Lecture Notes in Electrical Engineering 882

Mohd Hasnun Arif Hassan
Mohd Hafizi Zohari
Kumaran Kadirgama
Nik Abdullah Nik Mohamed
Amir Aziz *Editors*

Technological Advancement in Instrumentation & Human Engineering

Selected papers from ICMER 2021

MOREMEDIA

Springer

Lecture Notes in Electrical Engineering

Volume 882

The book series *Lecture Notes in Electrical Engineering* (LNEE) publishes the latest developments in Electrical Engineering - quickly, informally and in high quality. While original research reported in proceedings and monographs has traditionally formed the core of LNEE, we also encourage authors to submit books devoted to supporting student education and professional training in the various fields and applications areas of electrical engineering. The series cover classical and emerging topics concerning:

- Communication Engineering, Information Theory and Networks
- Electronics Engineering and Microelectronics
- Signal, Image and Speech Processing
- Wireless and Mobile Communication
- Circuits and Systems
- Energy Systems, Power Electronics and Electrical Machines
- Electro-optical Engineering
- Instrumentation Engineering
- Avionics Engineering
- Control Systems
- Internet-of-Things and Cybersecurity
- Biomedical Devices, MEMS and NEMS

For general information about this book series, comments or suggestions, please contact leontina.dicecco@springer.com.

To submit a proposal or request further information, please contact the Publishing Editor in your country:

China

Jasmine Dou, Editor (jasmine.dou@springer.com)

India, Japan, Rest of Asia

Swati Meherishi, Editorial Director (Swati.Meherishi@springer.com)

Southeast Asia, Australia, New Zealand

Ramesh Nath Premnath, Editor (ramesh.premnath@springernature.com)

USA, Canada:

Michael Luby, Senior Editor (michael.luby@springer.com)

All other Countries:

Leontina Di Cecco, Senior Editor (leontina.dicecco@springer.com)

**** This series is indexed by EI Compendex and Scopus databases. ****

More information about this series at https://link.springer.com/bookseries/7818

Mohd Hasnun Arif Hassan ·
Mohd Hafizi Zohari · Kumaran Kadirgama ·
Nik Abdullah Nik Mohamed ·
Amir Aziz
Editors

Technological Advancement in Instrumentation & Human Engineering

Selected papers from ICMER 2021

Springer

Editors
Mohd Hasnun Arif Hassan
Faculty of Mechanical and Automotive
Engineering Technology
Universiti Malaysia Pahang
Pekan, Pahang, Malaysia

Mohd Hafizi Zohari
Faculty of Mechanical and Automotive
Engineering Technology
Universiti Malaysia Pahang
Pekan, Malaysia

Kumaran Kadirgama
Faculty of Mechanical and Automotive
Engineering Technology
Universiti Malaysia Pahang
Pekan, Malaysia

Nik Abdullah Nik Mohamed
Faculty of Mechanical and Automotive
Engineering Technology
Universiti Malaysia Pahang
Pekan, Malaysia

Amir Aziz
Faculty of Mechanical and Automotive
Engineering Technology
Universiti Malaysia Pahang
Pekan, Malaysia

ISSN 1876-1100 ISSN 1876-1119 (electronic)
Lecture Notes in Electrical Engineering
ISBN 978-981-19-1576-5 ISBN 978-981-19-1577-2 (eBook)
https://doi.org/10.1007/978-981-19-1577-2

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

To SARS-CoV-2 that has been rocking the world since 2019, on your bike and give us a break!

Foreword

Research in the Mechanical Engineering field has evolved into becoming a multidisciplinary field that is not only focusing on Mechanical Engineering per se, but also integrating knowledge in electronics, instrumentation, programming, signal processing, materials science, and so on. This is due to the fact that the solution to a problem nowadays does not only rely on a single discipline approach, but often requires an integration with other disciplines. Take a car for example, a purely mechanical system in the past has now been highly integrated with electronics, control system, artificial intelligence, and so on that results in only 50% or less components in a car being purely mechanical. This is just one of many examples of what was then a purely mechanical system, now has evolved into an integrated system that involves other disciplines.

Technological Advancement in Instrumentation & Human Engineering gathers selected papers presented in the International Conference on Mechanical Engineering Research 2021 (ICMER 2021) that covers topics related to human engineering, instrumentation, materials, modelling and simulation, and signal processing. This proceeding book symbolizes the multidisciplinary Mechanical Engineering research has evolved into. It is hoped that this book will promote multidisciplinary research in the Mechanical Engineering field towards a more efficient and comprehensive solution to a problem at hand. As a wise man who goes by the name of Isaac Newton once said, “If I have seen further, it is by standing on the shoulders of giants”.

Regards,

Mohd Hasnun Arif Hassan
Corresponding Editor

Contents

Human Engineering

Driving Data Analysis for the Development of Kuala Terengganu Driving Cycle

I. N. Anida, J. S. Norbakyah, W. N. J. H. W. Yussof, P. Walker, and A. R. Salisa

Abstract Driving cycle plays a vital role in the production and evaluating the performance of the vehicle. Driving cycle is a speed-time data set and as an important input for vehicle emission models. A problem coming with the second-by-second speed driving data is to analyze the big driving data. To analyze this big data, it is necessary to choose best big data analysis methods, which give opportunity to store, preprocess, detect outlier and apply classification or clustering algorithms. In this study, a set of driving data is stored, managed and analyzed using Tall Arrays (TA) and *k*-means clustering algorithms in MATLAB for the development of Kuala Terengganu Driving Cycle (KTDC). The objectives of this paper are; to store and manage driving data using TA in MATLAB, to develop a KTDC by using *k*-means clustering, and lastly to analyze the energy consumption and emissions of KTDC. Firstly, the driving data is collected in five different routes in Kuala Terengganu city at go-to-work times. Then the data is stored and analyzed in the MATLAB. The development of KTDC is by using *k*-means clustering approach. Finally, the energy consumption and emissions of KTDC is analyzed by using AUTONOMIE software. KTDC is successfully developed with 35.15 km/h in average speed and 12 micro-trips.

Keywords Driving cycles · Hybrid electric vehicles · Fuel economy · Emissions · Tall arrays · *k*-means clustering · Micro-trips

I. N. Anida · J. S. Norbakyah · W. N. J. H. W. Yussof · A. R. Salisa (✉)
Faculty of Ocean Engineering Technology and Informatics, Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia
e-mail: salisa@umt.edu.my

J. S. Norbakyah · A. R. Salisa
Energy Storage Research Group (ESRG), Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia

Renewable Energy and Power Research Interest Group (REPRIG), Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia

P. Walker
School of Mechanical and Mechatronic Engineering, Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW 2007, Australia

M. H. A. Hassan et al. (eds.), *Technological Advancement in Instrumentation & Human Engineering*, Lecture Notes in Electrical Engineering 882,
https://doi.org/10.1007/978-981-19-1577-2_1

1 Introduction

Driving cycle is a representative speed-time profile of driving conduct of a particular locale or city [1–3]. The driving cycle characterizes the conduct of the vehicle on the road and has a broad range of users, from designing activity control frameworks to deciding the execution of vehicles. It is additionally utilized within the emission testing of vehicles for certification of emission standards [4]. It is widely used of application for vehicle manufacturers, environmentalists and traffic engineers.

Generally, the well-known drive cycles such as the WLTC (Worldwide harmonized Light vehicle Test Cycle), NEDC (New European Drive Cycle), FTP (Federal Test Procedure) and Japanese modes were seen originated from Europe, United States, Japan, China and India which are significant in the global automotive industry. Nevertheless, there are also other areas that developed local drive cycles such as Toronto, Canada [5], Iran [6], Singapore [7], Hong Kong [8] and others.

However, each one of the constructed driving cycles is not representing the actual situation in Kuala Terengganu (KT), the capital city of the state of Terengganu, Malaysia. According to [9], driving cycles play a critical role in vehicle design, and if vehicle manufacturers focus solely on a single driving cycle during vehicle construction and design, there is a risk that the design is optimized for this specific driving cycle, and the result for another driving cycle may be non-robust and sub-optimal. Therefore, an actual KT city work route driving cycle needs to be constructed and characterized in order to help other researchers to continue the studies related to emissions and fuel consumption in KT city.

Vehicle driving cycle is a series of point for speed of vehicle versus time which is mainly used to evaluate the performance of either the vehicle or engine. However, tremendous and big parallel data for thousands of seconds' data will cram the system and will cause various problems such as data storage efficiency, security issues, services quality, data governance policies and many more. Big Data (BD) is a new popular term used to describe the remarkably rapid increase in the volume of structured and unstructured data. Big data accuracy can lead to more assured decision making, and better decisions can lead to increased operational efficiency, cost savings, and risk reduction [10].

In the MATLAB R2020a version, users are presented with a data array format called Tall Arrays (TA) to make BD operations easy [11]. With this type of data, it is now possible to process BDs that are quite long, normally unprocessed and scattered. Moreover, with this new data type, users continue to use the MATLAB code and functions they are accustomed to.

In this study, a sample of TA application was performed using the actual driving data in Kuala Terengganu city. The second-by-second speed data are collected at Go-to-Work (GTW) time which at 7.30, 8.00 and 8.30 a.m. with 10 runs of data along five different routes. Then the data is analyzed to develop a Kuala Terengganu Driving Cycle (KTDC).

The structure of the work is organized as follows: in the second chapter, a brief description of the methodology involving data collection, TA in MATLAB, *k*-means clustering, and KTDC development was provided. Finally, in the third chapter, the results and analysis are presented.

2 Methodology

Figure 1 shows the flow chart to develop a driving cycle in Kuala Terengganu (KT) along five different routes which are Route A, B, C, D and E in Kuala Terengganu city at 7.30, 8.00, and 8.30 a.m. The KT driving cycle's inputs are second-by-second speed which collected at GTW times of 7.30, 8.00, and 8.30 a.m., with 10 data runs. All five routes were chosen for their traffic volume in KT city. In this research, the on-board measurement method will be used using a Global Positioning System (GPS). The data gathered then will be stored and managed using TA in MATLAB. Then the features from each micro-trip such as average speed and percentage of idle will be extracted. The clustering of the micro-trips using *k*-means method will be took place in order to develop the final driving cycle of Kuala Terengganu along Route A, B, C, D and E at GTW times.

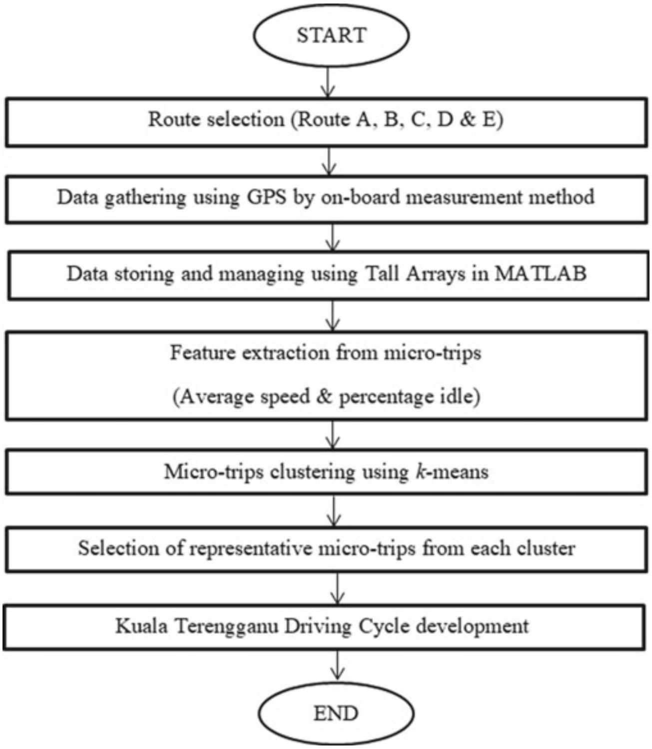

Fig. 1 Flow chart of the study

2.1 Data Collection

Route A, Route B, Route C, Route D, and Route E are the selected routes for KT driving cycle from Kampung Wakaf Tembesu to Wisma Persekutuan as shown in Fig. 2. According to the Malaysian Ministry of Works, these five routes are the most frequently used by Kuala Terengganu residents as 'route-to-work' routes [12]. In this study, GPS-based on-board measurement method speed-time data are collected along the selected route from Kampung Wakaf Tembesu to Wisma Persekutuan. Because of its population, Kampung Wakaf Tembesu was chosen as the starting point. On the other hand, Wisma Persekutuan was chosen as the end point because most government sectors are located in Wisma Persekutuan and it is nearby.

With 10 runs, data was collected at GTW times along the chosen road. There are three types of data collection techniques or methods: chase car technique, on-board measurement technique, and combination of on-board measurement and circulation driving. A chase car technique is a type of data collection that involves following a target vehicle while recording second-by-second speed data. On-board measurement is when speed-time data is collected using a real-time logging system installed on a specific vehicle along a predetermined route. Finally, the combination of two techniques of on-board measurement and chase car is also known as the hybrid method [13]. For the KT driving cycle, the on-board measurement technique will be used for

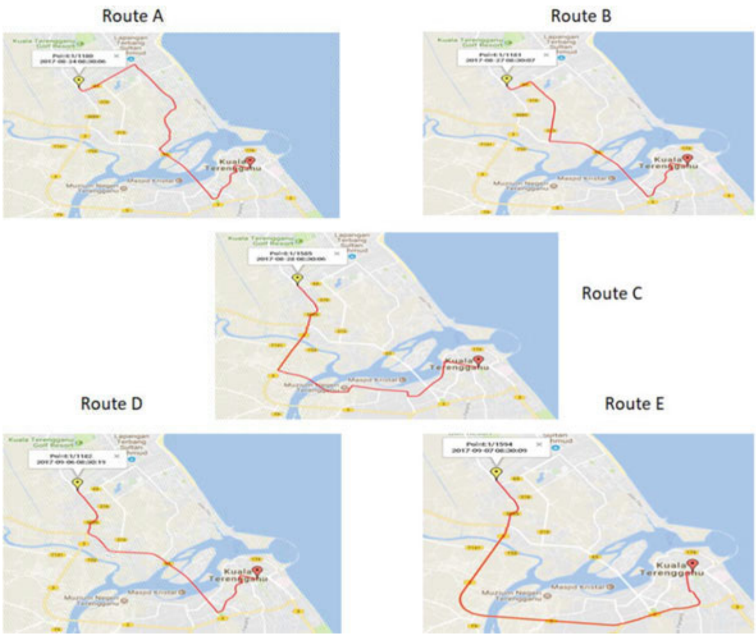

Fig. 2 Routes selection; route A, B, C, D and E

data collection because it is more appropriate for KT drivers' irregular behaviour to avoid risks such as accidents and sudden loss of control.

2.2 *MATLAB Tall Arrays*

Presently, the amount of data observed in business, science, and even individual applications has increased to such an extent where storing and processing data on personal computers is nearly impossible to produce significant results. In point of fact, an analytical system is made up of data collected from any scientific instrument and its associated systems. When working with a BD that is complex to tackle using standard methods on multiple devices, the following problems occur where [14]; Conventional methods do not work, accessing and analyzing data is difficult, learning new tools and programs for BD analysis is complicated, and managing data in physical environments is quite impossible.

Converting huge data sets into smaller bits is one technique to solve these issues. It will also be easy for users to use recognized or recognizable programming language codes (such as MATLAB codes). TA which is firstly introduced with MATLAB R2016b is a new filing scheme that treats different types of data in many different files as a single array/table [15]. TA supports many data types such as numeric, table, date time, categorical and character format. The data obtained can be used in conjunction with statistics and machine learning toolboxes. Figure 3 shows the stages of BD analysis in MATLAB [15].

In this work, MATLAB TA will be used to process a driving dataset which has the size of 60,000 × 6 and also to understand analyzing big data sets that cannot be processed with traditional functions and individual computers. Regardless of the fact that our dataset is not notably big, the functions and applications of the study can be used to examine any non-memory data, despite the number of rows. MATLAB TA analyses large volumes of data in the same way that it analyses small volumes of data. The most significant distinction is that tall arrays do not utilize memory to assess all rows of the dataset unless requested by the user.

A data store must be setup before processing using TA. Users will be able to access a data collection as a result of this. The data in the data collection can be found in

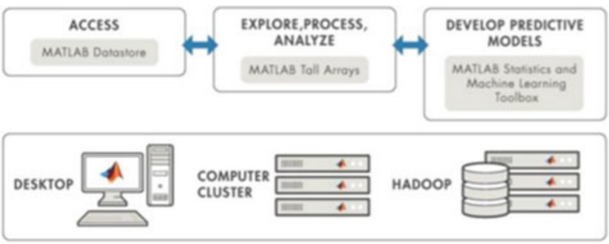

Fig. 3 Analyzing big data using MATLAB

one huge file or multiple smaller files. The data in TA could be numerical, logical, date, time category, or character structured. Only the "tall" function can be used to convert data in the data store into TA format. One of the most important features of MATLAB TA is that it does not immediately evaluate the operations performed, and it should not be evaluated until the user wants to see the results. For this study, only average speed and percentage idle will be calculated in order to develop a driving cycle.

2.3 K-means Clustering

Micro trips are used to develop a drive cycle. A micro-trip is a trip between two consecutive time points with zero vehicle velocity [16]. Each micro trip begins with an idle phase and concludes with a decelerating phase that approaches zero. The entire data set must be divided into a number of micro-trips. Following this process, a large number of micro-trips can be obtained for all collected data. The micro-trips are then divided into groups based on traffic conditions, such as congested traffic flow, medium traffic flow, and clear traffic flow. The *k*-means algorithm will be used to cluster the micro-trips.

K-means is one of the most basic unsupervised learning algorithms for clustering. The procedure follows a simple and easy method for classifying a given data set using a fixed number of clusters (assume *k* clusters). Clustering in N-dimensional Euclidean space, R_N, is the process of portioning a given set of n points into a number, say *K*, of groups, based on some similarity or dissimilarity metric. Let the set of n points $\{x_1, x_2, \ldots., x_n\}$ be presented by set *S*, and *K* clusters be represented by $C_1, C_2, \ldots., C_k$. Then:

$$C_i \neq \emptyset \text{ for } i = 1, \ldots, K$$
$$C_i \cap C_j = \emptyset \text{ for } i = 1, \ldots., K,\ j = 1, \ldots., K \text{ and } i \neq j$$
and:

$$U_{i=1}^{K} C_i = S \tag{1}$$

The *k*-means algorithm endeavours to solve the clustering issue by optimizing a given metric. The steps of the *k*-means algorithms are described briefly below (17):

Step 1: Determine a value for *k*. The value of *k* in this study is determined by the traffic condition, which includes congested traffic flow, medium traffic flow, and clear traffic flow.
Step 2: Set up the *k* cluster centres (randomly, if necessary).
Step 3: Assign the class memberships of the total data, N, to the nearest cluster centre.
Step 4: Re-estimate the *k*-cluster centres based on the memberships found above.

Step 5: Exit if none of the N data changed memberships in the previous iteration. Otherwise, proceed to Step 3.

2.4 KTDC Development

The proposed method for developing the driving cycle is to cluster micro-trips. To cluster the micro-trips, driving features must first be extracted. The micro-trips can be used to extract a variety of driving characteristics. However, only two features will be used for this purpose: average speed and percentage of idle. These two characteristics were chosen because they will have the greatest impact on emission [18]. The micro-trips are clustered into three groups as shown in Fig. 4 using the *k*-means clustering method. Each group has its own characteristics and represents a different traffic condition: clear traffic, medium traffic, and congested traffic.

The representatives of micro-trips are then identified in order to generate the driving cycle for each cluster. The representative micro-trips will be those that are closest to the cluster centre. The micro-trips chosen for each group are depicted in Figs. 5, 6 and 7. The micro-trips will then be combined to produce the final driving cycle of Kuala Terengganu along Routes A, B, C, D, and E at GTW times.

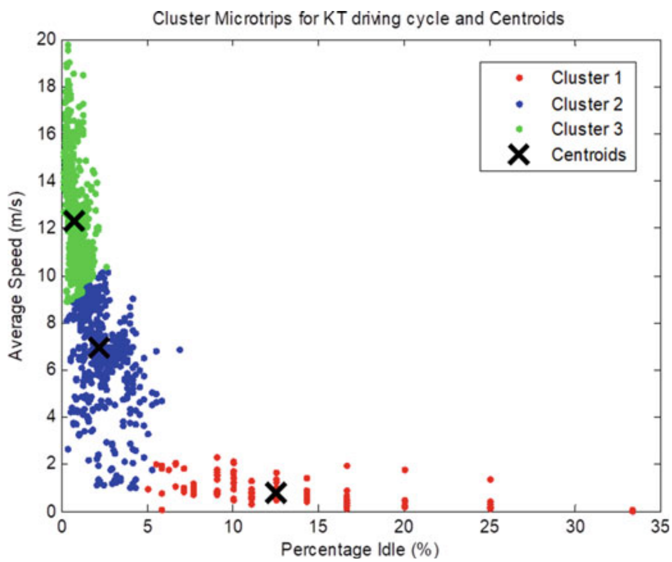

Fig. 4 Clustering of micro-trips

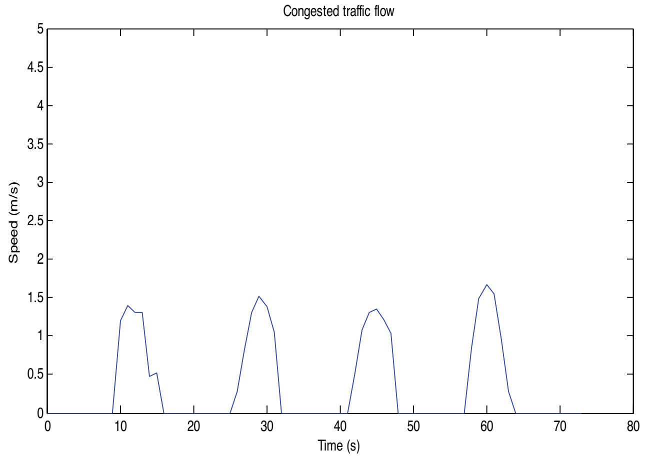

Fig. 5 Cluster one; congested traffic flow

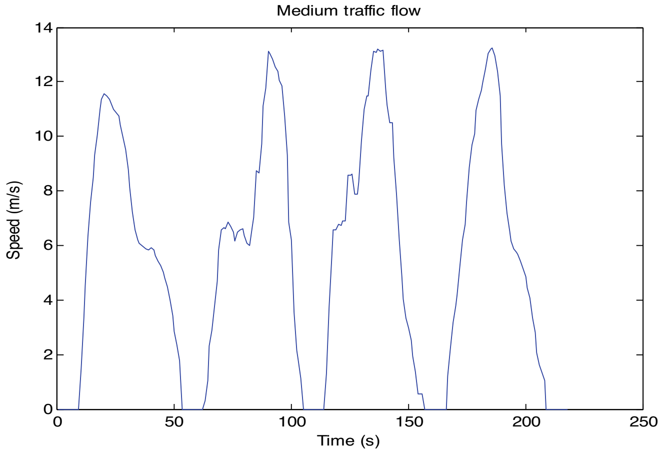

Fig. 6 Cluster two; medium traffic flow

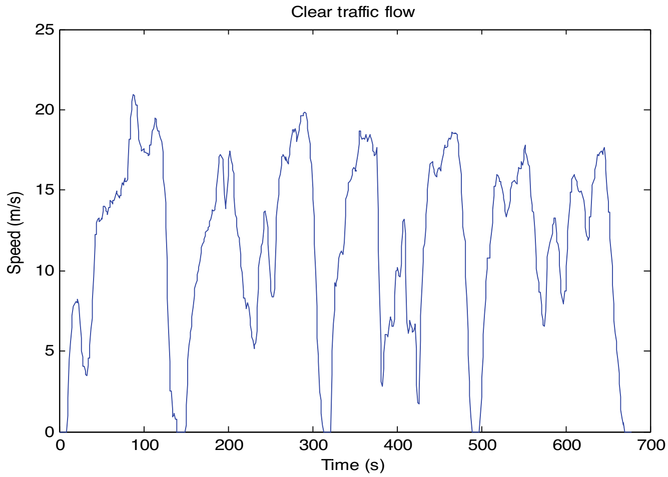

Fig. 7 Cluster three; clear traffic flow

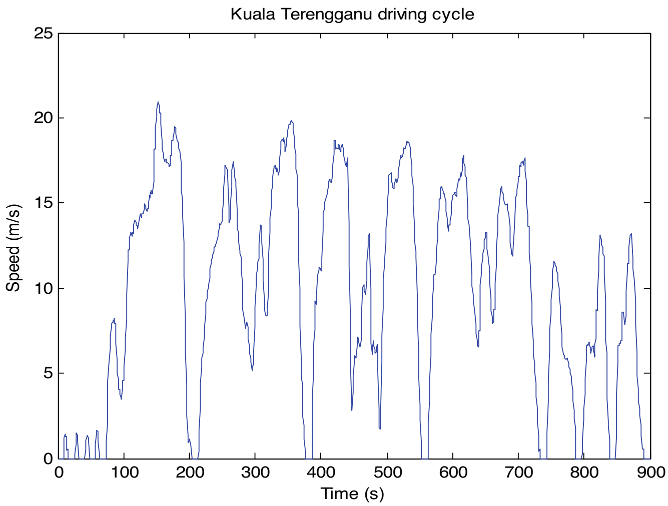

Fig. 8 Kuala Terengganu driving cycle

3 Results and Analysis

3.1 KTDC

After collecting all of the driving cycle data along five different routes at 7.30, 8.00, and 8.30 a.m., the final development of the KT driving cycle can be completed. Figure 8 depicts the final KT driving cycle. The total distance is 8.8 km, with a total of 12 micro-trips. Table 1 summarises the characteristics of the KT driving cycle in terms of nine assessment parameters. The developed driving cycle yielded the following results:

1) The speed range of more than 10 km/h was dominant. This is due to the high volume of traffic on KT city routes.
2) Micro-trips at higher speed ranges are longer than micro-trips at lower speed ranges. This is because a vehicle in free flow travels at a higher speed range with fewer stops due to less traffic congestion.
3) A 35.15 km/h average speed was recorded in the developed KT driving cycle, indicating that the vehicles are moving at a slower speed and that there are more micro-trips found below the average speed. As a result of the frequent stops along the road, more fuel is consumed and emissions are produced during that time period.

Table 2 listed the comparison of the KTDC and existing standard driving cycle which is NEDC (New European Driving Cycle) and UDDS (Urban Dynamometer Driving Schedule). From the table, it shows that there are not much different between KTDC, NEDC and also UDDS since all of them are driving cycle for the city. In Malaysia, NEDC is still being used by the local authorities for legislation purposes and by the local manufacturers and suppliers for evaluation purposes.

Table 1 Assessment parameters of KTDC

Parameters	KTDC
Distance travelled (km)	8.8
Total time (s)	953
Average speed (km/h)	35.15
Average running speed (km/h)	40.55
Average acceleration (m/s^2)	0.59
Average deceleration (m/s^2)	0.56
RMS (m/s^2)	0.73
Percentage idle (%)	12.10
Percentage cruise (%)	5.11
Percentage acceleration (%)	40.51
Percentage deceleration (%)	42.29

Table 2 Comparison between KTDC, NEDC and UDDS

Parameters	KTDC	NEDC	UDDS
Distance travelled (km)	8.8	11.02	12.89
Total time (s)	953	1180	600
Average speed (km/h)	35.15	33.6	31.51
Average running speed (km/h)	40.55	42.24	38.85
Average acceleration (m/s^2)	0.59	0.53	0.50
Average deceleration (m/s^2)	0.56	0.72	0.58
RMS (m/s^2)	0.73	0.14	0.68
Percentage idle (%)	12.10	16.95	17.66
Percentage cruise (%)	5.11	38.81	7.96
Percentage acceleration (%)	40.51	23.56	39.71
Percentage deceleration (%)	42.29	17.29	34.67

Table 3 Comparison of PHEV, HEV and conventional vehicle engine

Parameters	PHEV	HEV	Conventional
Fuel economy (mile/gallon)	212.84	53.78	32.2
Fuel consumption (l/100 km)	1.11	4.37	7.3
CO_2 emissions (g/mile)	26.1	103.3	172.82

3.2 Energy Consumption and Emissions Analysis

After developing the driving cycle, the fuel rate, such as fuel consumption and fuel economy, and emission can be calculated using AUTONOMIE software version v1210. AUTONOMIE is a design, simulation, and analysis tool for automotive control systems. It is forward simulation software that is mathematically based and it is based on MATLAB, with MATLAB data and configuration files and models built in Simulink. Table 3 displays the KTDC fuel rate and emissions for conventional engine vehicles, hybrid electric vehicles (HEV), and plug-in hybrid electric vehicles (PHEV). A vehicle's emissions will produce carbon monoxide (CO_2) gas.

The table clearly shows that the PHEV is the best powertrain among conventional engines and HEVs, with the lowest fuel consumption and emission values and the highest fuel economy value. This is due to the power split type PHEV's design, which incorporates one engine, two motor-generators (MGs), and multiple planetary gears [19]. As a result, it will reduce emissions and energy consumption.

4 Conclusion

The development of KT driving cycle is successfully done using micro-trips and *k*-means clustering method. The data are collected from predetermined initial location to final location along Route A, B, C, D and E at Go-to-Work times which are 7.30, 8.00, and 8.30 a.m. All the driving data is successfully stored and managed in the Tall Arrays in MATLAB for the development of the driving cycle. For the energy consumption and emission analysis, it is proven that PHEV uses less fuel consumption and emits less emission compared to HEV and conventional vehicle powertrain. For the future work, in order to develop an accurate Malaysian Driving Cycle, every states and cities in Malaysia should be considered.

References

1. Zhao X, Zhao X, Yu Q, Ye Y, Yu M (2020) Development of a representative urban driving cycle construction methodology for electric vehicles: a case study in Xi'an. Transp Res D 81:102279
2. Borlaug B, Holden J, Wood E, Lee B, Fink J, Agnew S, Lustbader J (2020) Estimating region-specific fuel economy in the United States from real-world driving cycles. Transp Res D 86:102448
3. Kaymaz H, Korkmaz H, Erdal H (2019) Development of a driving cycle for Istanbul bus rapid transit based on real-world data using stratified sampling method. Transp Res D 75:123–135
4. Arun NH, Mahesh S, Ramadurai G, Shiva SM (2017) Development of driving cycles for passenger cars and motorcycles in Chennai, India. In: Sustainable cities and society
5. Amirjamshidi G, Roonda MJ (2015) Development of simulated driving cycles: case study of the Toronto Waterfront Area. Transp Res D 34:255–266
6. Fotouhi A, Montazeri-Gh M (2013) Tehran driving cycle development using the *k*-means clustering method. Scientia Iranica Trans A Civil Eng 20(2):286–293
7. Ho S-H, Wong Y-D, Chang VW-C (2014) Developing Singapore driving cycle for passenger cars to estimate fuel consumption and vehicular emissions. Atmos Environ 97:353–362
8. Hung WT, Tong HY, Lee CP, Ha K, Pao LY (2007) Development of a practical driving cycle construction methodology: a case study in Hong Kong. Transp Res D 12:115–128
9. Nyberg P, Frisk E, Nielsen L (2016) Driving cycle equilance and transformation. IEEE Trans Veh Technol
10. Stergiou C, Psannis KE, Gupta BB, Ishibashi Y (2018) Security, privacy, efficiency of sustainable cloud computing for big data and IoT. Sustain Comput
11. Tordoff B (2016) Big data and tall arrays. In: MATLAB EXPO 2016 United Kingdom. http://www.matlabexpo.com/uk/2016/. Accessed 15 Apr 2018
12. Ministry of Works Malaysia (2015) 2014 Road Traffic Volume Malaysia (RTVM). Ministry of Works Malaysia, Highway Planning Division
13. Galgamuwa U, Perera L, Bandara S (2015) Developing a general methodology for driving cycle construction comparison of various established driving cycles in the world to propose a general approach. J Transp Technol 5:191–203
14. Ekici S, Unal F, Akbulut Y, Sengur A (2018) Electricity consumption analysis with Matlab Tall Arrays. In: IETS conference, May 2018
15. MathWorks (2018) Big Data with MATLAB. https://ch.mathworks.com/solutions/bigdata-matlab.html. Accessed 15 Apr 2018
16. Wang Q, Huo H, Yao H, Zhang Z (2008) Characterization of vehicle driving patterns and development of driving cycles in Chinese cities. Transp Res D 13:289–297
17. Maulik U, Bandyopadhyay S (2000) Genetic algorithm based clustering technique. Pattern Recogn 33(9):1455–1465
18. Fotouhi A, Montazeri-Gh M (2013) Tehran driving cycle development using the *k*-means clustering method. Scientia Iranica Trans A Civil Eng 20(2):286–293
19. Son H, Park K, Hwang S, Kim H (2017) Design methodology of a power split type plug-in hybrid electric vehicle considering drivetrain losses. Energies 10:437

Investigation on Lane Changing Manoeuvres of an Autonomous Vehicle by Using Model Predictive Control (MPC) and Proportional-Integral-Derivative Control (PID) for Steer-by-Wire (SBW) System

P. M. Heerwan, N. E. Z. Mohamad, M. A. Zakaria, M. I. Ishak, and S. M. Asyraf

Abstract Lane changing is one of the crucial tasks for an autonomous vehicle to avoid an obstacle. This task can be performed by controlling the throttle, brake, and steering actuators appropriately based on the analysis of the vehicle's surroundings. The problem with lane changing is that the control strategy is too complex and needs a high processor for real-time data analysis. In addition, lane changing involves high-level control for vehicle trajectory and low-level control for controlling the steering actuator. This study proposed a well-known method, namely Model Predictive Control (MPC), to determine the vehicle's lateral position and yaw angle during lane changing maneuver. The optimum steering angle command can control the steer-by-wire (SBW) system from the lateral position and yaw angle in MPC. The Proportional-Integral-Derivative (PID) controller is implemented to control the steering wheel angle in the SBW's system. Then, the SBW system will turn the wheel of the vehicle plant. From the simulation result, the PID controller can converge the error although the vehicle's speed is increasing. The result shows that the mean absolute error (MAE) of the SBW system decreases slightly from 0.0115 to 0.0079 as the speed increase from 16 to 41 km/h. From this study, it can be concluded that the MPC and PID controllers can control the vehicle's trajectory during lane changing by calculating an optimum lateral motion and yaw angle to provide an optimum steering angle for the vehicle to change lanes successfully.

P. M. Heerwan (✉) · S. M. Asyraf
Faculty of Mechanical and Automotive Engineering Technology, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia
e-mail: mheerwan@ump.edu.my

N. E. Z. Mohamad · M. A. Zakaria
Faculty of Manufacturing and Mechatronic Engineering Technology, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia

M. I. Ishak
College of Engineering, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia

M. H. A. Hassan et al. (eds.), *Technological Advancement in Instrumentation & Human Engineering*, Lecture Notes in Electrical Engineering 882,
https://doi.org/10.1007/978-981-19-1577-2_2

Keywords Steer-by-wire (SBW) · Model Predictive Control (MPC) · PID control · Lane changing · Autonomous vehicle

1 Introduction

According to a Malaysian Institute of Road Safety Research (MIROS) report, frontal collisions are the most frequent type of accident in Malaysia [1]. The fatality rate is also alarming, which is approximately 7000 road users have died due to road accidents every year, while thousands have experienced other incapacities [2]. Furthermore, the number of pedestrians killed in traffic accidents is also worrying as it accounts for about 25% of the total number of accidents every year [3].

On the other hand, research on autonomous vehicles has given significant attention due to the rapid development of sensors and computing technology. These developments have given rise to the possibility of autonomous vehicle technology, which reduces vehicle accidents that cause by human beings, such as distraction [5] and limits of attention and perception [6]. Since safety is the primary aspect of the autonomous vehicle, the controller must be robust to uncertain conditions [4]. One example of uncertain conditions is when a vehicle try to avoid an unforeseen obstacle. The steering system is the primary system to avoid obstacles and manoeuvre the vehicle to a safe trajectory. In the steering system of an autonomous vehicle, the controllers and actuators replace the mechanical linkage of the steering system. This system is also known as the steer-by-wire (SBW) system, where the control algorithm for vehicle trajectory and optimum steering wheel angle input is compiled in the controllers. Then, an electrical signal will be transferred to the actuator, connected to the steering rack and pinion to steer the wheel [3].

The autonomous system has many control techniques to track the vehicle's motion based on the reference trajectory. Trajectory tracking is crucial to force a vehicle to follow the parameterised references, such as the geometrical path against time. There are two critical variables in trajectory tracking: the reference trajectory and the actual vehicle position [6]. The main purpose of the controller is to decrease the error between the reference trajectory with exact vehicle position by correcting the position errors.

In [8], trajectory tracking development can be classified into conventional control and intelligent control method. The conventional control is an established control method such as MPC, PID and neural network. However, intelligent control is an improvement of the established control method. This improvement usually tackles the conventional control method problems and increases the controller's robustness and aggressiveness in various driving conditions.

Pauca et al. proposed MPC for the vehicle's longitudinal and lateral dynamics during the overpassing manoeuvre, where both the vehicle's longitudinal velocity and trajectory satisfy the reference and constraints [7]. The authors used a nonlinear vehicle model to simulate an actual vehicle model. Then, the vehicle model is linearised before being simulated into the MPC algorithm.

M. Ali et al. proposed the Imperialist Competitive Algorithm (ICA) for optimising PID control parameters of the SBW system. The 10 degrees of freedom vehicle model has been simulated, and the result shows that the lateral motion and vehicle yaw angle errors can be reduced. Furthermore, their proposed method can also tune the vehicle's plant output to the desired trajectory to maintain its stability.

An improved MPC method for a vehicle's lateral control is recently introduced in [8], where the vehicle's stability is improved during high-speed cases. Li et al. used the steady-state response as the controller input instead of the proportional effect of the error. The authors also used saturation limits to constraint the input. The improved MPC that can steer the vehicle does not exceed the side slip limitation as the controller's input is the steady-state error instead of heading error.

In [9], the authors introduce an adaptive and predictive controller that focuses on improving the autonomous vehicle's tracking performance and lateral stability by considering the actual control such as torque input to the steering actuator. The researchers used a regression model based on previous data from input–output where the recursive least squares (RLS) algorithm identifies the parameters. Therefore, a control model is developed to adapt to varying road conditions with faster settling time and minimal overshoot. However, as the road gets more slippery, the controller has less aggressive manoeuvres to maintain lateral stability.

Furthermore, Pratama et al. [10] introduced an adaptive trajectory controller that can follow the desired trajectory reference while estimating the unknown slip parameter. The authors used the backstepping algorithm, which can determine the unknown slip parameter of the vehicle based upon the Lyapunov criterion. The simulation and experimental results of this research show that the angular velocity error vector of the dynamic tracking wheel is around 0.06 rad/s.

This paper investigates the performance of lane changing manoeuvre of an autonomous vehicle with SBW when integrated with MPC and PID control system. This study has two main objectives; the first objective is to establish a reference value for lateral and steer angle by conducting several lanes changing experiments at various velocities using an actual vehicle without SBW. The second objective is to evaluate the lane changing manoeuvre when integrated with MPC and PID control system using simulation based on the experimental reference value. The simulation result shows that the MPC and PID controllers can control the vehicle's trajectory during lane changing while maintaining the vehicle's stability even though the speed is increasing.

2 Experimental Setup

Figure 1 shows the experimental vehicle model, namely UMP Test Car, that equipped with a gyroscopic sensor and encoder. By using a gyroscope and rotary encoder, the lateral position and yaw angular velocity can be adopted. This collected data will be used in the MPC-PID control system as a trajectory reference. This ensures the

Fig. 1 Experimental setup

MPC-PID controller can control the vehicle's trajectory with a minimal steady-state error during lane changing manoeuvres at a different constant speed.

In the experiment, the driver will drive a vehicle with a constant speed of 16 km/h. When a vehicle is steady at 16 km/h, the driver will turn the steering wheel to change the lane, as shown in Fig. 2. Concurrently, the assistant at the rear seat will record the trajectory data, yaw angle and linear velocity of the vehicle by using a gyroscopic sensor and rotary encoder, respectively. The data is saved in the ROS bag, where the message data from ROS topics is stored. ROS bag and ROS topic are the tools inside the Robot Operating System (ROS). The experiment is repeated for 20, 29 and 41 km/h, respectively.

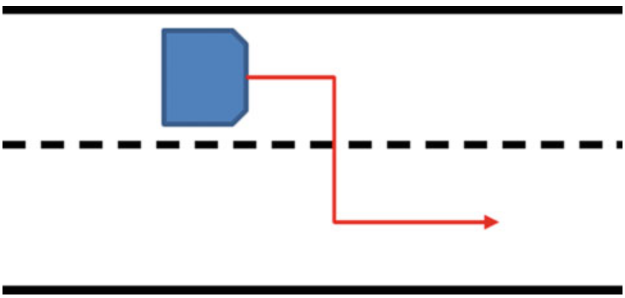

Fig. 2 Lane changing manoeuvre