

# Automated tuning of an engine management unit for an automotive engine

C Hardie, H Tait, S Craig, J G Chase\*, B W Smith and G Harris

Department of Mechanical Engineering, University of Canterbury, Christchurch, New Zealand

**Abstract:** Modern automotive engines are digitally controlled using an engine management unit (EMU) that is typically manually programmed using an engine dynamometer to obtain desired levels of power, emissions and efficiency. Closed-loop control of an engine dynamometer and EMU, combined with an overall engine tuning algorithm, is used to automate the tuning of the engine map for a four-cylinder engine. The tuning algorithm determines the air-to-fuel ratio necessary for each region of engine speed and throttle position to obtain the desired performance, automatically moving to each operating region in the map. Preliminary automated tuning results produce power output curves comparable with those delivered using the original manufacturer tuned EMU. At lower engine speeds data filtering is required and results in power outputs slightly lower than the factory-tuned engine. At higher speeds small improvements in engine efficiency, for equivalent performance, can be found. The research presented clearly demonstrates that engine tuning to a very high standard, equivalent to original equipment manufacturer engine performance, can be successfully automated, saving time and adding consistency to the engine tuning process.

**Keywords:** control, automation, engine tuning, dynamometer, engine management, engine map

## 1 INTRODUCTION

Engine management units (EMUs) are digital processors that actively control fuel flow and ignition timing to ensure optimal engine operation and are employed in most modern fuel injected engines. Engine tuning plays a vital part in ensuring that the EMU is programmed so that the engine performs well under a variety of conditions. The manufacturer-developed EMU parameters are typically developed to meet design compromises between performance, market and legislative specifications, particularly with regard to emissions. After-market EMUs are available to replace a damaged unit, to achieve specifications different to those used by the manufacturer, to enhance the operation of an engine intended for use both on land and in a marine environment or to enable the tuning and enhanced performance of a modified engine.

The air–fuel ( $A/F$ ) ratio is important in the combustion and tuning processes. If there is too much fuel, not all the fuel is burnt, causing high fuel consumption and

increased emissions of hydrocarbons and CO. Too little fuel can result in engine damage such as burnt exhaust valves. The amount of fuel supplied is controlled by the EMU, which controls the frequency and pulse width of the fuel injectors. At steady state, constant engine speed the pulse width is calculated from a combination of three inputs: ambient air temperature, the inlet manifold air pressure (MAP) and user-defined volumetric efficiency correction factors (VECFs).

A typical engine has a two-dimensional plane of steady state operating points with engine speed along the horizontal axis and throttle position or MAP along the vertical axis, as shown in Fig. 1. Each of these points has differing requirements for fuel and ignition timing. Most EMUs divide this plane up into finite rectangular elements, commonly referred to as ‘zones’, that make up an ‘engine map’ and where each zone has an associated user-defined VECF. The engine map acts as a look-up table for operating parameters used by the EMU to control engine operation.

Currently, engine tuning is typically performed by a skilled technician who manually adjusts the operating parameters for each zone of the engine map on a dynamometer or while driving. The engine is tuned to the desired specification for each zone in which the engine is expected to be operated. This process is iterative and

*The MS was received on 3 January 2002 and was accepted after revision for publication on 6 August 2002.*

*\* Corresponding author: Department of Mechanical Engineering, University of Canterbury, Private Bag 4800, Christchurch, New Zealand. email: g.chase@mech.canterbury.ac.nz*

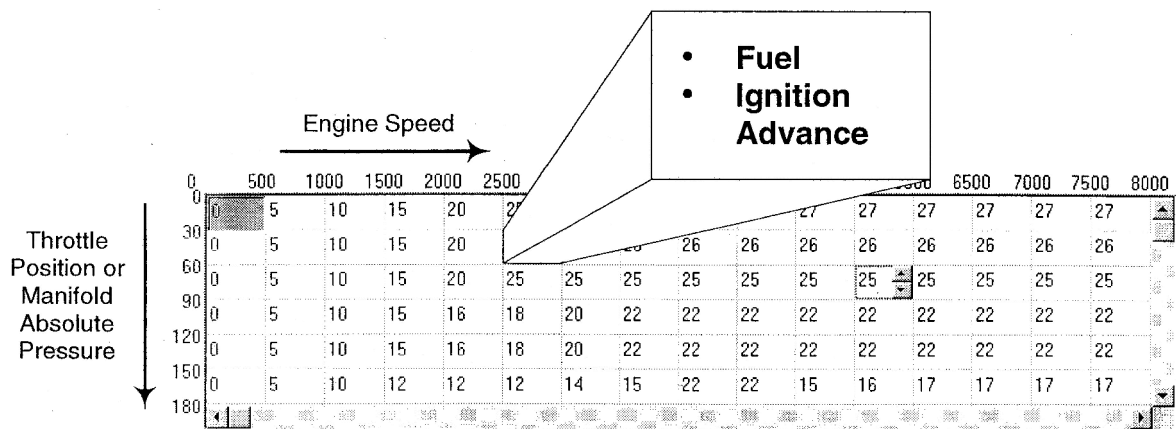


Fig. 1 Structure of the engine map

time consuming, creating the potential for significant time savings, additional repeatability and more accurate results via automation.

The primary goal of this research is the automation of the engine tuning process via engine tuning software that communicates with both the dynamometer controller and the EMU to create a closed-loop engine tuning system. The engine tuning algorithm that governs the process is designed to optimize automatically the engine control parameters for each operating zone, to achieve performance in accordance with user-defined specifications. In addition, the software ensures that potentially damaging situations are detected and responds by placing the engine in a non-damaging state, sounding an alarm and pausing the process until the problem can be corrected. This second requirement ensures that the process can be run largely unsupervised. Finally, the automation process, and the specific software developed, adds both speed and consistency to the engine tuning process.

The after-market EMU employed is developed by Link Electro-Systems of Christchurch, New Zealand. This EMU along with the PCLink control and communication software is used to control the engine. A servo-controlled Froude-Consine hydraulic dynamometer is employed for the tuning process. The automated tuning algorithm comprises software designed to interface with, and control, these two units for the end purpose of tuning the Link EMU parameters for every usable zone of engine map.

Engine tuning is a well-known art documented in a variety of texts on the topic of internal combustion engine performance [1]. Prior research in the area of automated engine tuning is scarce with most research in the area of automated engine tuning focusing on modifying specific engine functions, such as spark ignition, to enhance performance while the vehicle is in operation [2, 3]. Other works focus on systems to adaptively eliminate undesirable operating conditions or effects such as knock [4]. All of these efforts focus on applying auto-

mation to enhance engine performance, rather than automating EMU controller design. Engine tuning to meet specific performance criteria is typically performed largely by hand via a combination of experience, science and art. No references were found specifically addressing automation of the engine tuning process, although there is some prior research that generally references the possibility of performing this task [5].

The following sections address the test system and software developed to enable the automated tuning of an EMU using an engine dynamometer and the basic tuning algorithms developed. Subsequent sections test one of these tuning methods, compare the performance obtained with that provided by the manufacturer's EMU and summarize the results and conclusions.

## 2 AUTOMATED ENGINE TUNING TEST SYSTEM AND SOFTWARE

The experimental automated engine tuning system is tested on a 1992 Toyota 3S-GE engine from a Toyota MR2. This four-cylinder engine has two overhead camshafts, displaces 1998 cm<sup>3</sup> and generates manufacturer-listed peak outputs of 121 kW at 6800 r/min and 191 N m at 4800 r/min. Load is applied to the engine via a servo-controlled Froude-Consine hydraulic dynamometer that is controlled by a Test Automation TA2000 controller interfaced with a PC-based data acquisition system utilizing an Advantech PCL-818 analogue-to-digital converter (ADC) card. The test bed is set up so that either the original Toyota EMU or the Link EMU can control the engine. The dynamometer and Link EMU are independently controlled via a PC program to close the loop on the engine tuning process.

Specifically, the engine dynamometer and controller set and deliver values for engine speed, torque and throttle position. The Link EMU can deliver values for engine speed, throttle position, engine temperature, air temperature, the injector duty cycle, ignition advance,

the oxygen sensor and the entire engine map. Where there is overlap, such as engine speed, the control algorithm uses these values to determine when transients have stabilized, as both units will report the same value.

The automation of the engine tuning process is achieved via specific software written to communicate with and control independently both the engine dynamometer and the Link EMU. An overall tuning algorithm steps through each zone in the engine map and tunes the EMU parameters for each zone according to user specifications. User specifications that set the target(s) for the tuning process in each zone are set prior to tuning using a graphical user interface (GUI). A different GUI is used to provide the user with real-time information about the engine operation, displaying parameters such as desired engine speed, actual engine speed, torque, power, fuel flowrate and ignition advance.

The software is developed using Borland C++ Builder version 4.0. The overall closed-loop system architecture and communication links created are illustrated in Fig. 2, where the broken line encloses the software communication links created to close the loop on the tuning process. Figures 3 and 4 show the status window and tuning set-up window for the user interface.

The specific software structure consists of three primary C++ classes. Two classes handle commands for the EMU and dynamometer. A third class of low-level control functions enables closed-loop engine tuning, facilitating control of the separate dynamometer and EMU systems by a single global entity on the PC. More specifically, these functions enable 'get' and 'set' commands for quantities including fuel, master level, throttle position, torque, power, ignition advance,  $\lambda$  sensor, knock sensor, temperature, dynamometer engine speed, EMU sensed engine speed and the active zone. Further commands and functions are used to determine whether the system is running, whether engine speed has stabilized in a zone and communication of these values to and from the global tuning function.

In order to allow the engine to start and run, the Link EMU is programmed prior to installation with a basic fuel and ignition configuration, referred to as the base

map. Once a tuning algorithm is run on the engine, performance data are extracted for analysis and comparison. The power output, fuel consumption and efficiency curves are compared to evaluate the performance of the tested tuning algorithm.

### 3 ENGINE TUNING ALGORITHMS

Optimum engine performance is typically obtained at a constant air–fuel ratio ( $\lambda$ ) and the most desirable settings for emissions are obtained when  $\lambda$  is valued around 1.0 [1, 6]. Small reductions of  $\lambda$  can optimize power output but may lead to sizeable increases in emissions. Similarly, values of  $\lambda$  slightly greater than 1.0 tend to increase efficiency, up to a point, as the engine runs leaner. Ignition timing was initially tuned in the base map and not considered in this tuning process. Since changes to this variable are easily added as a second step, it was not considered until an understanding of the effect of automation on tuning  $\lambda$  was performed.

Several algorithms for engine tuning to meet specified performance requirements can be developed, implemented and tested using performance specifications developed by any method. Two basic approaches tune for either maximum torque or a specific target stoichiometric  $\lambda$  sensor reading (air–fuel ratio), enabling tuning for performance or economy, as desired.

There are two primary steps for each of the automated tuning algorithms:

1. Tune the master fuel setting.
2. Tune the fuel setting for each of the zones that can be tested.

The master fuel setting is a global multiplier for every fuel setting in the engine map. The higher the master fuel value, the higher the fuel input for every zone. This value is tuned to move every zone in the engine map collectively to a region close to the desired performance. Once the master fuel value is set, the engine map is fine tuned by individually tuning each zone. The master fuel is automatically set by the automated tuning software while running the engine at approximately 50 per cent of maximum engine speed and 50 per cent throttle, before specific automated tuning of each zone. Hence, the tuning algorithms all have two essential steps, an initial approximation using the master fuel setting and specific tuning of each operating zone in the engine map.

The engine is commanded to operate in the centre of the zone, for each zone tuned. Since each zone in the Link EMU is 500 r/min wide and 30° of throttle position high, the engine is operated from 250 r/min at 15° throttle in 500 r/min and 30° steps. Note that these steps are fixed by the Link EMU; however, zone size in general is a trade-off between engine map size and complexity, and the resolution required. It is important to note that not all zones are 'reachable' in that the engine does not

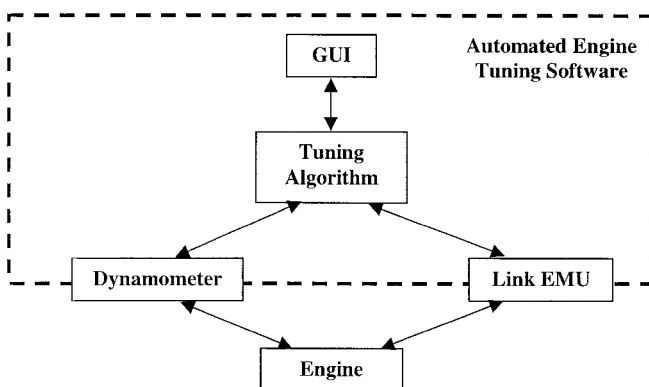


Fig. 2 Automated engine tuning system architecture

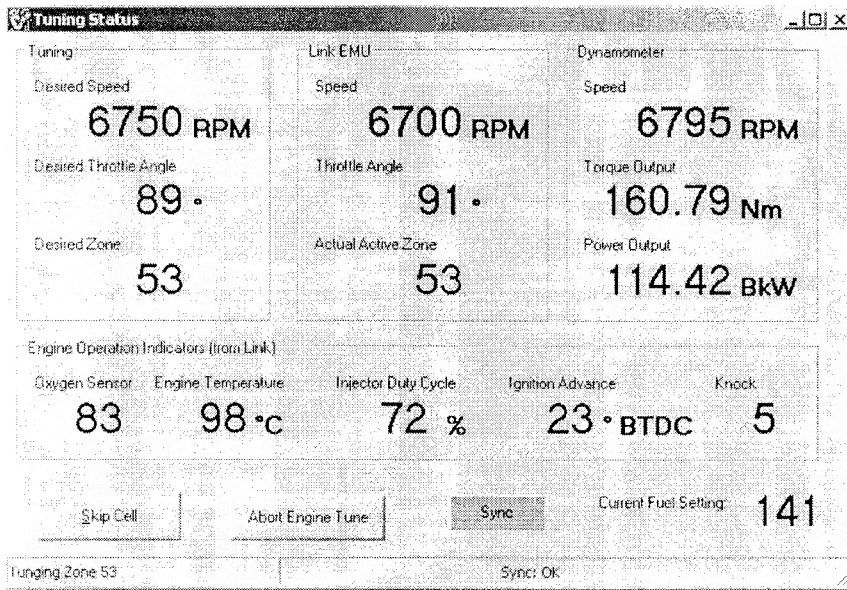


Fig. 3 Automated engine tuning status window

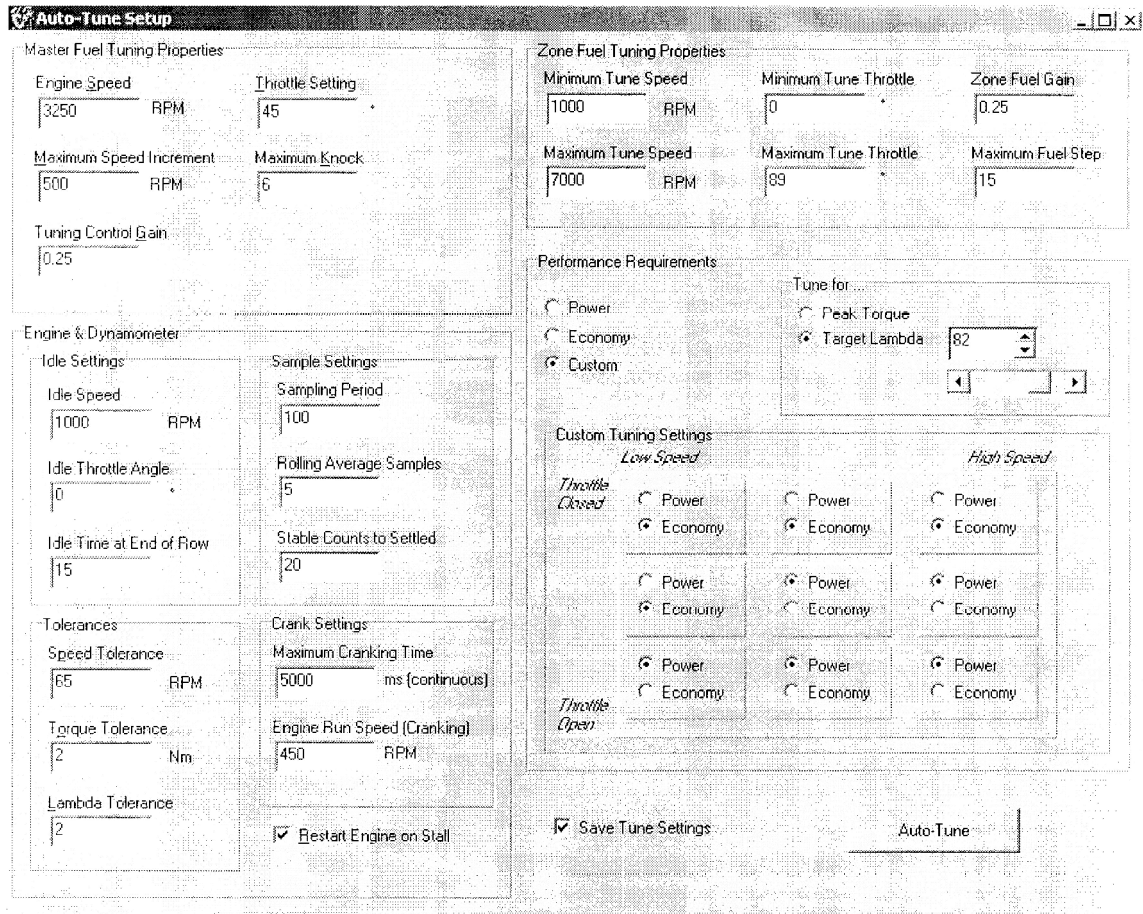


Fig. 4 Automated engine tuning set-up window

operate at 250 or 10000 r/min, for example. The automated system tunes the engine for each zone that is reachable by automatically moving on when steady state operation cannot be achieved in a given zone.

In the tuning process the engine is first incremented in speed over the entire row for a given throttle position. Each row is followed by incrementing and repeating the process for subsequent throttle positions. The addition

of a 15 s pause at idle before the throttle angle is increased allows time for engine cooling to prevent overheating as the cooling system employed for these tests was not adequate for high speed operating conditions. A larger cooling system would have alleviated this condition and eliminated the need for a pause.

Owing to a dynamometer controller fault, it was difficult to operate the system effectively at very low speeds. Hence, the tuning process was run for operating zones ranging from 500 to 7000 r/min. The user can restrict the operating range for engine speed to specified ranges as required. Note that if the engine reaches a zone in which it cannot operate, indicated by being unable to attain the commanded speed at the given throttle position, the system times out, automatically skips the zone and moves to the next row of throttle positions.

### 3.1 Tuning for target air–fuel ratio

This tuning method uses the factory  $\lambda$  sensor to monitor the amount of oxygen present in the exhaust gases and hence the air–fuel ratio. The goal is to tune the EMU for peak torque while holding a constant  $\lambda$  value to a small range around 1.0. The algorithm illustrated in Fig. 5 implements a simple proportional control system on both the master fuel setting and each of the zone fuel settings to perform this task. The control gain,  $K$ , controls the rate at which the actual air–fuel ratio approaches the specified level.

This simple linear proportional control system is extremely effective in terms of its settling time and the steady state error. In testing, the most appropriate gain for this system was found to be 0.25; however, the system can be easily modified to find the best value adaptively. Any positive value less than 1.0 should lead to steady convergence. The controller implemented has a slight non-linearity owing to a limiter placed on the change in controller output, to limit the amount the fuel can be changed in any given step of the algorithm. Figure 6 shows the basic steps taken in this tuning process.

During testing of this algorithm a tolerance of  $\pm 2$  per cent on the target  $\lambda$  sensor measurement is employed. Higher tolerances produce significant deviations from the target performance and lower tolerances make convergence difficult. Tolerances can be modified via the user interface for any system hardware.

Overall, this tuning algorithm employs simple control systems to tune each operating zone of the engine map automatically to match a desired  $\lambda$  value. This algorithm provides a solid base for further development and enables automated tuning. The control algorithm is equally effective if tuning for different target  $\lambda$  values in each zone.

### 3.2 Tuning for maximum torque output

Engine tuning for maximum torque provides a different tuning approach focused on maximum power. As with the prior algorithm this controller varies the amount of fuel supplied to the engine and logs the resulting torque output, while leaving ignition advance unchanged from the base map for this study. For example, when the fuel supply is increased, torque comes to a peak and then drops off, and the upper limit for fuel supply can be determined for that zone. This limit makes fuel sampling iterative, as the fuel value associated with the peak torque value must be found by first passing over the peak and then locating it via a simple peak detection routine with a tolerance of 1 per cent, or approximately 1.5 N m at peak torque.

Simple data smoothing is also performed so that spurious samples do not significantly affect the results. The end results are passed back to the overall controller and put into the engine map and the tuning algorithm is incremented to the next zone where the process is repeated. The automated software controlling the process handles moving from zone to zone. These basic process steps are essentially the same as those for the target  $\lambda$  method.

### 3.3 Data filtering

The Test Automation TA2000 unit employs a simple controller to offer throttle position, constant speed and constant torque control modes. This controller should normally control the dynamometer to within 10 r/min; however, during the tests it only held engine speed to within 30–60 r/min. The inability to hold engine speed tightly leads to sizeable speed fluctuations, particularly at low speed; however, it also presents a worst-case test situation for automated engine tuning.

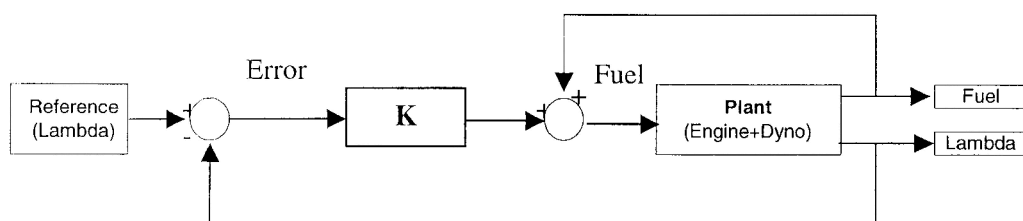


Fig. 5 Block diagram for target  $\lambda$  automated tuning algorithm

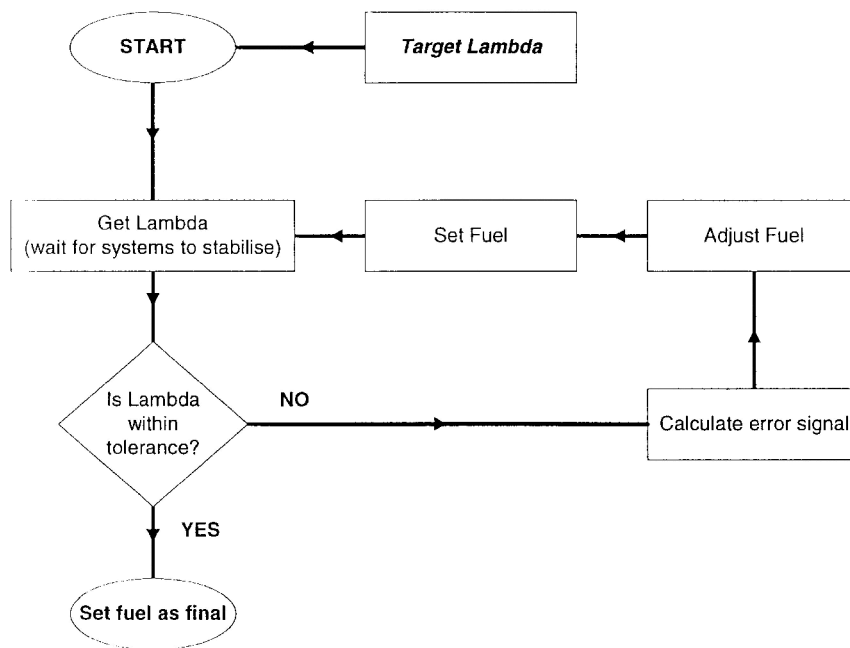


Fig. 6 Basic process steps for target  $\lambda$  controller

To address these issues, simple digital low pass filters are applied to smooth large variations in the data stream. The dynamometer is sampled by the ADC at 3 kHz and is set up to average the data in lots of 50–100 samples. To increase the smoothness further, a rolling average is performed using five filtered samples at a time. This filtering is applied to the engine torque, engine speed and oxygen sensor outputs. This approach low pass filters the data ranging from 0 to 12–60 Hz, depending on the exact filtering employed. Hence, interactions with dynamometer dynamics in the 30 Hz range are possible although no indications were noted as the implemented filtering keeps data bandwidth under 30 Hz.

#### 4 TESTING, RESULTS AND DISCUSSION

The user interface code was modified so that, each time the status screen is updated, the information displayed is also logged to a file. It was not possible to obtain the same level of information from the OEM EMU as from the Link EMU, making some comparisons difficult. To illustrate and prove the concept of automated engine tuning, the engine is tuned for a constant  $\lambda$  value using the Link EMU, and the resulting performance compared with the OEM EMU for torque and power.

Figures 7 and 8 show the resulting performance for the Toyota EMU and the automated engine tune respectively. These figures show the engine map in the horizontal plane with power on the vertical axis. The maximum power for this  $\lambda$ -tuned engine map is consistently achieved around 6750 r/min, matching the manufacturer's listed specifications and OEM EMU test

results. The automated engine tune experiences a slight tapering off in power soon after the peak output, while the factory EMU retained peak power for a more extended period. The maximum power was between 110 and 115 kW for each map, approximately 7 per cent lower than the manufacturer-listed value of 121 kW, perhaps owing to age and wear.

At lower engine speeds, variations of  $\pm 5$  kW occur owing to speed control issues with the dynamometer. When the dynamometer cannot hold the speed tightly, small changes in fuel input result in larger variations in  $\lambda$ , presenting a more difficult control problem. At higher speeds, the  $\lambda$  tuning procedure resulted in equivalent or slightly greater power output values. Overall, the automated engine tune using constant  $\lambda$  results in very similar torque and power performance to that of the OEM EMU. The OEM EMU is probably tuned for  $\lambda = 1.0$ , except at high speeds where fuelling may be increased to something approaching  $\lambda = 0.9$  to improve top end performance. In contrast, the automated  $\lambda$  tune method tunes this value to be within  $\pm 2$  per cent of  $\lambda = 1.0$  for all zones. However, the torque and power results are very similar to the OEM graphs and it is possible, with some added complexity, to tune each zone for a different target  $\lambda$  value to mimic, more exactly, the basic OEM tuning process.

The torque surfaces in Fig. 9 show the relatively flat torque curve associated with this engine for both the automated engine tune and manufacturer's EMU test results. Maximum torque output was approximately 180 N m at 5500 r/min for the OEM map using the exact data, with the automatically tuned results within 2 per cent of this value. The region of the torque curve where

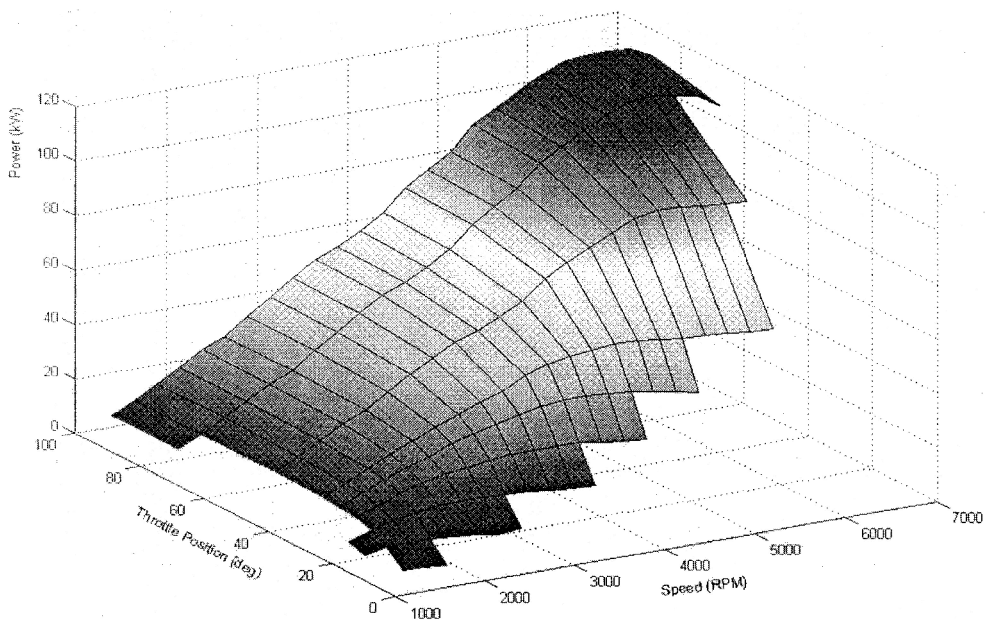


Fig. 7 OEM EMU power output (kW)

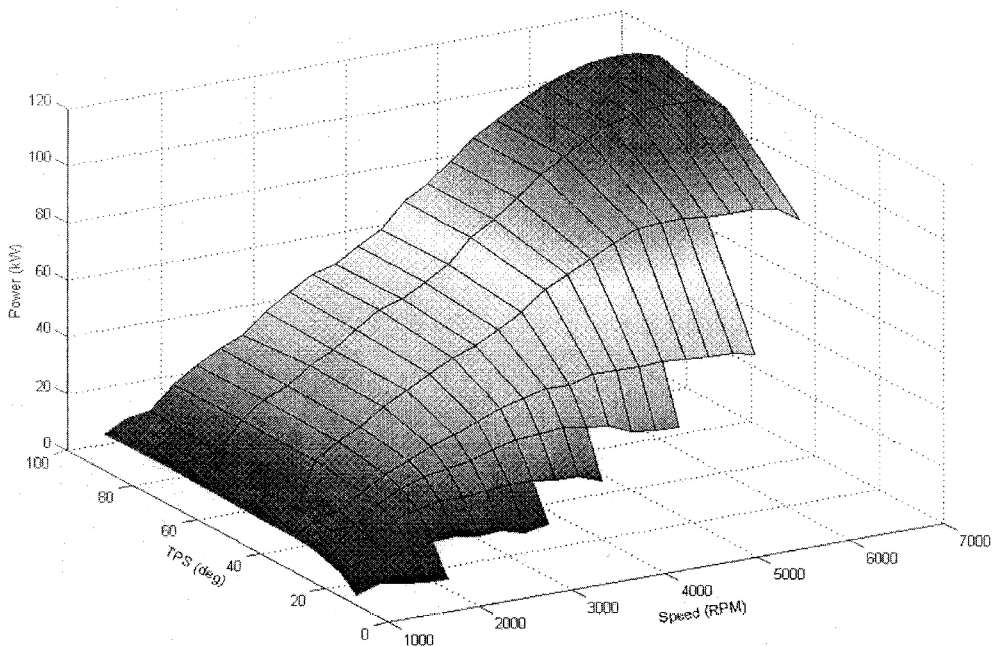


Fig. 8 Automated engine tuned for constant  $\lambda$  Link EMU power output (kW)

automated engine tuning achieved the smoothest result was in the top half of the engine speed range around the maximum torque values, where the dynamometer holds the engine speed more tightly. This result is due to the dynamometer controller delivering steady motion with tight speed control more rapidly at higher speeds. At lower engine speeds the impact of variation in speed control, as a percentage of the total, and the size of the engine map zones are evident in the less smooth result for the automated engine tune. The peak torque values for both cases are approximately 11 N m lower than the listed specification, probably as a result of age and wear

of this 1992 engine. The overall result is that the automated engine tuning concept is very close, at a first trial, to the OEM tune in terms of peak torque, meeting the aims of this project and proving the basic concept.

As expected, the manufacturer's EMU is programmed to achieve a high level of performance from the engine. The automated engine tune results in essentially equivalent performance to the manufacturer-listed specifications. Greater engine speed resolution of the engine map at lower engine speeds and throttle settings would limit some of the roughness of the automatically tuned torque response. The current engine map is defined in 500 r/min

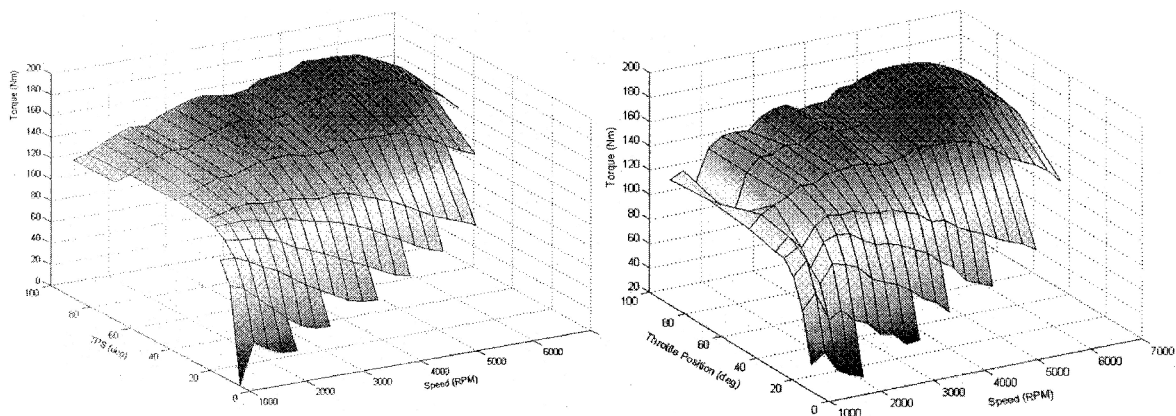


Fig. 9 OEM torque output (N m, left) and automated constant  $\lambda$  tuned EMU torque output (N m right)

increments and steps of this size at engine speeds around 1500 r/min represent significant changes in engine operating condition. However, greater engine speed resolution of the engine map would enable a smoother automatically tuned result. Greater engine speed resolution would also enable a potentially smoother, more optimal map to be obtained for this region and a potentially better overall result.

The automated target  $\lambda$  algorithm required approximately 30 min to complete. During this time, the centre of each Link EMU engine map zone was tuned for all zones in the 500–7000 r/min range and in the 0°–90° throttle range, where the first zone would be tuned at 750 r/min and 15°. Of all 48 operating zones available in this region of the engine map, only 36 were attainable by the test engine and the rest represented operating conditions that the engine could not reach (e.g. 6750 r/min and 15° throttle). Here again, higher resolution in the engine map may provide better results and greater flexibility, particularly at lower engine speeds.

At Canterbury, an equivalent manual test, tuning 50–100 zones for greater resolution, would take approximately 4 h. A very experienced technician could tune the equivalent 36 zones in as short a time as 1 h with a well-known engine. Given better dynamometer controller tuning to hold engine speed to a far tighter tolerance, the engine tuning tests presented here would take approximately 8–10 min, and an equivalent 50–100 zone test would take approximately 25–35 min. The increased speed of the automated tune would result directly from the ability to find and hold the steady state engine speed required, far more quickly than with the controller tuned. In either case, automation of this process offers significant time savings.

## 5 CONCLUSIONS

The development of a closed-loop, automated engine tuning system is demonstrated and the potential to produce consistent results, significant time savings and

resulting performance of a high standard is illustrated. Automation eliminates the requirement for full-time oversight by skilled technicians and engineers. While the tuning routines developed are simple, results from automated tuning of the Link EMU have reached a standard approaching that of the factory EMU for a much lower investment of time. Power and torque output data illustrate how the engine map can be tuned from a very rough base map to a polished result that enables smooth operation over the entire engine operating range. Power and torque results for the automated engine tune are reasonably similar to those obtained from the original manufacturer EMU with some roughness at lower engine speeds. The entire tune took approximately 30 min to deliver results and faster times could be achieved with better dynamometer controller settings. These times are significantly faster than comparable hand-controlled tests at the same facility illustrating the potentially large time saving available through automating these tests.

These automated results are achieved through the development of independent software interfaces between a PC an engine dynamometer and an after-market EMU. Further software encapsulates a global tuning algorithm using closed-loop feedback to command the system to desired states and to tune the engine map. A graphical user interface enables user input of tuning specifications for the entire map, each individual zone or larger regions of the engine map. There are also a variety of safety shut-offs and automatic restarts included to minimize the need for human intervention.

In summary, automated torque and target  $\lambda$  engine tuning procedures are developed and the  $\lambda$  tune is tested. The system has the flexibility to allow for further advancement in automated engine tuning. Areas where further work will have significant impact include the development of more sophisticated tuning algorithms that include emissions control, thermal efficiency, ignition timing and exhaust systems. Other areas of future research include tuning for dynamic conditions such as acceleration, expansion to include fuel usage



analysis, combinations of torque and  $\lambda$  tuning in different regions and repeating the system and results on a chassis dynamometer.

## REFERENCES

- 1 **Taylor, C. F.** *The Internal Combustion Engine in Theory and Practice*, 1985 (MIT Press, Cambridge, Massachusetts).
- 2 **Scotson, P. G.** and **Wellstead, P. E.** Self-tuning optimization of spark ignition automotive engines. *IEEE Mag.*, 1990, **10**(3), 94–101.
- 3 **Holmes, M.** and **Cockerham, K.** Adaptive ignition control. In Proceedings of Sixth International Conference on *Automotive Electronics*, London, 12–15 October 1987.
- 4 **Holmes, M., Wilcocks, D. A. R.** and **Bridgens, B. J.** Adaptive ignition and knock control. In Fisata 1988, Dearborn, Michigan, 25–30 September 1988, SAE paper 885065.
- 5 **Beck, T.** Current trends in the design of automotive electronic systems. In Proceedings of the Conference on *Design, Automation and Test in Europe*, 2001.
- 6 *Automotive Handbook*, 1993 (Bosch GmbH, Germany).