Tire-Road Friction Estimation Utilizing Smartphones

Michael Jaynes, Ram Dantu
Department of Computer Science & Engineering
University of North Texas
Denton, Texas
michael.jaynes@my.unt.edu, rdantu@unt.edu

***Abstract*—Tire-road friction is an important parameter for a number of different safety features present in modern-day vehicles, and the knowledge of this friction may also prove useful to the driver of a vehicle while it is in motion. In particular, this information may help inform a driver of dangerous low-traction situations that he or she may need to be aware of. Furthermore, since a growing number of drivers have access to Bluetooth-enabled smartphones, it is worth exploring how these devices may be leveraged in concert with vehicular CAN-bus networks to provide valuable safety information.**

Keywords—vehicular safety; CAN-bus; friction; mobile computing; traction

I. Introduction

Vehicular safety is an ever-present concern in the automobile industry, as vehicle manufacturers are always trying to find ways to ensure that their cars can withstand damage and keep all passengers safe, as well as pedestrians and other drivers. More and more recently, they have turned to technology to help assist in this effort, introducing features such as rear-collision cameras and parking assists, collision avoidance systems, and many other safety features. In addition, modern automobiles have incorporated advanced computer networks connected to many different sensors which produce a large quantity of valuable safety data for use in these applications, as well as in potential safety applications on the mobile platform.

More specifically, an estimation of the tire-road friction coefficient for each individual wheel can help provide for greater stability and traction control, whether in driver-operated or automated systems. There has been fair amount of research done in this area focused on either the incorporation of additional vehicular sensors and on-board computer systems or off-line data analysis, but there has been comparatively little work done in the area of mobile computing solutions to questions of vehicle dynamics. This paper aims to encourage more exploration of this research path by demonstrating the feasibility of such an approach for one particular model of consumer automobile using an Android application and a very inexpensive piece of hardware.

It should be noted that there have been other Android applications developed specifically for the purpose of vehicular maintenance that can connect to an intra-vehicular network through that vehicle's Onboard Diagnostics (OBD-II) port. The important distinction between this work and many other applications is that most applications communicate with a vehicle through the set of standardized, mandatory OBD-II PIDs, which largely provide data needed for state emissions testing. This application, on the other hand, aims to access and utilize several more vehicle dynamics parameters, discussed in Section III, which are more relevant to vehicular safety.

II. Background and Related Work

A. The Automotive CAN-bus

Since 2008, all consumer vehicles sold in the United States implement some form of a Controller Area Network (CAN) bus for the purpose of vehicle diagnostics. The CAN-bus protocol does not utilize traditional network addressing; rather, the CAN packets are broadcast with either 11-bit or 29-bit CAN IDs to all of the nodes (mostly Electronic Control Units or ECUs) of the network, and these nodes decide whether or not the packets will contain data relevant to that node. Some vehicles also contain a high-speed and a low-speed CAN-bus, which allows the vehicle to delegate safety and driving-essential functioning to the high-speed bus and entertainment or other nonessential functions to the low-speed CAN. Many of these CAN-bus networks contain a significant amount of vehicle dynamics data that is constantly being broadcast while the vehicle is running. Such data can potentially be utilized in novel and vital ways by researchers if it can be collected and correctly scaled. However, since most vehicle manufacturers consider the addressing schemes of their vehicular networks a trade secret, this data is difficult to access and parse for the purpose of developing safe-driving applications, and there haven't been major developments studying mobile computing solutions for data collection or safety applications.

There have been several pieces of literature published recently that delve into the characteristics of the CAN-bus network, along with the security flaws inherent within the protocol. Several of these are more abstract in nature, but there are several that focus on practical experiments that demonstrate in no uncertain terms some of the vulnerabilities of these vehicle networks [1][2]. Two examples of these serious security weaknesses are the system's vulnerability to Denial of Service (DoS) attacks and the lack of any authentication methods for CAN messages. The first of these is due to the priority-based arbitration of CAN messages; the 11-bit or 29-bit CAN IDs also serve as arbitration IDs, meaning that the value of this ID determines the priority that packet is given by the bus. Therefore, if an attacker is able to send a flood of packets at a high priority, the vehicle will have extreme difficulty processing other, lower-priority packets that are

IEEE IRI 2014, August 13-15, 2014, San Francisco, California, USA
978-1-4799-5880-1/14/$31.00 ©2014 IEEE

necessary for ordinary functioning of the vehicle. Furthermore, as has been discussed in the literature, the CAN packet protocol does not include authentication fields or even source identification, which means that vehicle components that do not implement any form of security layer on top of the CAN-bus cannot determine if a message being broadcast is genuine or not. These are important concerns for this project, due to the fact that a smartphone application communicating over Bluetooth with a vehicle's CAN-bus while it is in motion creates new security challenges that must be careful addressed in future works.

B. Friction Estimation Algorithms

There has been significant research directed at various approaches to estimating the friction coefficient between the road and the tires of a vehicle in motion. Several of these [3][4][5][6] are based at least partially on wheel slip ratio or slip-slope, and some of these also incorporate torque [4], GPS measurements [4], or longitudinal acceleration [4][5][6]. Based on the parameters that are easily accessible through the test vehicle's CAN-bus, this research focused on the algebraic, longitudinal acceleration-based approach for average friction coefficient found in [5].

The longitudinal friction coefficient between the road and a tire, μ_x, is given by:

$$\mu_x = \frac{F_x}{F_Z} \tag{1}$$

where F_x is the longitudinal force and F_Z is the vertical force acting upon the tire. Newton's second law of motion can be used to determine the longitudinal effort for both the front and rear wheelbases of the vehicle:

$$F_{Xf} = M_{eq\text{-}f}\,\gamma_x,\ M_{eq\text{-}f} = \frac{F_{Zf}}{g}\ ;\ F_{Xr} = M_{eq\text{-}r}\,\gamma_x,\ M_{eq\text{-}r} = \frac{F_{Zr}}{g}$$

where M_{eq} is the equivalent mass for either the front or the rear of the vehicle, and γ_x is the longitudinal acceleration. These equations can be combined with (1) to demonstrate that the front and rear longitudinal friction coefficients are equal and determined by longitudinal acceleration:

$$\mu_{xf} = \mu_{xr} = \frac{F_{xf}}{F_{Zf}} = \frac{F_{xr}}{F_{Zr}} = \frac{\gamma_x}{g} \tag{2}$$

However, it is important to note that this will provide something that resembles an average front and rear friction coefficient in driving situations where the friction is not equivalent between the front and rear of the vehicle. Future work on this project will strive to calculate individual wheel friction coefficients.

III. Experimental Setup

This research was performed on a recent model of a popular consumer automobile. Naturally, due to the closed nature of vehicle network structures, the application cannot yet be made universal, even with regards to all vehicles made by the same manufacturer. One hope for the overall big picture of this type of research is that vehicle manufacturers will be more willing to work closely with researchers in the future, particularly as the industry works towards autonomous vehicles.

A. CAN Sniffing

Since modern automobiles implement some form of CAN-bus architecture, the foundation for this work involved the use of a CAN traffic sniffing tool. This tool allowed reverse engineering of various useful CAN arbitration IDs utilized by the vehicle's CAN-bus on both the high-speed and medium-speed CAN networks (this vehicle contained both). While watching the traffic on the CAN network, it is possible to see which bytes are changing in the various packets being broadcast. When certain actions are performed, such as opening doors, turning the steering wheel, or unbuckling and buckling seatbelts, it is possible to isolate the specific bytes that are changing in response. Furthermore, using the manufacturer's proprietary, module-based diagnostic software, several 29-bit CAN IDs were discovered that correspond to data request packets that are broadcast to the appropriate Electronic Control Units (ECUs) when the diagnostic software is run on the vehicle. The values in the responses to these packets can then be interpreted and scaled based on the interpretations of the raw bytes that the diagnostic software provides. Through this method, the researchers were able to determine the correct CAN packet to send in order to request and parse such data parameters as steering wheel angle, longitudinal acceleration, vehicle velocity, acceleration and velocity of each wheel, and brake pressure, among others.

B. Android Application

There is a considerable variety of options for devices that are compatible with the OBD-II diagnostic port, but the one that was chosen for this project was chosen for its affordability and Bluetooth capabilities. The device chosen was a Bluetooth-enabled Elm327 device. This device plugs into the vehicle's OBD-II port, which is located underneath the steering column on most vehicles, and sends its output to the Bluetooth connection. While this particular model is relatively devoid of extra features, the Elm327 integrated circuit which powers it still provides a command set (known as AT commands) that allows for powerful CAN-bus communications. It is these commands that are utilized by the Android application.

With the Elm327 device plugged into the test vehicle's diagnostic port (shown in Fig. 1), the smartphone that is running the application needs to be paired with the device. After pairing is completed, the application can be run. When the application starts, a check is performed to ensure that there is both a Bluetooth adapter and a paired OBDII device. If not, the application will not run. Once the paired device is found and instantiated to an Android BluetoothDevice class, the user is given a choice of whether or not to log the data that is

Fig. 1. Elm327 device plugged into OBDII port.

collected. After pressing a button on the application user interface to begin logging, a BluetoothSocket class is opened, and the communication between the device and the application is run on a separate thread from the main user interface thread. When this thread is initialized, several AT commands must be sent to the Elm327 device in order to direct it to the correct CAN-bus network and the appropriate CAN arbitration IDs. These are sent very quickly, and it generally takes less than a second for the application to begin parsing, logging, and displaying the data. A screenshot of this application while it is running can be seen in Fig. 2. This screenshot shows the current readings and calculations for several data parameters, mostly focused on speed and acceleration. There are further parameters that the application is designed to collect as well, but those were disabled during the testing that was done for this paper, in order to improve the frequency of data collection.

In addition to converting the bytes received from the CAN-bus and scaling them appropriately, the application makes some calculations. As can be seen in Fig. 2, the wheel slip ratio for each wheel is calculated by the application. This calculation is performed using the equations found in [4]:

$$\sigma_x = \frac{(r_{\text{eff}}\omega_w - V_x)}{V_x} \text{ during braking} \quad (3)$$

$$\sigma_x = \frac{(r_{\text{eff}}\omega_w - V_x)}{(r_{\text{eff}}\omega_w)} \text{ during acceleration.} \quad (4)$$

Since the velocity is provided for each wheel in the same data payload as the vehicle's longitudinal velocity and longitudinal acceleration, the program simply checks to see if the vehicle is slowing down (negative acceleration) and calculates the wheel slip values using the appropriate equation.

IV. Results

To collect the data for this paper, the test vehicle was driven while the application was running and logging data. It was driven along the same short stretch of straight road five times. The road condition was normal dry asphalt. At the end of each period of data collection, a braking action of normal pressure was applied. Two of the plots of these tests are displayed in Figs. 3 and 4.

Another series of tests were run on a similar stretch of asphalt road while the road was wet and it was raining. Similar

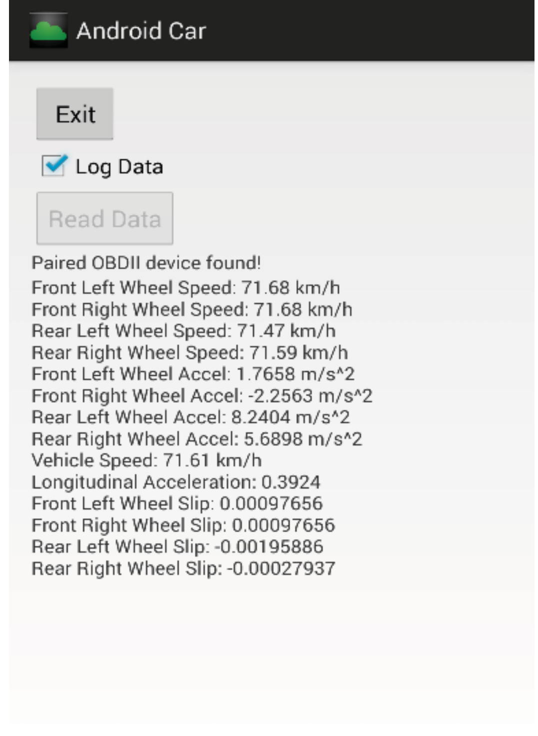

Fig. 2. Screenshot of Android application running.

to the previous tests, a braking action occurred at the end of the test, although these tests were slightly longer. Two plots of the average friction coefficient for those tests are shown in Figs. 6 and 7. There seems to be a sharper increase in friction coefficient when the braking action occurred during the wet road trials.

V. Conclusions and Future Work

There are several avenues of future research for this project. Naturally, increasing the accuracy of the estimations is a clear goal, and there are a number of additional vehicle dynamics that can be taken into consideration in order to achieve this goal. Furthermore, many more repeated trials need to be performed on a variety of road conditions, including dirt/rough roads and ice/snow. Also, if the Android application is able to quickly and accurately estimate individual wheel friction coefficients, such as described in [4], these friction coefficients can be used in conjunction with the wheel slips for each wheel to attempt to predict and alert the user about dangerous road conditions. Finally, the implementation of some form of security layer on top of the Bluetooth connection between the smartphone and the vehicular network, is an important future step in ensuring that the application attempts

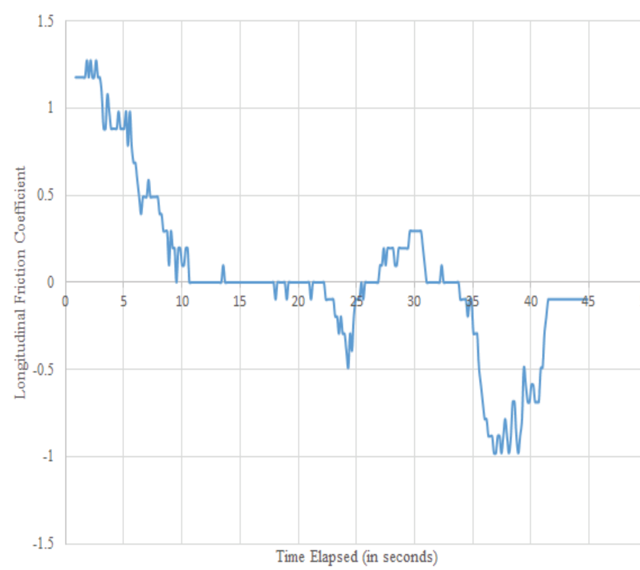

Fig. 3. Dry asphalt driving test.

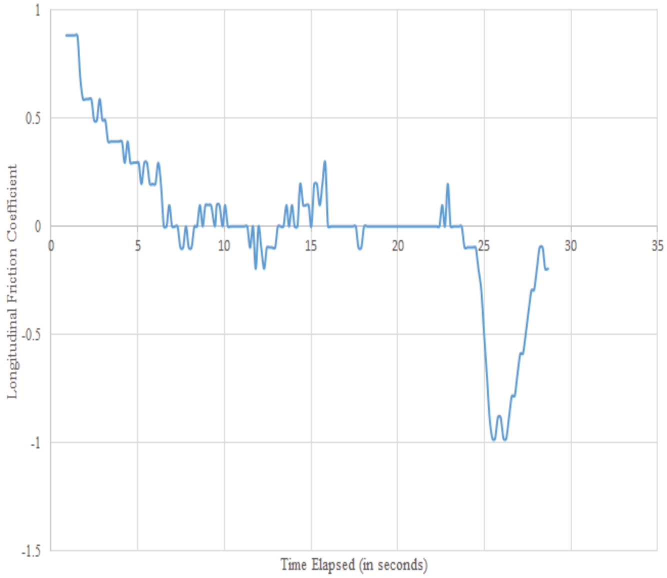

Fig. 4. Dry asphalt driving test.

to handle or mitigate any cybersecurity concerns it might introduce.

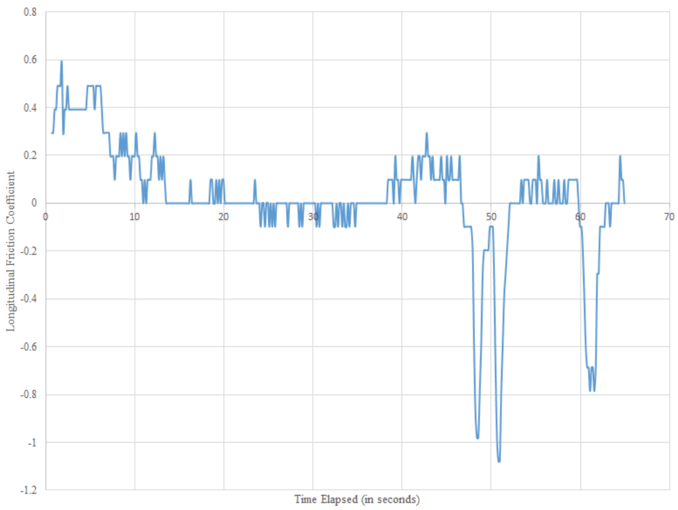

Fig. 5. Wet asphalt driving test.

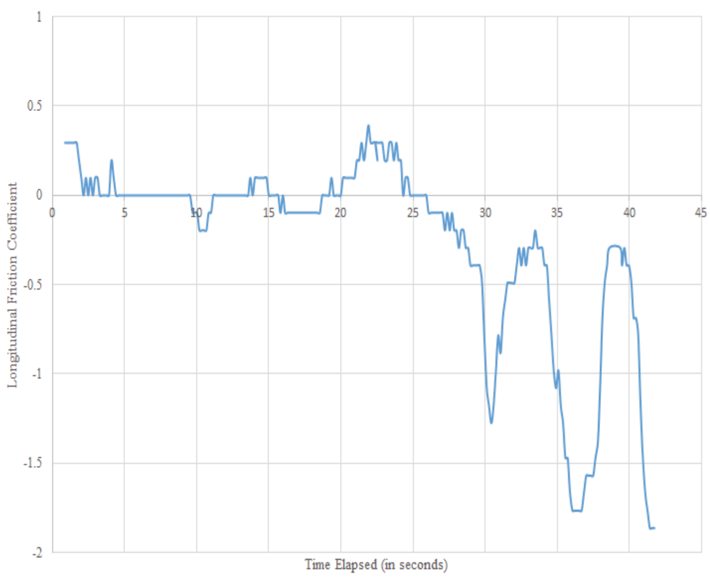

Fig. 6. Wet asphalt driving test.

ACKNOWLEDGMENT

This research was sponsored by the National Science Foundation under grant CNS-1229700. The writers would also like to acknowledge Joshua Siegel of The Massachusetts Institute of Technology for his expertise and assistance with reverse engineering many of the arbitration IDs utilized in this project.

REFERENCES

[1] K. Koscher, et. al., "Experimental Security Analysis of a Modern Automobile," in *IEEE Symp. on Security and Privacy, 2010*, pp. 1-6.

[2] T. Hoppe, et al., "Security threats to automotive CAN networks – practical examples and selection short-term countermeasures," *Rel. Eng. and System Safety*, vol. 96, no. 1, pp. 11-25, 2011.

[3] F. Gustaffson, "Slip-based Tire-road Friction Estimation," *Automatica*, vol. 33, no. 6, pp. 1087–1099, 1997.

[4] R. Rajamani, et al., "Algorithms for Real-Time Estimation of Individual Wheel Tire-Road Friction Coefficients," *IEEE Trans. Mechatron.*, vol.17, pp.1183,1195, Dec. 2012.

[5] J. Villagra, et al., "A diagnosis-based approach for tire-road forces and maximum friction estimation," *Control Eng. Practice*, vol. 19, no. 2, pp. 174-184, Feb. 2011.

[6] J. Wang, et al., "Friction Estimation on Highway Vehicles Using Longitudinal Measurements," *J. of Dynamic Syst., Measurement, and Control*, vol. 126, no. 2, pp. 265–275, Jun. 2004.