低	不接触	第	中	中	错误的机动	交通骚乱，事	地图发行者的证明

故

七、安全与隐私威胁:自动自动车辆的案例。

在此，我们通过列出攻击表面和描述在这个表面上可以执行的攻击来调查自动自动化车辆的潜在网络攻击。

自动车辆可以使用多个传感器感知其环境。最近的实现[22]-[25]使用不同的组件组合:测距传感器(激光雷达、雷达)、GPS 和斯坦福自动自动车辆地图;牛津 RobotCar 的立体摄像机和激光;立体相机，三维激光雷达，雷达，和全球定位系统的自动自动车辆。然而，未来的自动自动化车辆可能集成更多的组件，因此，我们考虑以下攻击表面。

基础设施标志:由道路运营商或政府机构安装的路标(静态或动态)通知司机。

机器视觉:用于物体检测的视频图像处理(道路、障碍物、路标等)。

GPS:全球定位系统，用于定位和定位集成地图。我们假设车辆包括多个 GPS2(例如，一个用于导航显示的 GPS，一个用于自动化)。

车载设备:包括用户携带的手持设备。它可以通过蓝牙、Wifi、Zigbee 或通用串行总线连接到信息娱乐系统。这可以代表一种售后服务、智能手机或平板电脑[26]。

声学传感器:识别经过训练的/已知信号的声学传感器。例如，碰撞声传感器检测到的碰撞速度比安全气囊传感器的速度快[27]，因此，可以用来提前触发安全气囊或紧急制动。该组件还考虑超声波系统，例如超声波声纳。

雷达:利用微波辐射(无线电波)来探测物体的主动系统。

道路:车辆行驶的材料/结构，包括道路的轮廓。

车载传感器:任何车载传感器，提供有关车辆内部状态的信息(车轮的转速、轮胎压力等)。

里程表传感器:用于惯性导航的车轮编码器和惯性传感器(加速度计、陀螺仪等)。惯性测量相对于远程攻击的相对阻力是军事无人系统倾向于使用惯性测量单元(imu)作为主要导航传感器的原因之一。

电子设备:一般来说，车辆是一种复杂的电子设备，但这也适用于汽车使用者使用的个人移动设备。

地图:在非实时检测道路[28]的情况下，使用地图给自动自动车辆提供纵向和横向的方向。

高威胁:根据表 I，优先级是确保高威胁(参见“高”栏“成功概率”)，即摄像头(盲)和 GPS 欺骗/干扰。GPS 干扰很便宜(大约 20 美元)，一些更昂贵的 GPS 干扰器甚至超过了干扰，在我们的系统中执行 GPS 欺骗(中等威胁)，在那里他们复制信号并提供错误的位置[29]。一个专业的汽车窃贼可以通过使用 GPS/GSM 干扰器来阻止汽车的防盗系统知道和报告车辆的位置，从而继续他/她的偷窃业务。此外，由于环境限制，GPS 信号可能难以探测，因此 GPS 干扰很难探测到。提出了多种缓解措施[30]:系统级对策，基

于接收天线的对策，基于接收的对策，终端/应用水平的对策，以及反 GPS 干扰和欺骗干扰的后台对策[31], [32]。

相机可以被高亮度的红外发光二极管或红外激光器所蒙蔽，这是很便宜的(大约 0.75 美元/LED)。因此，一种由红外发光二极管(IR)发光的缓解方法是滤除颜色。然而，这种过滤也可以被抵制。例如，军事解决方案是使用可以随机改变颜色的“波长-敏捷”激光，使任何过滤都无用[33], [34]。

中等威胁:中等威胁是电磁脉冲(EMP)、地图中毒、雷达混乱、激光雷达混乱、车内设备感染和车内传感器的操纵。

EMP 攻击的目标是破坏电子设备，如机载传感器和处理器(ECU)。EMPs 很容易[35], [36]，而且创建起来也很便宜。例如，Yeh[36]创建了一个大约 300 美元的 EMP 生成器。然而，我们保持低的可行性，因为发电机不够强大，无法关闭整个汽车(但足够用于智能手机等小型电子设备)。

Jeske 展示了地图如何被毒化的一个例子[37]。在他的攻击中，他展示了攻击者如何控制导航系统，并且在浮动汽车数据分布广泛的情况下，可以主动控制交通流。这次攻击表明，交通数据的真实性无法得到保证。然而，自动自动化系统通常依赖于地图来驱动车辆，因此，地图应该被认证。

对雷达的一种攻击是使用数字射频存储器(DRFM)中继器创建幽灵车辆。DRFM 将接收到的信号进行数字化，并在数字存储器中存储一个一致的拷贝。根据需要，信号被复制并重新传输。作为原始信号的一种相干表示，发射雷达将不能将它与接收的其他合法信号区别开来，并将其作为障碍物进行处理。针对雷达干扰问题，提出了一些军事应用的对策。特别是 Lu 等[38]提出的一项对策，旨在取消 DRFM 雷达干扰。

与手持设备的车辆连接的增长的同时，车辆的网络风险正在增加。当手持设备与车辆相连时，病毒和恶意软件通过车载娱乐系统或车辆信息终端侵入汽车电子产品。Onishi[39]使用常用的漏洞评分系统计算器来评估车载网络的网络安全漏洞。由于感染率为 1%(病毒或恶意软件感染)，死亡和受伤的总人数为 4230 人，相当于美国全国每年(2008 年)所有交通事故死亡人数的 10%[39]。由 EVITA 项目提出了车内传感器和诱骗网络的一些保护机制。感兴趣的读者被转发到 EVITA 交付品。

即使攻击成功的概率是一个重要的指标，因为它显示了发生的可能性，对目标车辆的直接后果是非常重要的。事实上，如表附录 A-1 所示，对相机的盲目攻击有很大的成功几率，但如果有其他信息来源(如照相机、雷达和激光雷达)，其直接后果就是关闭相机。因此，驾驶员和自动化的后果是低的，因为车辆可以继续充分发挥。这证明了在一个关键直接后果(例如:错误反应、禁用车辆自动化和崩溃)中成功的低或中等概率也应该被考虑。

在缓解技术柱中，我们表示“其他数据来源”其他传感器或远程传感器(即:其他车辆)。从表中可以得出这样的结论:自律自动化车辆应该始终考虑不同的信息来源(在驱

动环境中可以使用的程度)，以确保足够的冗余水平，这允许识别相互冲突的信息，并减少决策过程中的不确定性。使用其他数据来源将会增加自动化系统的成本，但这是值得的，因为它极大地改进了决策，从而提高了用户的安全性。一个挑战是数据融合到最合适的动作(见第四节)。表中提出了其他缓解技术，如认证、入侵检测系统或防干扰 GPS，这些技术要求更换设备或更新软件。它还可能增加机载系统的计算开销。

自律自动车辆的局限性包括有限的视线和它不能“透视”的物体/角落。例如，一辆自动驾驶的自动车辆到达山顶时，无法扫描即将到来的道路，因此，完全相信它的位置和地图来决定下一个轨迹。因此，自动自动化车辆可以从其他车辆的远程信息中获益，因为它们可以提供其他的观点。在下一节中，我们将研究合作自动化车辆的攻击表面，并展示了将合作技术与自动化技术结合起来以实现安全和隐私目的的好处。

附录 A- 2 协同自动化车辆的攻击面

目标	方法	攻击可行性	是否物理接触	被司机发现的可能	被系统发现的可能	成功率	车辆的后果	危险产生	缓解技术
基础设施（RSU）	伪造 WSA	高	不接触	低	低	高	给司机错误的反应和通知	取决于错误信息的本质	证明
	地图数据库中毒	高	不接触	中高	中	中高	系统产生错误的决定	交通骚乱或者安全问题，取决于攻击的本质	真实性检查
	DoS	高	不接触	低	高	中	不能处理新信息	不适用的信息需求，导致安全信息被拒绝	证明；撤回
	关闭公共设施	低	不接触	中	高	中	没有信息可获得	无效的信息需求，导致安全信息被拒绝	使公共设施更坚固
安全系统	伪造 LTC	低	不接触	低	中	低中	颁发错误的证书	无效的信息发送	证明
	伪造 CRL	中	不接触	低	中	中	颁发错误的证书取消列表	忽略来自有效车辆的信息	证明

	伪造 PC	中	不接触	低	中	中	颁发错误的笔名证书	无效的信息发送	证明
	拒绝笔名分配	高	不接触	低	高	中	隐私减少	无	不当行为报告
	存储 LTC-ID	低	不接触	低	低	低	如果权力被盗用则隐私被破坏	无	CoPRA
其他车辆	伪造 BSM	高	不接触	低	低	高	错误反应	错误响应	证明；其他信息来源
	DoS	高	不接触	低	高	中	不能处理新信息	无效的信息需求，导致安全信息被拒绝	证明；撤回
	地图数据库中毒	高	不接触	中高	中	中高	错误反应	交通骚乱或者安全问题，取决于攻击的本质	不当行为报告
	欺骗 DCC 机制	中	不接触	低	中	中	产生更多需要处理的信息	引导条件降低	不当行为报告
	远程闪存固件，重新启动	低	不接触	低	高	低	一段时间内系统关闭	没有合作的信息	预防远程控制
	阻塞笔名交换	中	不接触	低	中	中	隐私减少	无	不当行为报告
任何地方	通过发布假命令或 dos 的外部通信链接攻击车	低	不接触	低	低中	中	潜在的使车辆报废或发送不安全的移动命令	交通骚乱	不当行为报告；安全编码

辆 的 CAN-bus								
位置跟踪	中	不接触	低	低	中	隐私减少	无	笔名系统

八、安全与隐私威胁:合作自动化车辆的案例。

一个合作的自动化车辆使用无线通信技术来执行车辆到 x 的通信(V2X)。专用的短程通信(DSRC)或局部热平衡是 V2X 的潜在技术，但在本文中，我们是技术不可知论者。需要注意的是，考虑到视距通信(如可见光、红外、雷达作为载波)，由于其固有的有限范围，V2X 通信可能会降低威胁级别。

除了自动自动化系统提供的攻击面外，表 II 显示了以下攻击表面(即从哪里可以发动攻击)为合作的自动化车辆。

基础设施:基础设施定义了不移动的车辆通信中涉及的实体集。路边通信单元、地图服务器和交通信号都是基础设施实体的例子。这些实体可以广播诸如路边警报和信号相位和定时[40]等信息。

安全系统:安全系统包括管理安全相关信息的基础设施实体。长期证书颁发机构、笔名证书颁发机构(PCA)和注册管理局(RA)都是认证机构的例子。

其他车辆:配备合作系统的其他车辆，能够以可理解的形式向接收车辆发送信息。

任何地方:这个类别包括来自任何地方的攻击(基础设施、安全系统、其他车辆)。

人们应该注意到，这些攻击表面除了表 1 中所显示的外，一个不同之处是，高威胁是触发错误反应的，因为它们直接影响到用户的生命。因此，成功的概率与直接结果之间有更紧密的联系。其他的威胁并不会危及整个自动化系统，但主要是抑制一个信息来源(即系统不认为这个实体是定义期间的信息来源)。因此，引发错误反应的攻击被认为是最危险的，因为它们直接影响到用户的生命，因此具有最高的风险。

高威胁:高威胁是虚假安全信息的注入和地图数据库中毒。

在第一个高威胁中，基础设施(RSU)或邻居车辆可以注入虚假信息(WAVE 服务广告，基本安全信息(BSM))，它会产生错误的反应(例如，假刹车)，对司机、乘客和周围的车辆造成威胁。

缓解技术主要要求建立一个认证系统和一个错误行为检测系统。实际上，经过验证的车辆可以发送虚假信息，只有通过错误行为检测系统才能检测到。错误行为检测系统是 OBU 的一个软件模块，而认证机制可能需要一个更复杂的系统，以建立一个公共密钥基础设施。

第二大威胁是地图数据库中毒。这种攻击不同于自动自动车辆的“地图中毒”，因为中毒攻击并不是针对收集浮动汽车数据的在线服务器，而是针对本地存储在车辆上的地图数据库。OBU 在所谓的本地动态地图(欧洲的 LDM)或地理信息系统(美国地理信

息系统)中存储所有信息的内容(新的兴趣点、障碍、建设站点等)。从真实世界的本地表示，错误行为检测，网络数据聚合，以及更广泛的，决策。因此，中毒数据库将影响整个合作系统。在这里，缓解技术是一个错误行为检测系统，它在将数据存储到地图数据库之前执行可信性检查。

中等威胁:人们可以注意到，与自主自动化车辆相比，合作自动化车辆的低等威胁更小，但中等威胁更大。拒绝服务(DoS)会导致车辆不处理任何新传入的消息，因为系统中有太多的消息要处理。其后果可能是信息的不确定性增加(来自传感器)，但也可能是对安全关键信息的否认。作为缓解技术，身份验证机制将识别攻击者，并且可以触发撤销过程，以防止该车辆在未来发生错误行为。但是，请注意，身份验证和撤销机制并不能防止无线电通信的干扰，这是一种可行的、廉价的 DoS 攻击，而且很难减轻。与 DoS 攻击类似，攻击者可以通过将高通道繁忙比率(CBR)发送给他/她的单跳邻居来降低通道条件并增加发送或接收消息的数量，从而欺骗分布式拥塞控制(DCC)[42]机制。然而，由于 DCC 机制提供了最低的服务质量，这种攻击在空间(单跳)和影响方面是有限的。

对安全系统的攻击主要是中等威胁。伪造的长期证书(LTC)或假名证书(PC)将产生无效的消息(即无效的签名)，然后将被接收者忽略。存储假证书撤销列表(CRL)的 OBU 将拒绝来自有效 OBUs 的消息，这将危及被攻击车辆的合作系统。事实上，这可能是一个严重的安全问题，如果这导致没有警告或避免崩溃。一种缓解方法是在使用前对 CRL、LTC 和 PC 进行身份验证。

合作系统将允许远程访问汽车数据库。例如，CarSpeak[43]使汽车能够查询和访问被其他车辆捕获的感觉信息，其方式与它从本地传感器获取信息的方式类似。这允许从远程传感器读取数据，但应该确保不可能注入消息。Rouf 等[44]提出了另一个远程访问 CAN 总线而没有物理访问的例子。作者发现，在 CAN 总线控制器和无线轮胎压力监测传感器之间的数据传输机制中存在漏洞，这使得误导性数据被注入到车辆的系统中，并允许远程记录特定车辆的运动情况。研究人员使用了 1500 美元的设备，包括无线电传感器和特殊软件，窃听和干扰两个不同的轮胎压力监测系统。压力传感器包含独特的 id，所以仅仅窃听就能使车辆的远程识别和跟踪。除此之外，读数可以被改变，并在仪表板上形成警告灯，或者甚至完全摧毁 ECU。

作为合作技术(V2X)，可以通过无线电信标显示位置、速度和方向等信息，它本质上支持短期位置跟踪。为了保护乘客的长期隐私，一项缓解措施就是使用匿名管理系统。因此，车辆将根据提供足够安全和隐私的隐私政策改变假名[45]。这并不会影响自动化系统本身，因为自动驾驶任务没有直接的结果。因此，这种威胁被认为是一种媒介，但应加以处理以确保用户的接受。

我们可以得出这样的结论:缓和技术与用于保护 V2X 通信的技术相似[46]-[48]，但成功攻击的后果是不同的。因此，由于这些潜在的危及生命的后果，实施减缓技术可

能会有所不同，以确保新的要求。例如，安全机制必须更有效率(即较低的通信和计算开销)。然而，自动化的公路系统，所有车辆都是合作的和自动化的，将启用不同的安全机制。由于自动化系统更可预测，它将提供一个更稳定的网络，从而使对称密码术(它比当前的标准化非对称加密技术更轻量级和更有效)。车队是另一个很好的例子，可以应用组签名方案。

九、结论和未来的挑战

本文对自动车辆的网络安全威胁进行了识别，并对这些威胁的严重程度和潜在的缓解或克服这些威胁的策略进行了评估。这是一项初步的探索性研究，以确定在开发车辆自动化系统时需要面对的挑战，并开始将这些挑战中最重要的部分放在首位。本文最重要的一个方面是对自主和合作的自动车辆的并行考虑，指出它们面临的威胁和可以用来管理这些威胁的策略之间的相似之处。对于一个或另一个方法的相对安全性，没有做出任何价值判断，但对于这两种方法来说，需要考虑安全威胁。合作自动化车辆的额外信息源可以提供额外的工具来验证车辆状态，确认或对抗攻击，但它们也可以为攻击者提供额外的伤害机会。因此,车辆必须有足够的冗余在任何输入源允许共识的存在决定攻击一个形态,特别是如果形态包含了多个信息来源(e.g.GPS 本地化和合作通信),如果对虚假信息的反应很可能是极具破坏性的。系统还应该设计成在跨多个模式的协同攻击时优雅地失败。

本文的主要目标是提高人们对该问题重要性的认识，并鼓励其他人将他们对潜在网络安全威胁的想法添加到自动化车辆上，并提出可以克服这些威胁的对策。

这一初步研究发现，GNSS 欺骗和注入虚假信息是最危险的攻击（最可能的或最严重的)。在自动驾驶车辆中，全球导航卫星系统(GNSS)在精确的地图上扮演着关键角色。因此，操纵 GNSS 数据可能会引起不稳定和不准确的操作，这可能会危及乘客的生命。因此，安全 GNSS 信号是强制性的。选择性可用性或反欺骗模块(SAASM)硬件是一个解决方案，但代价昂贵且访问受限。在合作的自动化车辆中，额外的高威胁是注入虚假信息，从而引发不恰当的反应。除了保护外部攻击者的身份验证之外，还需要进行错误行为检测来检测内部和非故意的攻击。错误行为检测系统的部署需要 OBU 的软件更新，但也需要对当前标准化安全体系结构(如 ETSI 参考体系结构)进行根本性的更改。

致谢

作者想要感谢 E. Fok, Z. Brooks，以及由 J. Christian Gerdes 领导的斯坦福大学动态设计实验室的成员们对他们的深刻见解和建议。

参考文献

[1] M. Williams,“PROMETHEUS-The European Research Programme for Optimising the Road Transport System in Europe,” in Proc. IEEE Colloq. Driver Inf., 1988, pp. 1–9.

[2] S. E. Shladover, “‘AHS Demo ’97 Complete Success’ and ‘The GMPATH Platoon Scenario’,” Intellimotion, vol. 6, no. 3, pp. 1–3, 1997.

[3] A. Benmimou, M. Lowson, A. Marques, G. Guistiniani, and M. Parent, “Demonstration of advanced transport applications in CityMobil project,” Transp. Res. Rec., J. Transp. Res. Board, no. 2110, pp. 9–17, 2009.

[4] T. Robinson, E. Chan, and E. Coelingh, “Operating platoons on public motorways: An introduction to the SARTRE platooning programme,” in Proc. 17th ITS World Congr., 2010, pp. 1–11.

[5] Y. Suzuki et al., “Development of automated platooning system based on heavy duty trucks,” in Proc. 17th ITS World Congr., 2010, pp. 1–11.

[6] E. van Nunen, M. Kwakkernaat, J. Ploeg, and B. D. Netten, “Cooperative Competition for Future Mobility,” IEEE Trans. Intell. Transp. Syst., vol. 13, no. 3, pp. 1018–1025, Sep. 2012.

[7] S. E. Shladover et al., “Automated vehicle control developments in the PATH program,” IEEE Trans. Veh. Technol., vol. 40, no. 1, pp. 114–130, Feb. 1991.

[8] R. E. Fenton and R. J. Mayhan, “Automated highway studies at the Ohio State University-an Overview,” IEEE Trans. Veh. Technol., vol. 40, no. 1, pp. 100–113, Feb. 1991.

[9] E. D. Dickmanns, “Vision for ground vehicles: History and prospects,” Int. J. Veh. Auton. Syst., vol. 1, no. 1, pp. 1–44, 2002.

[10] M. Wolf, A. Weimerskirch, and C. Paar, “Security in Automotive Bus Systems,” in Proc. Workshop Embedded IT-Security Cars, 2004, pp. 11–12.

[11] T. Hoppe, S. Kiltz, and J. Dittmann, “Security threats to automotive CAN networks-practical examples and selected short-term counter-measures,” in Proc. Comput. Safety, Rel., Security, 2008, pp. 235–248.

[12] F. Sagstetter et al., “Security challenges in automotive hardware/software architecture design,” in Proc. Conf. DATE, 2013, pp. 458–463.

[13] K. Koscher et al., “Experimental security analysis of a modern automobile,” in IEEE Symp. Security Privacy, May 2010, pp. 447–462.

[14] S. Checkoway et al., “Comprehensive experimental analyses of automotive attack surfaces,” in Proc. 20th USENIX SEC, 2011, pp. 1–16.

[15] “Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems,” Surface Veh. Inf. Rep. J3016, Jan. 17, 2014.

[16] A. Panchenko and L. Pimenidis, “Towards practical attacker classification for risk analysis in anonymous communication,” in Proc. 10th IFIP TC-6 TC-11 Int. Conf. CMS, 2006, pp. 240–251.

[17] M. Raya and J.-P. Hubaux, “Securing vehicular Ad Hoc networks,” J. Comput.

Security, vol. 15, no. 1, pp. 39–68, Jan. 2007.

[18] SAE International, SAE J1739, Potential Failure Mode and Effects Analysis in Design (Design FMEA), Potential Failure Mode and Effects Analysis in Manufacturing and Assembly Processes (Process FMEA), Jan. 15, 2009.

[19] A. Aijaz et al., “Attacks on inter vehicle communication systems—An analysis,” in Proc. 3rd Int. WIT, 2006, pp. 189–194.

[20] B. Khaleghi, A. Khamis, F. O. Karray, and S. N. Razavi, “Multisensor data fusion: A review of the state-of-the-art,” Inf. Fusion, vol. 14, no. 1, pp. 28–44, 2013.

[21] N.-E. E. Faouzi, H. Leung, and A. Kurian, “Data fusion in intelligent transportation systems: Progress and challenges—A survey,” Inf. Fusion, vol. 12, no. 1, pp. 4–10, Special Issue on Intelligent Transportation Systems, 2011.

[22] M. Montemerlo et al., “Junior: The Stanford entry in the urban challenge,” J. Field Robot., vol. 25, no. 9, pp. 569–597, Sep. 2008.

[23] P. Newman et al., “Navigating, recognising and describing urban spaces with vision and laser,” Int. J. Robot. Res., vol. 28, pp. 1–28, Oct. 2009.

[24] J. Levinson et al., “Towards fully autonomous driving: Systems and algorithms,” in Proc. IEEE IV Symp., 2011, pp. 163–168.

[25] A. Geiger et al., “Team AnnieWAY’s Entry to the 2011 Grand Cooperative Driving Challenge,” IEEE Trans. Intell. Transp. Syst., vol. 13, no. 3, pp. 1008–1017, Sep. 2012.

[26] J. Joy, A. Raghu, and J. Joy, “Architecture for secure tablet integration in automotive network,” in Proc. FISITA World Automotive Congr., 2013, pp. 683–692.

[27] M. Feser, D. McConnell, T. Brandmeier, and C. Lauerer, “Advanced crash discrimination using crash impact sound sensing (CISS),” SAE World Congress & Exhibition, Apr. 2006.

[28] Y. Gao, Y. Song, and Z. Yang, “A real-time drivable road detection algorithm in urban traffic environment,” in Proc. ICCVG, 2012, pp. 387–396.

[29] Royal Academy of Engineering, Global Navigation Space Systems: Reliance and Vulnerabilities, Royal Academy of Engineering, 2011.

[30] C. Dixon, C. Hill, M. Dumville, and D. Lowe, “GNSS vulnerabilities: Testing the truth,” Coordinates Mag., vol. 8, no. 3, pp. 13–20, Mar. 2012.

[31] T. E. Humphreys, B. M. Ledvina, M. L. Psiaki, B. W. OHanlon, and P. M. Kintner, Jr., “Assessing the spoofing threat: Development portable GPS civilian spoofer,” in Proc. 21st Int. Techn. Meet. Satellite Division ION GNSS, 2008, pp. 2314–2325.

[32] G. Hancke, “Security of embedded location systems,” Secure Smart Embedded Devices, Platforms and Applications, pp. 267–286, 2014.

[33] M. Naimark, How to ZAP a camera: Using lasers to temporarily neutralize camera sensors, [Online; accessed 27-January-2014], 2002. [Online]. Available:

http://www.naimark.net/projects/zap/howto.html

[34] K. N. Truong, S. N. Patel, J. W. Summet, and G. D. Abowd, “Preventing camera recording by designing a capture-resistant environment,” in Proc. 7th Int. Conf. UbiComp, 2005, pp. 73–86.

[35] E. Aerospace, High-Power Compact Microwave Source for Vehicle Immobilization, Nov. 2011, [Online; accessed 27-January-2014]. [Online]. Available: https://www.ncjrs.gov/pdffiles1/nij/grants/236756.pdf

[36] D. Yeh, Electromagnetic pulse generator, [Online; accessed 27-January-2014]. [Online]. Available: http://72.52.208.92/~gbpprorg/mil/herf/FinalReportDavidYeh.pdf

[37] T. Jeske, “Floating car data from smartphones: What Google and Waze know about you and how hackers can control traffic,” in Proc. BlackHat Europe, Mar. 2013, pp. 1–12.

[38] G. Lu, D. Zeng, and B. Tang, “Anti-jamming filtering for DRFM repeat jammer based on stretch processing,” in Proc. 2nd ICSPS, 2010, vol. 1, pp. 78–82.

[39] H. Onishi, “Paradigm change of vehicle cyber security,” in Proc. 4th Int. Conf. CYCON, 2012, pp. 1–11.

[40] “Dedicated short range communications (DSRC) message set dictionary,” Warrendale, PA, USA, SAE J2735, Draft Rev. 35, 2014, to be published.

[41] N. Bißmeyer, J. Petit, and K. M. Bayarou, “CoPRA: Conditional pseudonym resolution algorithm in VANETs,” in Proc. 10th IFIP/IEEE Annu. Conf. WONS, 2013, pp. 9–16.

[42] Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part, ETSI TS 102 687 V1.1.1, 2011, ETSI TC ITS.

[43] S. Kumar et al., “CarSpeak: A content-centric network for autonomous driving,” SIGCOMM Comput. Commun. Rev., vol. 42, no. 4, pp. 259–270, Aug. 2012.

[44] I. Rouf et al., “Security and privacy vulnerabilities of in-car wireless networks: A tire pressure monitoring system case study,” in Proc. 19th USENIX Conf. Security, 2010, pp. 1–16.

[45] S. Lefèvre, J. Petit, R. Bajcsy, C. Laugier, and F. Kargl, “Impact of V2X privacy strategies on intersection collision avoidance systems,” in Proc. 5th IEEE VNC, 2013, pp. 71–78.

[46] Intelligent Transport Systems (ITS); Security; Security Header and Certificate Formats, ETSI TS 103 097 V1.1.1, 2013, Standard, TC ITS.

[47] Intelligent Transport Systems (ITS); Security; Trust and Privacy Management, ETSI TS 102 941 V1.1.1, 2012, ETSI TC ITS.

[48] Intelligent Transport Systems (ITS); Security; Security Services and Architecture, ETSI TS 102 731 V1.1.1, 2010, ETSI TC ITS.

附录 B 外文原文