Development of a Robotic Driver for Vehicle Dynamometer Testing

Hazim Namik, Tokushu Inamura, and Karl Stol
Department of Mechanical Engineering
The University of Auckland, New Zealand
hhaz001@ec.auckland.ac.nz

Abstract

This paper presents the development of a robotic driver for the automation of dynamometer based vehicle testing. The aim is to successfully follow industry standard test cycles used for emissions testing and produce repeatable results. Success will be measured by not exceeding the limits set for human drivers for a successful test. The design of this robotic driver is unique; it uses a single linear motion to actuate the brake and throttle pedals controlled by two cascaded PID controllers (vehicle speed and pedal actuator position). A full SIMULINK model has been created and used for simultaion and rapid control prototyping using dSPACE. The system has been developed and tested from simulation through to actual dynamometer testing. Test results show that by time-shifting the drive cycle 2 seconds forward, the robotic driver successfully adhered to the ADR 37/01 standard, and produced repeatable results. Further development is needed to improve the performance and overcome current system limitations such as motor saturation and accelerator pedal stick-slip.

1. Introduction

Today's automobile industry places improvement of low energy consumption and emissions as one of the primary areas for progress [Muller *et al.*, 1992]. In context of the Kyoto Protocol, the significance of improvement in these areas is evident. The consequence of more stringent emission testing procedures is that there is increased pressure on vehicle testing facilities. Optimising the productivity of these test facilities is, therefore, important.

Two types of dynamometers are used for automotive testing applications: engine dynamometers and vehicle (or chassis) dynamometers. As the name suggests, the former type measures solely the engine characteristics and requires the engine to be removed from the vehicle to enable testing. Figure 1 shows the latter type, where the driving wheels of the test vehicle are placed on a set of rollers. This method allows the real driving characteristics of the vehicle to be tested while maintaining a static position.

Current methods of vehicle dynamometer emission testing require trained and qualified personnel to undertake prolonged periods of repetitive test cycles. Deviations caused by human error result in inconsistent measurements and require averaging of results over multiple tests. In the case of large deviation exceeding the speed tolerances, results are rendered useless, requiring further tests. Typical speed tolerances are ±3 km/h; this varies depending on the testing standard followed. Both the American Industry/Government Emissions Research (AIGER) and German car manufacturers state the merit of reducing variability emission testing by introducing robotic drivers [Thiel *et al.*, 1998]. The accuracy of a robotic driver ensures the delivery of repeatable results; only single test runs are required, cutting the time of the testing procedure. Simultaneously this liberates the test driver from the menial testing process in a noisy and uncomfortable environment. Thus the time of personnel can be applied to more meaningful and effective work. Such a robot must be able to be retrofitted in the widest possible range of vehicles so that a single robot can perform tests on multiple vehicles consecutively.

To achieve automated vehicle dynamometer testing the following objectives are set:

- Follow chosen standard drive cycles to a degree where it will consistently produce results that do not invalidate the drive cycle standards.
- Produce repeatable results allowing comparison between specific vehicles and models.
- Function on a variety of vehicles.

This paper presents preliminary results to achieve the objectives above.

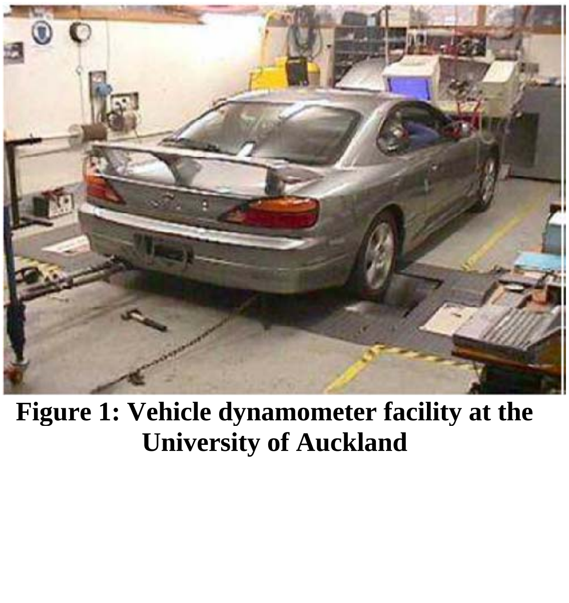

Figure 1: Vehicle dynamometer facility at the University of Auckland

1.1. Existing Robotic Driver Systems

The development of robotic drivers for the specific purpose of dynamometer based testing has been attempted as early as 1974 and reported better repeatability than their best human driver [Gryce, 1974]. Since then, numerous successful attempts have been made.

Various models of robotic drivers for dynamometer testing exist at present. Some commercially available examples are presented below. Commercial robotic drivers used for dynamometer tests include the SAP 2000 by Stahle and ADS-7000 (see Figure 2). Both are used for emissions testing for manual and automatic transmission vehicles without the need to modify the vehicle and achieve high accuracy and performance. The ADS-7000 has a learn feature to characterise the vehicle performance while operating. It has been reported that current commercially available models are complicated and expensive [Romanchik, 2004]. The robot driver presented in this paper attempts to achieve autonomous testing by a simpler and more cost effective means. Simultaneously, this system will provide a platform for future development of autonomous vehicle testing.

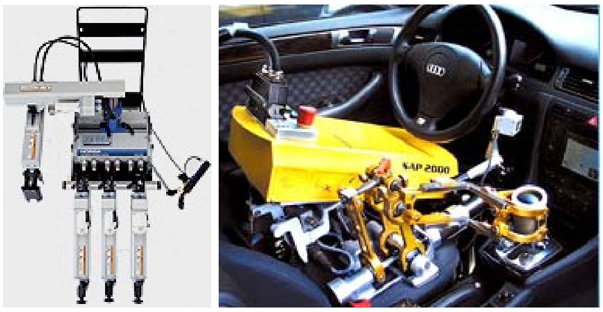

Figure 2: ADS 7000 (left) Stale SAP 2000(Right) [Romanchik, 2004]

Thorough comparisons between robotic drivers and human drivers have been conducted in literature. It has been found that robotic drivers have a tendency to produce 10% to 20% higher results in emission tests than a human driver [Thiel, 1998] .This can be attributed to differences in driving style. Human drivers tend to reduce the variation in acceleration by looking ahead of the current speed, smoother transitions between speeds result. This assists the comfort of the passengers. A robotic driver tends to closely follow the exact speed of the drive cycle resulting in more frequent acceleration changes. It can therefore be said that repeatability and comparability of results can be assured, but does not accurately indicate the performance of a human driver. Comparisons to human driver performance are outside the scope of this paper. The primary focus is to develop a robotic driver that reduces variability of test results such that cross car comparisons can be made with higher efficiency.

Several control approaches to have been considered by researchers developing robotic drivers. These include a robust controller and a learning controller. A brief description of each controller is discussed below.

The robust controller developed by Muller *et al.* [1992] uses an h_∞ feed forward approach which is good for systems with uncertainties. Being robust does not require parameter tuning to account for system variations. The controller focuses on minimising accelerator pedal usage to improve fuel efficiency; however, it was found that a single variable control was not adequate for successful implementation. Therefore, a multivariable controller was developed to control the vehicle and engine speeds by using the clutch and the accelerator pedals simultaneously. This kind of multivariable control can not be applied for this project since the robotic driver used for this project is designed for vehicles with automatic transmission. However, a feed-forward approach might help improve the robot's performance as suggested by Moriyama *et al.* [1991]. It was found that a minimum of one second of information should be given ahead to have acceptable fuel economy, meaning a smoother velocity profile.

The idea behind a learning controller is to mimic how human drivers learn the vehicle properties before making the actual test [Moriyama *et al.*, 1991]. The robot performs a series of tests to determine the power-speed relationship of the vehicle, engine torque with accelerator pedal position, engine braking[1] relationship with engine speed, and braking force with brake pedal position. Once these relationships are known, they are tabulated and stored in memory.

The speed controller cycle starts by calculating the required power to reach the desired speed. It is followed by calculating the amount of engine braking in the system at that moment in order to determine which pedal to actuate [Moriyama *et al.*, 1991]. The controller then retrieves the required pedal position from the lookup tables generated during the learning phase. There is no distinct speed controller as such in this design due to the learning process which means that there are no parameters to tune for different vehicles.

From the above reviewed controllers, it was found that none of them used standard Proportional-Integral-Derivative (PID) controllers and the authors did not mention why they were not considered. Therefore, our approach is to use simple PI or PID speed controllers.

The outline of the remainder of this paper is as follows: Section 2 presents the design of the system and its components. Section 3 explains the modeling and simulation developed using SIMULINK. The hardware implementation and its results are presented in section 4. Section 5 contains discussion of the results. Finally the conclusions drawn from this project thus far are shown in Section 6.

2. System Design

2.1. Overview

The system consists of three major components apart from the vehicle itself. These are the vehicle dynamometer facility

[1] Engine braking is the power absorbed in compressing and hence heating the air inside the cylinders of the engine

with its equipment, the robotic driver to drive the pedals, and the controlling computer (see Figure 3). The computer contains the data for the test cycle to be run.

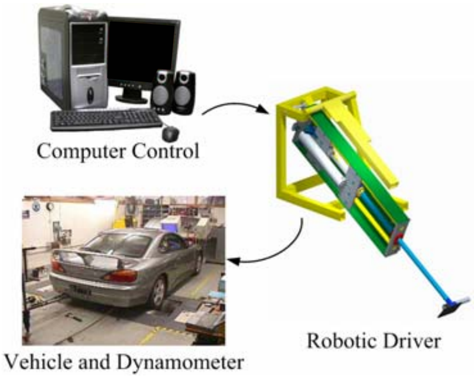

Figure 3: General system layout

The controller block diagram is shown in Figure 4, where two cascade controllers are used: the pedal actuator position controller and vehicle speed controller. The speed controller sends position commands, based on the current speed error, to the position controller. These position commands represent the desired pedal positions.

The position controller controls the position of the robot's foot that actuates either the brake or accelerator pedal. Based on the desired position from the speed controller and position feedback from the encoder, the controller changes the voltage applied to the DC motor by changing the polarity and duty cycle of the Pulse Width Modulated (PWM) signal being sent to the driver circuit. The driver circuit amplifies the control signal, provides sufficient current to drive the motor, and isolates the computer from the motor for protection from induced voltage spikes by the motor.

2.2. Robotic Driver

The original design for this robotic driver was developed by the Grand Challenge New Zealand team to compete in the DARPA Grand Challenge; an autonomous vehicle race based in the USA. This robot platform has been adapted for dynamometer testing purposes.

As shown in Figure 5, the robot is restrained to the driver seat using a standard seat belt. The steering actuator shown is not required for dynamometer testing applications as only vehicle speed control is required. However it serves the function of an extra contact point to constrain the position of the robot frame.

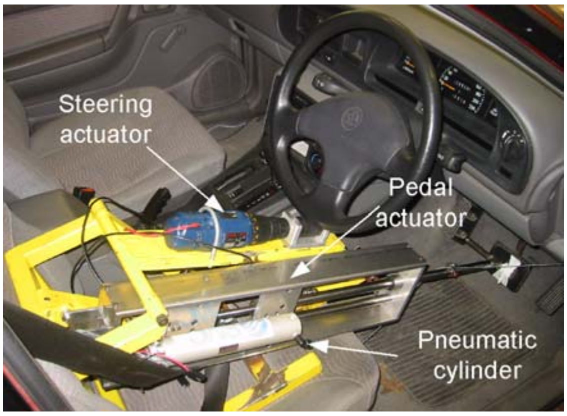

Figure 5: Robot installed in test vehicle

The pedal actuator uses a single ball screw drive powered by a 12V DC motor to actuate both pedals. This is achieved by directly pushing the brake pedal and pulling on the accelerator pedal above its pivot point (see Figure 6). Therefore, a single linear motion controls both pedals. This unique design eliminates issues with timing between the engagement and release of the pedals and the need for multi-variable control. A quadrature encoder is used to give position and directional feedback for the controller. A pneumatic emergency stopping system is implemented to ensure safe operation. The pneumatic cylinder is visible in Figure 5. Upon deployment, the cylinder will force the pedal actuator into full brake.

Preliminary testing on test vehicles has shown that the robotic driver actuates both pedals successfully. The time taken from full throttle to full brake is approximately one second. This defines the maximum rate of deceleration of the vehicle.

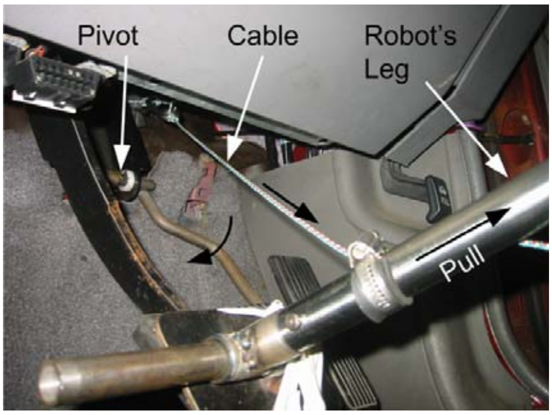

Figure 6: Pedal mechanism operation

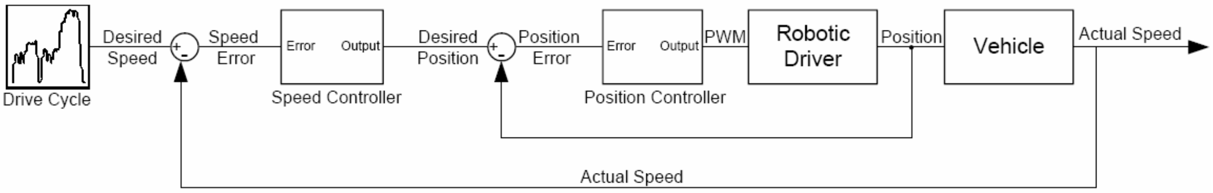

Figure 4: Robotic driver control structure.

2.3. Controller Hardware Selection

There are many possible options for controller hardware and software to achieve the control design discussed in section 2.1. A wide range of hardware/software options were considered including: microcontroller/C language based control, FGPA/Labview Real Time control, Linux based computer control and dSPACE/Control Desk. From this the control hardware/software was selected through a formal process. Valuation and weighting matrices were created to select the optimal solution. Several categories were considered in the scoring process such as available I/O, communications, development time, price, real-time capabilities, etc. The selected controller hardware was the dSPACE DS1104 R&D Controller Board because of its real-time capability, short estimated development time, ability to develop graphical user interfaces (GUI) easily and the high chance for reuse in other research applications.

2.4. Interfacing

A schematic block diagram for the system setup is shown in Figure 7. All control calculations are conducted in real-time on the dSPACE card. The card interfaces with the development GUI designed for this project to control the drive cycle test procedure as well as recording data for later analysis.

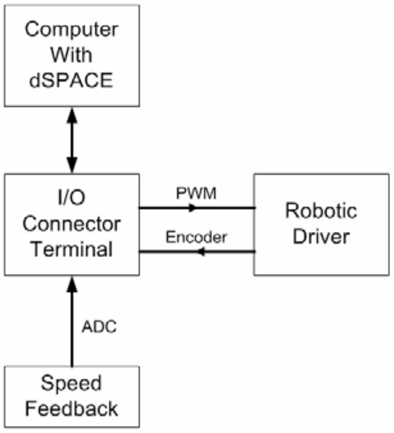

Figure 7: System setup block diagram

The dSPACE card connects to a dedicated I/O board which interfaces the computer with the physical system. The digital PWM signal sent through the I/O board is amplified to power the 12V DC motor of the pedal actuator. The feedback from this is realised through the rotational quadrature encoder that is connected to a dedicated channel on the I/O board. The velocity feedback from the vehicle is also interfaced with the I/O board.

3. Modelling and Simulation

Modelling the system using MATLAB/SIMULINK is necessary to verify the controller design and investigate different approaches to the problem without the necessary time and expense of testing with actual hardware. Furthermore, the selected control hardware uses SIMULINK models to generate real-time C code for rapid control prototyping. Consistent with Figure 4, the physical system is divided into two major models; the robotic driver model and the vehicle model.

3.1. Robotic Driver Model

The robotic driver model includes the following systems: DC motor dynamics and ball-screw dynamics as shown in Figure 8. A static friction model is included in the ball-screw system because of the observed stick-slip characteristics of the real robot.

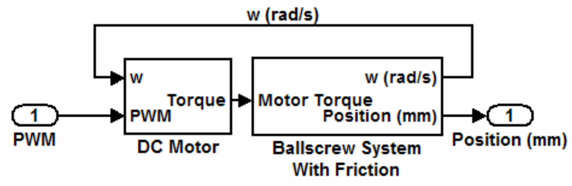

Figure 8: Robotic driver subsystems

The 12V DC motor is modelled by a first order system that takes the coil resistance and inductance into account. These values are extracted from the motor's datasheet. The motor transfer function – ratio of output current to input net voltage (applied voltage less the back electromagnetic force (emf)) – is shown in Equation (1) where *L* is the coil inductance and *R* is the coil resistance. Motor saturation was taken into account in the form of limiting the current to the listed saturation current by the manufacturer of ±6A. The motor torque is proportional to the current by a torque constant which is also listed in the motor datasheet.

$$\frac{I(s)}{V(s)} = \frac{1}{L \cdot s + R} \tag{1}$$

The ball-screw drive system model calculates the linear position of the robot foot by finding the net torque on the ball-screw and hence the angular acceleration. Then using kinematics as well as the screw pitch, the linear velocity and position are found by integrating the acceleration and velocity respectively. However, to get the net torque the two main loads applied to the ball-screw must be calculated; these are the pedals resistance and the friction in the system.

The resistance force that the pedals apply on the robot's foot, collected from experimental data, is shown in Figure 9. It can be seen that the relationship is nonlinear with pedal position for the braking pedal. This nonlinearity is due to brake assist found on most modern cars.

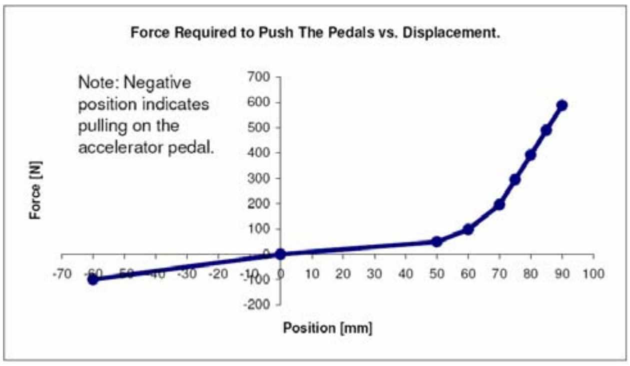

Figure 9: Axial resistance force from the pedals

The friction in the linear actuator is modelled by a simple static model that takes coulomb and viscous friction into account [Olsson *et al.*, 1998]. The limits for static and dynamic frictions are found experimentally. The experiment was carried out by slowly increasing the applied current to the motor until the foot starts to move; this gives the breakaway torque. Once the foot is moving, the current is reduced until it stops again; this indicates the minimum friction torque in the system. To verify the static friction model, a simulation of a simple sliding block with applied sinusoidal force was carried out. The results, shown in Figure 10, agree with the simulations done by Olsson *et al.* [1998].

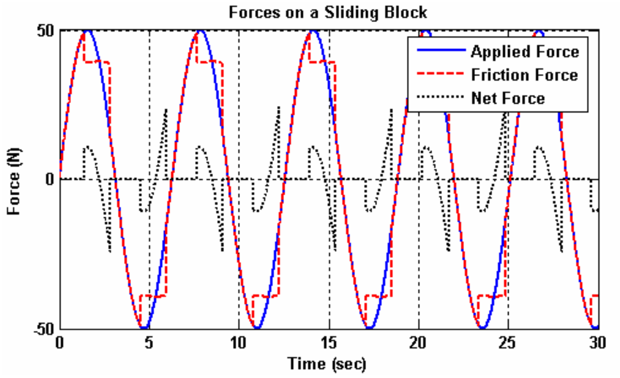

Figure 10: Sliding block simulation results with static friction model

3.2. Vehicle Model

The vehicle model is divided into several smaller subsystems; these include an engine map, automatic transmission, power transmission, braking system, and external forces such as drag and rolling resistance. These subsystems were modelled to be as simple as possible for the purpose of preliminary testing.

The engine map shown in Figure 11, taken from a Mazda 3 (MZR 2.0L) [Chang Tu, 2004], calculates generated engine torque given the throttle position and the current engine speed in revolutions per minute (rpm). The engine map data includes engine braking since the net torque was measured from a real engine output shaft [Chang Tu, 2004].

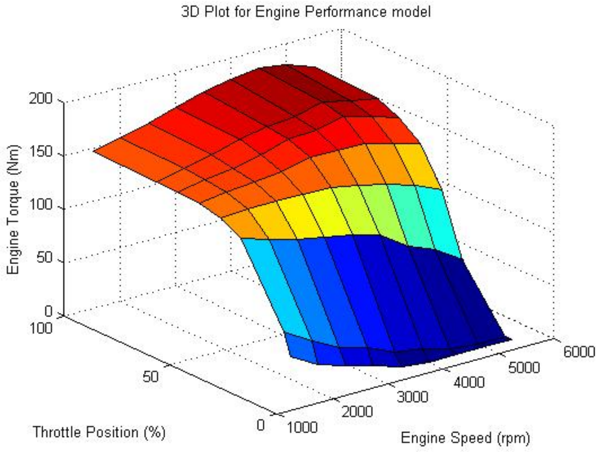

Figure 11: Engine map (adapted from [Chang Tu, 2004])

The engine torque is then transmitted thought the current transmission gear ratio, the differential, and the wheels to give the linear force on the car. The vehicle is modelled using a single degree-of-freedom, which assumes the engine and wheels are rigidly coupled. The net force on the vehicle is found by subtracting the applied external forces that include rolling resistance and aerodynamic drag. Integration then provides the vehicle speed as well as the engine rpm.

A simple gear changing algorithm was developed to account for the change in torque transmission ratio in a real-world situation. The automatic transmission was modelled with 4 gears with each gear having a velocity range; for example, 2nd gear has a range starting from 20km/h to 40km/h. The transmission ratios were loosely based on an average car [Nice, 2006]; the values were modified to give a suitable acceleration profile for an average car which takes approximately 12 seconds to reach 100km/h from rest.

Figure 12 shows the modelled relationship between applied braking force on the vehicle and the brake pedal displacement. The relationship is derived experimentally by Moriyama *et al.* [1991] and calibrated to match the average braking performance. To calibrate the braking force, a vehicle was simulated to fully apply the brakes from 120km/h by measuring the stopping distance. The relationship in Figure 12 was derived when the stopping distance matched that of an average car of approximately 90m [Safe Drive Training, 2004].

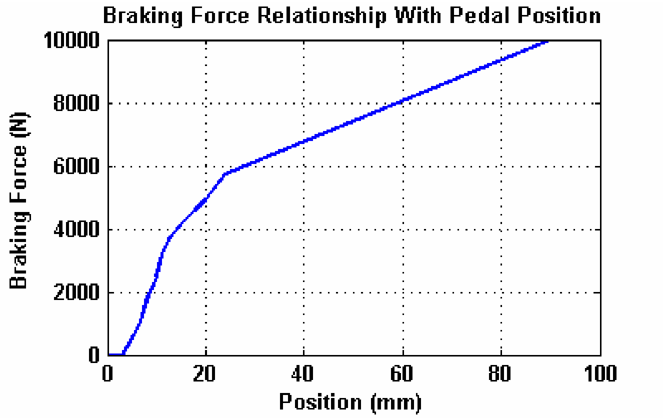

Figure 12: Applied braking force vs. brake pedal position

3.3. Controller Design

The designed speed controller is a classical PI controller; the reasons for such a controller include simplicity, and sensitivity to feedback noise is less than a controller with derivative term. The selected position controller is a classical PID controller due to fast system dynamics and little or no noise from encoder feedback.

The controllers are tuned using the Ziegler and Nichols Ultimate Cycle Method (UCM) to minimise the error [Bolton, 2003]. This method gives rough values for controller gains and may require fine tuning later. Such was the case for both controllers when the UCM gave large integral gain for the controllers. The simulations show that the controller performance is excellent except when step inputs are used; the integral gain is reduced to stabilise the system. Integrator saturation was included to prevent integrator windup and consequential performance degradation with changing set points.

3.4. Simulation Results

The complete system model is simulated for the IM240 drive cycle which is an abridged version of the standard Australian drive cycle used for emissions testing [ADR 37/01, 1995]. To pass the drive cycle the vehicle speed must lie within ±3.2km/h of the desired speed and/or the speed must be within ±1 second of the desired speed according to the Australian Design Rule (ADR) 37/01 as shown in Figure 13. However, the vehicle speed is allowed to exceed these limits for no longer than 2 seconds at a time.

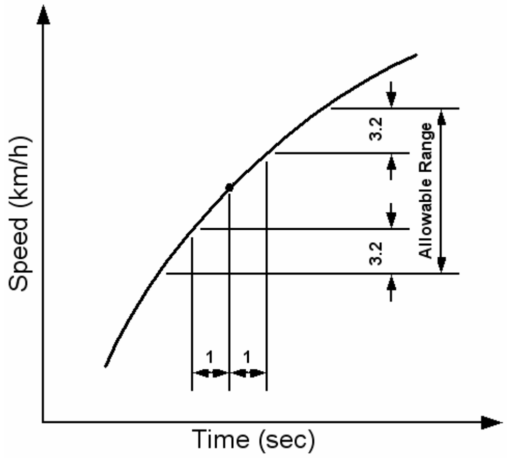

Figure 14: ADR drive cycle speed tolerances (adapted from [ADR 37/01, 1995])

The simulated response is shown in Figure 14. From the first graph, it can be seen that the vehicle speed is well within these limits. In fact, the maximum speed error is under 0.5km/hr which means that the choice of the speed controller was adequate. However, the vehicle model is very simple and it is primarily used to give an indication of the feasibility of using the selected controller.

The position controller performance is similar to that of the speed controller. The second graph of **Error! Reference source not found.** shows the simulated pedal position almost exactly on top of the desired pedal position set by the speed controller. The maximum error for the position controller is less than 0.4mm, which is less than the allowable maximum error of ±1mm. The modelled friction seems to have an effect when the set point is fixed; this is where stick-slip is observed where the system oscillates around the set point driven by the integral term of the controller.

4. Hardware Implementation

Implementing the designed system on real hardware was achieved in incremental steps to verify and correct, if necessary, the performance of each system. The approach started with interfacing the hardware and the computer program, then position feedback was calibrated, followed by simple pedal position control. Hardware-in-the-loop (HIL) testing followed with the simulated vehicle running in real-time with the actual robotic driver. The final step was to implement the system using a real vehicle on the chassis dynamometer.

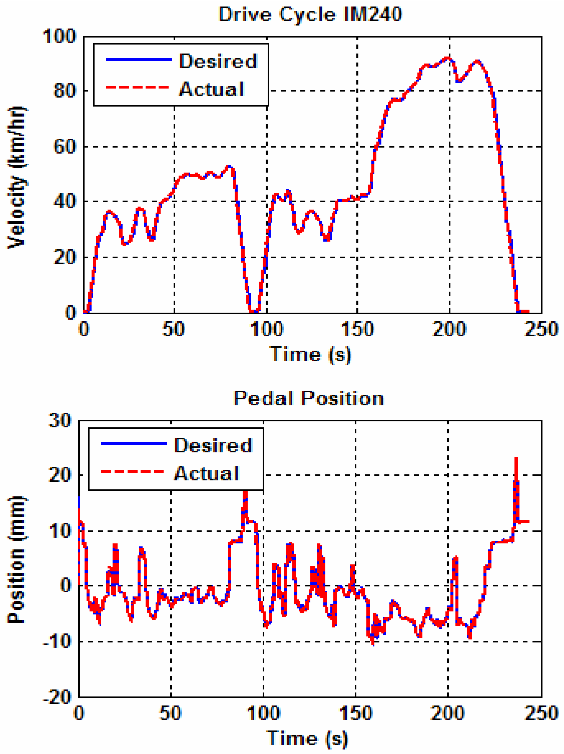

Figure 13: Simulated drive cycle

Hardware and software interface was carried out successfully and tested by applying voltage to the motor through the GUI to move the pedal position in either direction. Position encoder calibration was achieved by analytical means using the gear ratios, encoder counts per revolution, and ball-screw pitch. The encoder was verified by driving the robot foot through different positions and comparing the encoder output with the actual distance measured using a digital calliper accurate to 0.01mm. Results show that the encoder has an average error of 0.04mm with maximum error of 0.1mm; therefore, the encoder has good accuracy and repeatability for its application. Using the encoder feedback, it is established that there is little backlash in the system; the backlash is in the order of 0.1mm.

With the encoder maximum error known, the position controller has a dead band of 0.1mm to reduce jitter and oscillation around the set point due to stick-slip. The dead band resets the integrator term to switch off the motor, preventing it from rapidly overheating when the pedal position error is inside that band. This can be done safely since the ball-screw system cannot be back-driven. Testing the position controller by giving it step, ramp, and sinusoidal inputs shows that the parameters derived for the controller using simulation are adequate. However, the position controller was tuned on the real hardware using the UCM giving parameters close to those previously derived from simulation. The controller performance is consistent with the different input test signals giving a maximum error of 0.4mm.

Hardware-in-the-loop simulation, shown in Figure 15, was achieved by inserting the developed vehicle model in the real-time SIMULINK model. The speed controller takes the simulated vehicle speed from the vehicle model (based on position feedback) and gives a desired pedal position. The position controller then sends voltage signals in the form of a PWM signal to the actual robotic driver. Performance results are shown in Figure 16. The figure shows accurate speed control with a maximum error of 1km/hr. The position controller is consistent with its performance of a maximum error of 0.4mm.

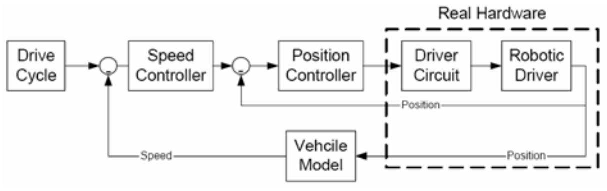

Figure 15: HIL implementation

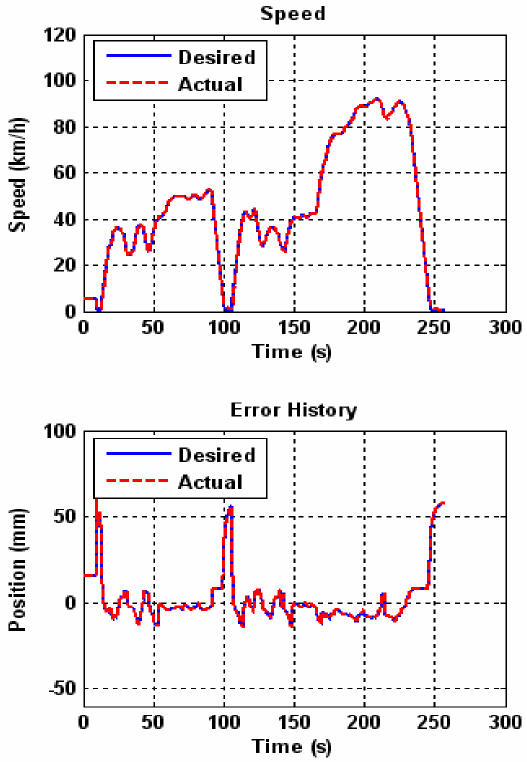

Figure 16: HIL simulation results

During the setup stage for actual dynamometer testing, it was found that the speed feedback signal is extremely noisy (see Figure 17). A 1st order passive low pass filter with cut-off frequency of 1Hz was applied to the signal with the results shown in the figure; the signal is then passed through a software filter of the same cut-off frequency to eliminate transmission noise. The feedback is then calibrated to give the measured speed in km/h given the input voltage. However, the filtering process delays a signal by a certain phase lag according to its frequency; this can be overcome by feeding the drive cycle ahead to the controller by the same amount of time.

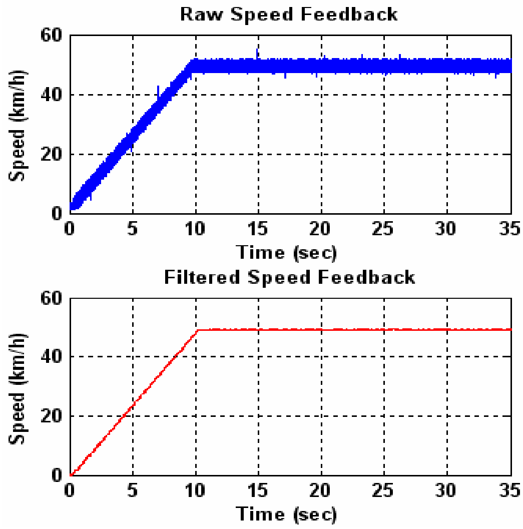

Figure 17: Speed feedback signal before and after filtering

5. Dynamometer Testing Results

Initial testing on the vehicle dynamometer exposed significant friction in accelerator pedal. The resulting stick-slip characteristics reduced the accelerator pedal resolution and amplified the nonlinearity during the changeover from braking to acceleration. This large nonlinearity was observed to have two effects on the system:

1. Steady-state error during ramps in desired speed; a derivative term was added to improve performance.
2. System performance such as damping is insensitive to different controller parameters, as shown in Figure 18.

Another observation made was the saturation of the pedal actuator before fully engaging the brake; this affects the maximum deceleration rate of the vehicle during testing.

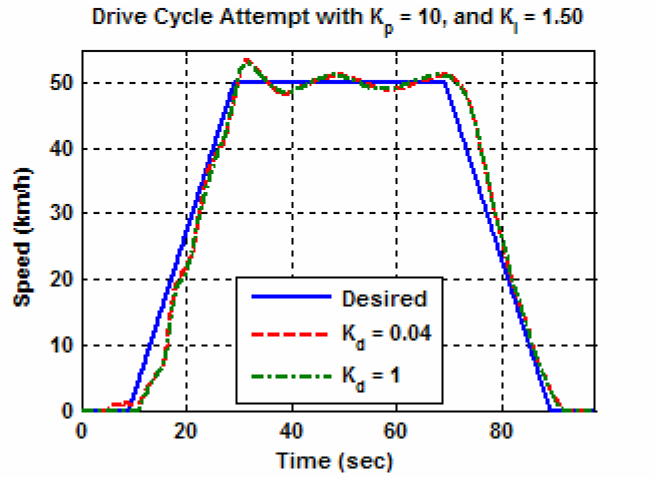

Figure 18: Independence of system performance from controller parameters

One of the main objectives was to achieve repeatable results. Therefore, two drive cycle tests were carried out consecutively using the same controller parameters; tests results are shown in Figure 19. The two tests are almost identical indicating high repeatability; however, at time of 100s there is a noticeable difference. This could be caused by: deteriorating motor performance with increasing coil temperature, stick-slip of the throttle pedal, or changing vehicle performance due to increasing engine temperature. Further testing is required to determine the true repeatability of the system and the possible source of this variability. Nevertheless, the system produced repeatable results on the same vehicle and hence will reduce the number of tests required per vehicle. Since no tests were done using different vehicles, repeatable results allowing for cross comparisons between vehicle models cannot be inferred.

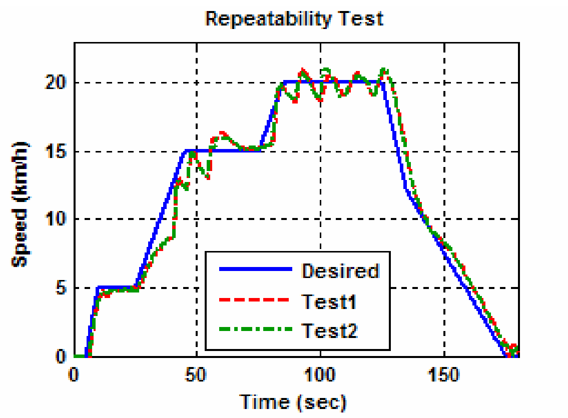

Figure 19: Repeatability test results

Due to the above controller performance, high speed testing (above 80km/h) was deemed to be unsafe. Furthermore, the dynamometer is limited to operations below 100km/h. Therefore, testing the on IM240 drive cycle was limited to the first 95 seconds where the speeds never exceed 60km/h. The best performance, shown in Figure 20, was obtained by time-shifting the drive cycle 2 seconds forward. There are three instances where successful test limits are exceeded: at 5.5, 40.6, and 92.75 seconds. In all these instances, the actual speed was outside the limits for less than 2 seconds, which means that the drive cycle successfully adhered to the ADR standard.

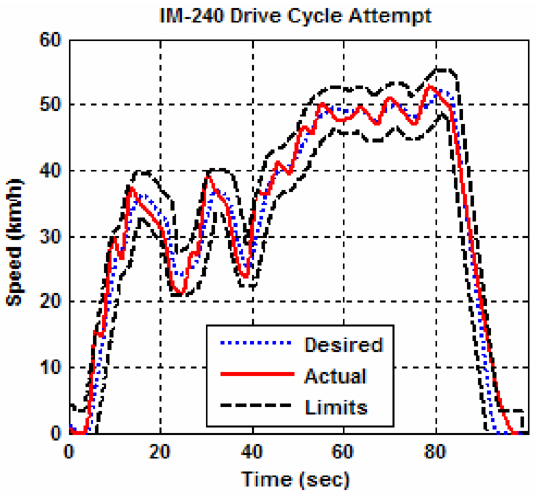

Figure 20: Drive cycle performance with 2 sec time-shift

The last deceleration phase in Figure 2 where the speed deviates from the desired, is due to the saturation of the pedal actuator, preventing it from pressing the brake pedal fully.

6. Discussion

The results show that the developed robotic driver model with static friction is a good representative model for the current system used. This can be verified by the controller error results from the simulation and hardware implementation where the error in both cases has a maximum value of 0.4mm. The position controller performance is excellent and well within the allowable range of ±1mm.

The vehicle model was shown to be a poor representation of the real vehicle; this was expected since the model constructed was very simple. The main difference between the model and the actual system was the failure to predict the nonlinearity in the change from braking to throttle region. However, it was a valuable tool in the development of the controllers for actual implementation.

7. Conclusions

- The robotic driver successfully adhered to the Australian Design Rules for emissions testing in following a standard drive cycle using PID controllers; the drive cycle was time-shifted 2 seconds forward to compensate for delays in the system.
- The robotic drive is capable of producing repeatable results, although further testing is required.
- The robotic driver has been developed to perform vehicle dynamometer testing meeting its design specifications.
- Problems with pedal stick-slip, large nonlinearities, and motor saturation limit the performance of the developed controllers to narrowly passing the tested drive cycle.
- A full SIMULINK model is developed consisting of the robotic driver and vehicle models. The robotic driver model is a good approximation of the real system while the simple vehicle model is not.

8. Future Development

The main focus for future work is to eliminate or mitigate the effects of the discussed limitations by either selecting a new controller design, such as the one described by Moriyama et al. [1991], or building on the developed controllers.

Using the accuracy, good pedal actuator performance and the data logging capability of the system, a vehicle model can be developed using system identification techniques. The use of other dynamometer sensors such as absorbed power may help to develop a better vehicle model.

A simple fuzzy PID controller can be developed to limit the use of the brakes and avoid the nonlinear region. If the speed exceeds the desired, then limit the controller output to throttle regions only; allowing the car to slowly decelerate under its own friction and external forces. However, if the error is big or it is changing rapidly, allow the use of the brakes. This controller will require bumpless transfer between those states.

The next step for the developed system is to use small embedded controllers. This solution will make the system smaller, easier to setup, and affordable for use in commercial testing.

9. Acknowledgements

The authors would like to thank the Grand Challenge New Zealand team for collaborative development of the robotic driver. We would also like to thank Stephen Elder and Mr. Nigel Boielle from the Energy and Fuel Research Unit at the University of Auckland for dynamometer facility support.

References

[ADR 37/01, 1995] ADR 37/01 Emission Control for Light Vehicles (1995) Motor Vehicle Standards Act (Section 7). Australian Design Rule 37/01. Act of parliament, Australia. Department of Transport and Regional Services.

[Bolton, 2003] Bolton W. (2003) *Mechatronics: Electronic Control Systems in Mechanical and Electrical Engineering.* Third edition. Pearson Education Limited, Harlow. pp 300-301.

[Chang Tu, 2004] Chang Tu, M.. (2004). Development of a Real-Time Cotnroller for an Engine Dynamometer. Mechanical Engineering Project, School of Engineering, The University of Auckland.

[Gryce, 1974] Gryce, R.T.(1974) Ford auto/emission driver system. Proc. International Automobile Engineering and Manufacturing Meeting (Toronto), Oct 21-25.

[Moriyama *et al.*, 1991] Moriyama, A., Murase, I., Shimozono, A., and Takeuchi, T. (1991) A robotic driver on roller dynamometer with vehicle performance self learning algorithm. Proc. International Congress and Exposition (Detroit), February 25 – March 1.

[Muller *et al.*, 1992] Muller, K., and Leonhard, W. (1992) Computer Control of a Robotic Driver for Emission Tests. Proc. International conference on Industrial Electronics, Control, Instrumentation, and Automation (San Diego), Nov 9-13. pp 1506 - 1511 vol.3.

[Nice, 2006] Nice, K. *How Automatic Transmissions Work*. Retrieved July 27, 2006 from: http://auto.howstuffworks.com/automatic-transmission.htm

[Olsson *et al.*, 1998] Olsson, H., Astrom K., Canudas de Wit, C., Gafvert, M., and Lischinsky, P. (1998) Friction Models and Friction Comensation. European Journal of Control, **4** (3), 176-195.

[Romanchik, 2004] Romanchik D. *Robot Drivers Take The Drudgery Out of Testing*. Retrieved March 5 2006 from : http://www.reed-electronics.com/tmworld/article/ CA377207.html?industryid=21385& text=robot+driver

[Safe Drive Training, 2004] Safe Drive training (2004), *Stopping distance.* Retrieved August 17 2006 from: http://www.sdt.com.au/STOPPINGDISTANCE.htm

[Thiel *et al.*, 1998] Thiel, W., Grof, S., Hohenberg,G. and Lenzen,B. (1998) Investigations on Robot Drivers for Vehicle Exhaust Emission Measurements in Comparison to the Driving Strategies of Human Drivers. Proc. SAE International Fall Fuels and Lubricants Meeting and Exposition (San Francisco), Oct 19-22.