J. Comp. Appl. Res. Mech. Eng. Vol. 11. No. 2, pp. 409-423, 2022 *DOI: 10.22061/JCARME.2021.7412.1987*

Journal of Computational and Applied Research in Mechanical Engineering

jcarme.sru.ac.ir

SRTTU

JCARME

ISSN: 2228-7922

Research paper

Braking intensity recognition with optimal K-means clustering algorithm

Ali Mirmohammad Sadeghi, Abdollah Amirkhani* and Behrooz Mashadi

School of Automotive Engineering, Iran University of Science and Technology, Tehran 16846-13114, Iran

Article info:

Article history:

Received: 11/10/2020

Revised: 12/07/2021

Accepted: 16/07/2021

Online: 19/07/2021

Keywords:

Vehicle safety systems,

Clustering,

K-means algorithm,

Hydraulic brake system.

***Corresponding author:**

amirkhani@iust.ac.ir

Abstract

Recognizing a driver's braking intensity plays a pivotal role in developing modern driver assistance and energy management systems. Therefore, it is especially important to autonomous and electric vehicles. This paper aims at developing a strategy for recognizing a driver's braking intensity based on the pressure produced in the brake master cylinder. In this regard, a model-based, synthetic data generation concept is used to generate the training dataset. This technique involves two closed-loop controlled models: an upper-level longitudinal vehicle dynamics model and a lower-level brake hydraulic dynamic model. The adaptive particularly tunable fuzzy particle swarm optimization algorithm is recruited to solve the optimal K-means clustering. By doing so, the best number of clusters and positions of the centroids can be determined. The obtained results reveal that the brake pressure data for a vehicle traveling the new European driving cycle can be best partitioned into two clusters. A driver's braking intensity may, therefore, be clustered as *moderate* or *intensive*. With the ability to automatically recognize a driver's pedal feel, the system developed in this research could be implemented in intelligent driver assistance systems as well as in electric vehicles equipped with intelligent, electromechanical brake boosters.

1. Introduction

Driver behavior recognition systems play a pivotal role in the development of advanced driving assistance systems (ADAS) and automated or semi-automated vehicles. In this regard, a great deal of research has been conducted on the development of driver behavior recognition systems [1, 2].

To address the difficulty of identifying abnormal driving behaviors, Jia et al. [3] proposed a recognition system based on a long short-term memory network and convolutional neural network (LSTM-CNN).

By this means, they were able to recognize dangerous driving behaviors (e.g., rapid acceleration, sudden braking, and rapid lane changing) that could jeopardize traffic safety. Zhang et al. [4] proposed a deep learning framework by combining convolutional neural networks (CNNs) and recurrent neural networks to learn features of driving behaviors from Controller Area Network-BUS sensor data. Xing et al. [5] proposed a driver intention inference

scheme to recognize the lane change intentions of drivers on highways. In another study, Xing et al. [6] took advantage of a low-cost camera and deep CNNs to develop a behavior recognition system that checks a driver's normal condition or his/her distraction during driving. The developed system achieved an accuracy of 91.4%.

K-means is an iterative clustering algorithm in which each iteration includes two phases: assigning queries to the adjacent cluster and updating the centroid of each cluster. The iteration terminates either in a fixed number of rounds or when the updates of the cluster centroids reach a given threshold [7]. Although K-means clustering problems are straightforward conceptually, they are computationally difficult to handle (NP-hard) [8, 9] and may require the application of powerful optimization algorithms [10].

Das et al. [11] used the differential evolution (DE) algorithm to solve an automatic K-means clustering problem for unsupervised, large data sets. The term "automatic" implies that the optimization algorithm is responsible for finding the best positions of cluster centroids and the best number of clusters. Due to its outstanding capabilities, the adaptive particularly tunable fuzzy particle swarm optimization (APT-FPSO) algorithm [12] can be effectively used to solve optimal K-means clustering problems.

This paper attempts to develop a system for recognizing a driver's braking intensity in a vehicle passing through six consecutive NEDC cycles. The problem is presented in the form of optimal K-means clustering, and the APT-FPSO algorithm is applied to find the best number of clusters as well as the best positions of cluster centers. The dataset which was used in previous computations is synthetically generated by employing a model-based technique. The results of this research can be reliably used in the development of advanced vehicle safety and control systems.

2. State of the art

A driver's behavior is manifested by the way he/she treats the gas/brake pedal. By using a combination of the gaussian mixture model (GMM) and hidden Markov model (HMM), Wang et al. [13] proposed a driver's braking intent prediction method based on his/her driving history. Compared to the support vector machine (SVM) filtering method, the proposed technique outperformed the rival methods with an accuracy of 90%, sensitivity of 84%, and specificity of 97%. Based on a considered driving cycle and the recognition of driver behavior from the accelerator pedal position and its rate of change, Guo et al. [14] proposed an optimal control strategy for plug-in hybrid electric vehicles. They improved energy consumption by 3.69% in the new European driving cycle (NEDC). By exploiting a hybrid unsupervised-supervised learning technique, Lv et al. [15] attempted to estimate the pressure of the master cylinder and, thus, eliminate the pressure sensor from the electronic stability program (ESP) module. A GMM model categorized a driver's braking intensity into three low, moderate, and intensive clusters for a vehicle traveling the NEDC.

In many real-world phenomena, there is not sufficient information to be used to train a learning machine. Besides, experimental data collection can sometimes become arduous and time-consuming. In such cases, scientists can take an alternative approach to model an original phenomenon as closely and accurately as possible. This imitation model can then be used to provide a good approximation of the responses to the real phenomenon. Nikzadfar and Shamekhi [16-19] designed an MLP artificial neural network (ANN) to predict semi-static in-cylinder phenomena in diesel engines. They used an engine dynamometer as an experimental test setup to generate a training dataset. The training data set was, though, insufficient in size. To deal with dataset scarcity, they used the experimental data to calibrate the thermodynamic model of an engine in AVL-Boost software. Then using the engine's calibrated model, they generated a dataset of sufficient size.

Gao et al. [20] have proposed a hybrid PSO-K-means algorithm that takes advantage of the Gaussian estimation of the distribution method to empower PSO. Compared with eight classic or state-of-the-art algorithms on five real-world datasets, the proposed method appeared to be superior in robustness and performance.

Bosch has developed an electromechanical brake booster (iBooster) that provides electric vehicles with unprecedented features such as a system of energy management through regenerative braking, adjustable pedal feel, automatic emergency braking (AEB), and so on [21, 22]. The driver of a car equipped with iBooster will be able to manually adjust the braking style to comfort or sport. However, iBoosters can be augmented with the technology that automatically recognizes a driver's driving style by observing his/her interactions with the brake pedal.

3. Concept of model-based synthetic data generation

In this study, the training dataset was generated using a model-based, synthetic data generation technique. In this approach, a phenomenon of interest is not measured directly; instead, an accurate model of the phenomenon is employed to approximate it and generate the desired dataset. This method is helpful when it is difficult to collect rich data, and experimentation is costly or time-consuming. Two closed-loop control models were developed to generate the training dataset for this work, an upper-level model for vehicle longitudinal dynamics and a lower-level dynamic model for brake hydraulics. The block diagram for these closed-loop systems is shown in Fig. 1. These two models will be discussed in detail in the following two subsections.

3.1. Vehicle longitudinal dynamics model

Vehicle longitudinal dynamics include engine behavior analysis, tire tractive force generation, retarding longitudinal forces (i.e., rolling resistance, aerodynamic force, and grade force), and the drivers' gear shifting habits [23-25]. In this paper, vehicle longitudinal dynamics are modeled by considering the simplifying assumption that the road ahead of the vehicle is straight and without slope. Also, the engine model is simplified to a first-order filter.

The vehicle engine is of the TU5 model having the power of 105 hp at 5800 RPM and a maximum torque of 142 N.m. at 4000 RPM. The vehicle model and characteristics and the relevant governing equations will be presented in this subsection. Furthermore, a fuzzy controller that mimics a driver's accelerating/braking habits will be described. At the end of the section, a flowchart that illustrates the interconnection between the different parts of the model will be presented.

3.1.1. Governing equations

An experimental dataset from the engine torque-RPM map of the test car can be used to get the engine torque (T_e) at any specified RPM and throttle opening. T_w can be found from the following equation:

$$T_w = T_e * n_g * n_f, \tag{1}$$

in which n_g is the gear ratios. Besides, n_f is the differential (final drive) gear ratio. Engine torque (T_e) is conveyed by the transmission system to the tires to generate a traction force, which is expressed as:

$$F_t = \left(\frac{n_g * n_f}{r_w}\right) * T_e, \tag{2}$$

where r_w is the wheel effective radius. Vehicle acceleration (a_{veh}) can then be found by using Newton's second law of motion [23]:

$$a_{veh} = \frac{F_{Tot}}{m_{veh}} = \frac{F_t - F_d - F_{rr} - F_b}{m_{veh}}, \tag{3}$$

where m_{veh} is the vehicle mass. $F_d = c\ {v_{veh}}^2$ is the aerodynamic drag force with the overall aero drag coefficient of c. F_{rr} denotes the rolling resistance force that is commonly approximated as a polynomial function of the vehicle speed [23]. Finally, F_b represents the brake force.

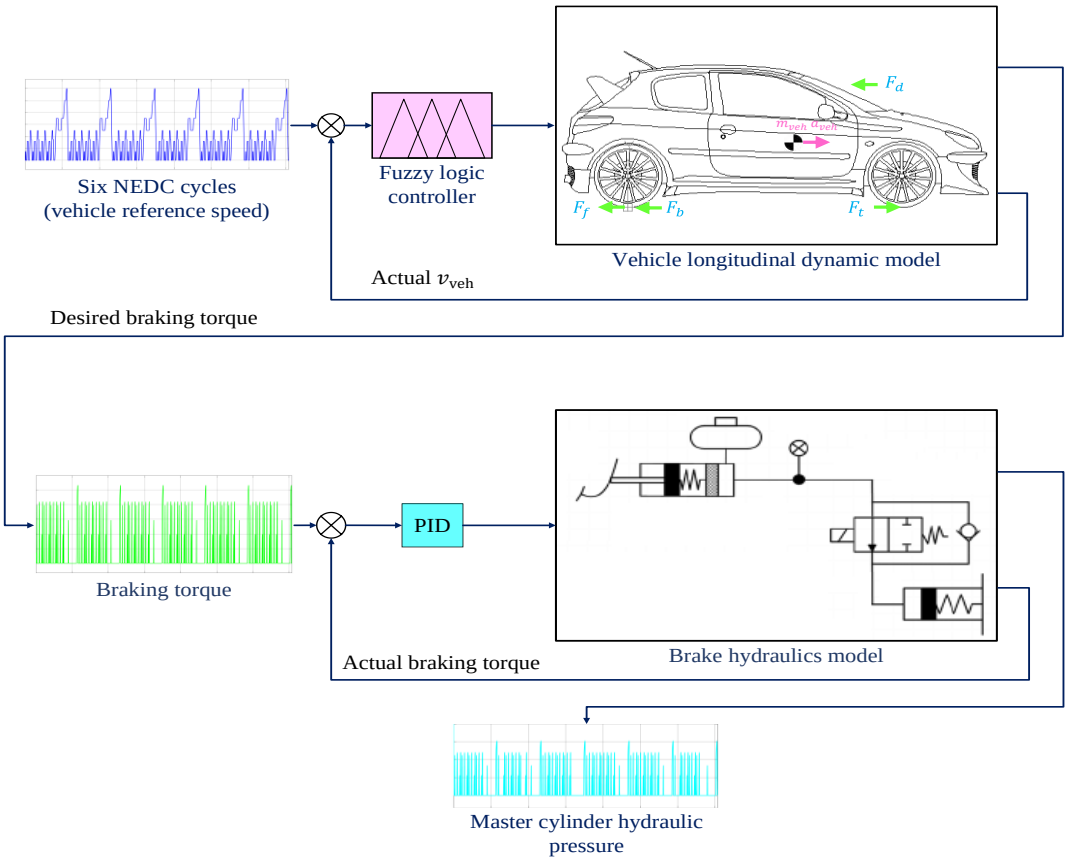

Fig. 1. Block diagram of the model-based synthetic data generation procedure.

All of the mentioned forces act in the opposite direction of the vehicle motion. Obviously, the integration of a_{veh} results in vehicle velocity (v_{veh}). Next, the engine speed (RPM$_e$) can be found using the following relationship:

$$\text{RPM}_e = \frac{30 * v_{veh} * n_f * n_g}{\pi * r_w}. \quad (4)$$

After each gear shift, a 0.5 sec time delay (known as shifting time) is considered in torque conversion to simulate a driver's gear shifting habit more realistically [26]. In other words, the converted torque becomes zero after each gear shift for a period of shifting time. According to this table, the algorithm receives the engine operating point (engine load and RPM) and suggests the required gear number/ratio. For a more elaborate explanation of gear shifting thresholds, readers may refer to [27]. Table 1 and
Table ***2*** tabulate the specifications of the vehicle model and the gearbox, respectively.

Table 1. Vehicle specifications.

Parameter	Value
Vehicle mass	1300 kg
Wheel radius	0.29 m
Front area	2.18 m^2
Aero drag coefficient	0.4
Engine type	TU5 JP4
Engine volume	1587 cc
Engine power	105 hp @ 5800 rpm
Engine maximum torque	142 N.m. @ 4000 rpm
Gearbox	5 speed

Table 2. Comparison of air-core diameter, discharge coefficient, and spray cone angle.

Gear number	Gear ratio (n_g)
1	3.6
2	2.8
3	2
4	1.5
5	1
Final drive (n_f)	4.29

3.1.2. Simulating the driver behavior

A closed-loop control system is required for the vehicle model to track the NEDCs. In this work, two fuzzy logic controllers (FLCs) are used for this purpose. One of the developed FLCs is responsible for applying a traction force to vehicle tires, and the other is in charge of braking. The input to the FLCs is the error between the vehicle speed and the NEDC reference speed.

The two outputs of the FLCs are the throttle pedal angle (when traction is needed) and the braking force. When the sign of the above-mentioned error becomes negative, the FLC will issue a traction force command; otherwise, it will give a braking command when the error sign is positive. An "OR" logical operator is used between the two to avoid any interference between traction and braking commands. For more information on the membership functions of the FLCs, please refer to [28]. The setpoint of the mentioned control system is the vehicle speed in the NEDC cycle. The vehicle speed in the NEDC is compared to the vehicle's current speed, and the resulting error is fed to the FLCs. Then, based on the error sign, the FLCs issue either traction or braking control command. The block diagram of the described vehicle model can be seen in Fig. 2. According to this figure, the yellow (2D) and the blue (1D) look-up tables correspond to part- and full-throttle engine torque-speed maps. These maps have been obtained experimentally through the previous works [27, 28].

Fig. 3 demonstrates the output of the closed-loop controlled vehicle longitudinal dynamics model. As is shown, owing to the performance of FLCs, the NEDC reference speed (the purple dashed line) is well tracked by the vehicle longitudinal speed (the solid black line).

Fig. 3 also illustrates the braking torque (the solid green line), which may be considered as the essence of the vehicle longitudinal dynamics model. Braking torque is used as the reference (desired) value for the lower-level, closed-loop controlled brake hydraulics model, which will be discussed in the next section.

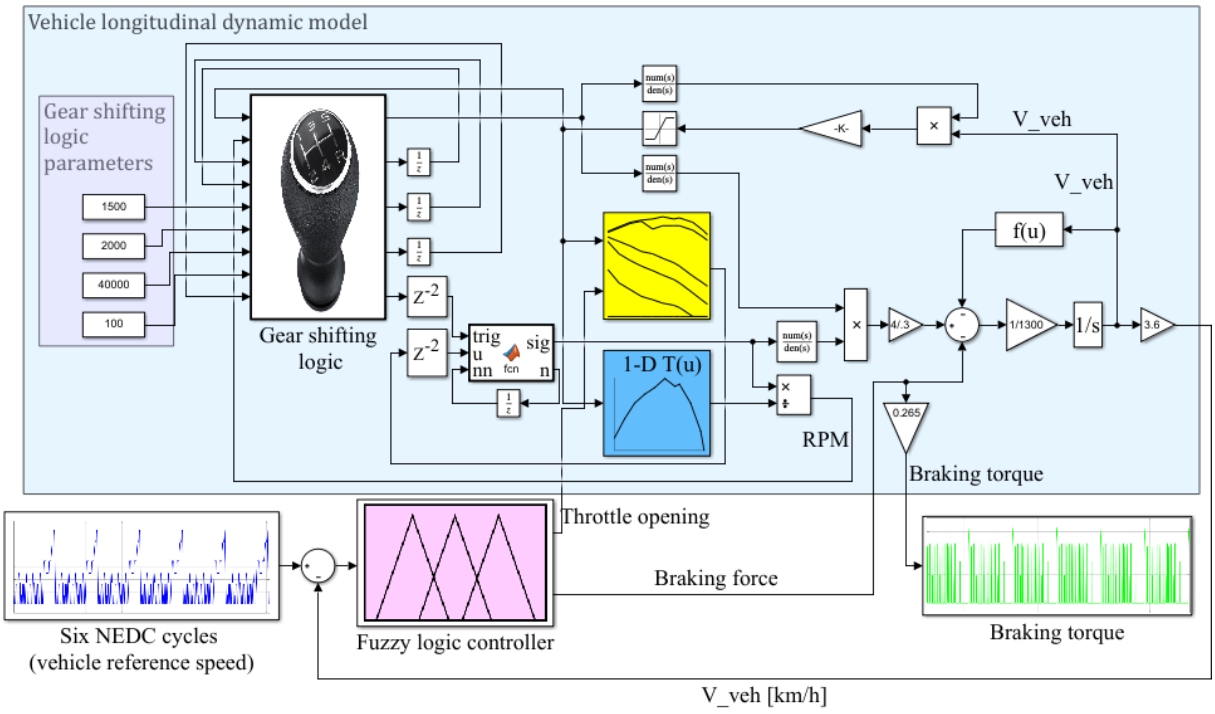

Fig. 2. Block diagram of the closed-loop vehicle longitudinal dynamic model in Simulink.

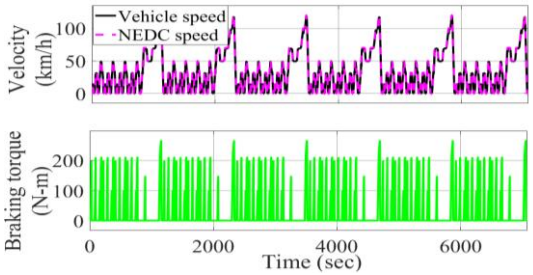

Fig. 3. Tracking the six NEDC cycles and extraction of the braking torque.

3.2. Modeling the brake hydraulics

In the last subsection, a vehicle longitudinal model was set to track six consecutive NEDC cycles, and the associated braking torques were obtained as a result. Remember that the main objective here is to get brake pressure data from the master cylinder as the training dataset. In this regard, the dynamic model of brake hydraulics is used to relate vehicle braking torque to brake master cylinder pressure.

This subsection is dedicated to the dynamic modeling of the closed-loop controlled brake hydraulics and, thus, extracting the hydraulic pressure data from the master cylinder.

In this study, the plant under control is the brake system hydraulics, which can be modeled both statically and dynamically, depending on the application and area of use. For the sake of generating training datasets for data-driven applications, it would be particularly advantageous to model a phenomenon with slow transient dynamics [29, 30]. In this work, although static modeling of the brake system would have sufficed due to the relatively fast transient behavior of the hydraulic oil flow, the dynamic modeling method was adopted so that the hydraulic pressure behavior in the transient regime is not overlooked. Moreover, the dynamic modeling of brake hydraulics becomes vital when designing vehicle stability control systems such as the anti-lock brake system (ABS), electronic stability program (ESP).

The dynamic modeling of brake hydraulics has been exhaustively discussed in the literature [31-35]. The approach is to model the pressure buildup phase in an ABS. Such a system consists of a booster, a tandem master cylinder (TMC), a hydraulic modulator block, four-wheel cylinders (calipers), and the pipelines connecting the modulator to the TMC and the calipers. The overall pressure buildup process is schematically displayed in Fig. 4. Since the performance of the controller is of no concern here and the only thing that matters is the successful tracking of the braking force, we do not need to worry about the controller parameters.

Therefore, the controller gains can be chosen arbitrarily so long as the braking force of the hydraulic model well tracks the considered reference (desired) value. In this study, the controller gains are chosen as $K_p = 100$, $K_i = 50$, and $K_d = 0.1$. The governing equations and the dynamic modeling of brake hydraulics will be discussed in the next section. As can be seen, the booster magnifies the pedal force and applies it to the master cylinder (MC), causing its piston to move.

Owing to the fluid inertia, the displacement of the piston pressurizes the oil in the master cylinder. The difference between oil pressures in the MC and the MC2SCV pipeline generates an oil flow of Q_{mc} from the MC into the MC2SCV pipeline. Because of the fluid inertia, the difference between the pressures of oil flows entering and exiting the MC2SCV pipeline increases the fluid pressure. This process continues until the oil in the caliper is pressurized. The pressurized oil in the caliper forces the wheel cylinder piston to move forward and tightly push the brake pad against the brake disc (rotor), thereby generating a clamping force at the interface of the pad and disc.

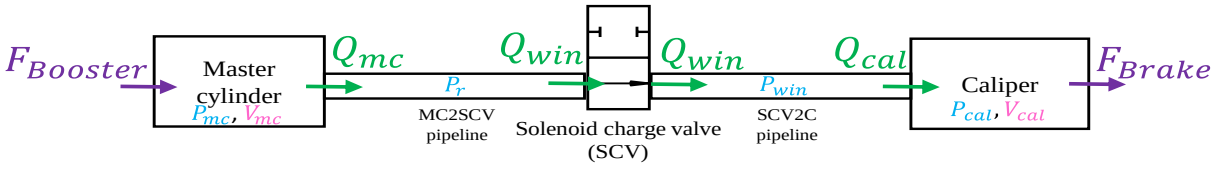

Fig. 4. Schematic view of the pressure build-up phase in the brake hydraulic system.

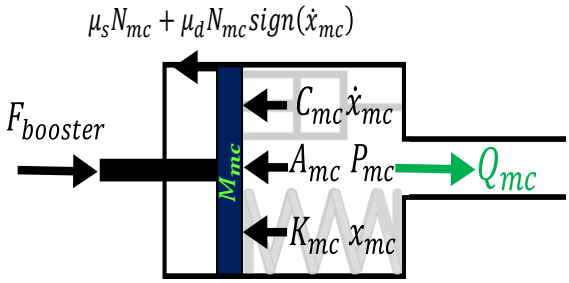

Fig. 5. Free body diagram of the master cylinder.

The governing equations of the explained process will be given next. After the brake booster, the next component to be modeled is the MC, responsible for conveying pressurized oil through the pipelines. Fig. 5 depicts the free body diagram (FBD) of the MC and the forces exerted on the MC piston.

The dynamic movement of the MC can be modeled as a one-degree of freedom (DOF), spring-mass-damper system.

Fig. 6 delineates the closed-loop controlled brake hydraulics, dynamic model. As is observed, a proportional-integral-derivative (PID) controller is responsible for getting the actual braking torque to the desired value obtained from the longitudinal vehicle dynamics model.

$$\begin{aligned} M_{mc}\ddot{x}_{mc} = {} & F_{\text{Booster}} - P_{mc}A_{mc} \\ & - K_{mc}x_{mc} - C_{mc}\dot{x}_{mc} \\ & - \mu_s N_{mc} \\ & - \mu_d N_{mc} tanh(\dot{x}_{mc}), \end{aligned} \tag{5}$$

Newton's second law of motion results in:

where M_{mc}, K_{mc}, and C_{mc} are the MC piston mass, spring stiffness, and damping coefficient, respectively.

P_{mc} and A_{mc} denote, respectively, the MC pressure and cross-sectional area. x_{mc}, $\dot{x}_{mc}$, and $\ddot{x}_{mc}$ stand for the MC piston displacement, velocity, and acceleration, respectively. $\mu_s N_{mc}$ and $\mu_d N_{mc} sign(\dot{x}_{mc})$ represent the static and the dynamic friction forces applied to the MC piston, respectively. Note that in the governing equations, $tanh(\cdot)$ is used instead of the $sgn(\cdot)$ function; because $sgn(\cdot)$ is a static function that adversely affects and hinders the dynamic simulation.

The MC pressure in the above equation can be obtained by integrating its rate of change ($\dot{P}_{mc}$), as follows [36]:

$$\begin{aligned} P_{mc} &= \int \beta_{mc} \frac{(A_{mc}\dot{x}_{mc} - Q_{mc})}{V_{mc}} dt, \\ V_{mc} &= A_{mc}(x_{max} - x_{mc}), \end{aligned} \tag{6}$$

where β_{mc} and V_{mc} are the bulk modulus and the volume of the MC chamber, respectively. Also, Q_{mc} denotes the oil flow leaving the MC and entering the pipeline, and it can be calculated as [36]:

$$\begin{aligned} Q_{mc} = {} & C_{mc}C_d A_{0mc} \sqrt{\frac{2}{\rho}|P_{mc} - P_r|} \\ & * tanh(P_{mc} - P_r), \end{aligned} \tag{7}$$

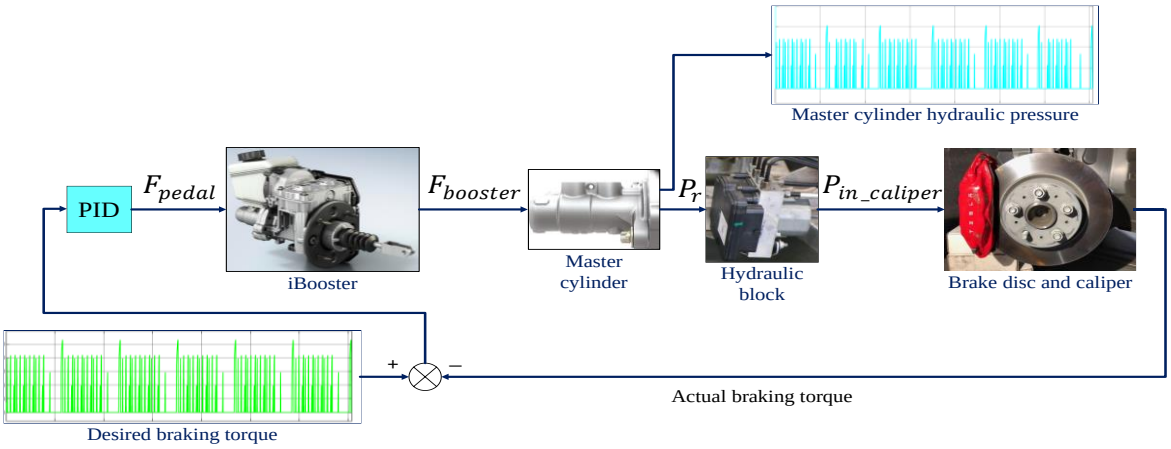

Fig. 6. Closed-loop controlled brake hydraulics and dynamic model.

where $C_{mc}C_d$ is the MC orifice discharge coefficient, and A_{0mc} is the MC outlet cross-sectional area. ρ is the brake oil density, and P_r is the oil pressure in the MC2SCV pipeline, which can be obtained by integrating its rate of change ($\dot{P}_r$):

$$P_r = \int \beta_r \frac{(Q_{mc} - Q_{win})}{V_r} dt, \tag{8}$$

In the above equation, β_r and V_r are the bulk modulus and the volume of the MC2SCV pipeline, respectively. Also, the oil flow Q_{win} leaving the MC2SCV pipeline can be obtained as:

$$Q_{win} = C_{win}C_d A_{0win} \sqrt{\frac{2}{\rho}|P_r - P_{win}|} * tanh(P_r - P_{win}), \tag{9}$$

where P_{win} is the pressure of oil flowing between the solenoid charge valve (SCV) and the caliper. $C_{win}C_d$ and A_{0win} represent the SCV discharge coefficient and the SCV outlet cross-sectional area, respectively. P_{win} can be determined using the following equation:

$$P_{win} = \int \beta_{win} \frac{(Q_{win} - Q_{cal})}{V_{win}} dt, \tag{10}$$

In Eq. (10), β_{win} and V_{win} are the bulk modulus and the volume of the SCV2C pipeline, respectively. Besides, Q_{cal} denotes the oil flow into the caliper from the SCV2C pipeline, and it is obtained as

$$Q_{cal} = C_{cal}C_d A_{0cal} \sqrt{\frac{2}{\rho}|P_{win} - P_{cal}|} * tanh(P_{win} - P_{cal}), \tag{11}$$

where $C_{cal}C_d$ and A_{0cal} stand for the caliper inlet discharge coefficient and cross-sectional area, respectively. Also, P_{cal} is the oil pressure in the caliper, and it can be calculated from the following equation:

$$P_{cal} = \int \beta_{cal} \frac{(Q_{cal} - A_{cal}\dot{x}_{cal})}{V_{cal}} dt, \tag{12}$$

$$V_{cal} = A_{cal}x_{cal},$$

In the above equation, β_{cal} and V_{cal} are the bulk modulus and the volume of the caliper, respectively. A_{cal} is the caliper cross-sectional area, and $\dot{x}_{cal}$ and x_{cal} denote the caliper piston velocity and displacement, respectively. According to Eq. (12), to get the oil pressure in the caliper, $\dot{x}_{cal}$ and x_{cal} must be known and obtained. Like the MC, the brake caliper is commonly modeled as a one-DOF, spring-mass-damper system. Therefore, by applying Newton's second law of motion to the caliper FBD, the brake clamping force is determined as:

$$\begin{aligned} M_{cal}\ddot{x}_{cal} = {} & P_{cal}A_{cal} \\ & - K_{cal}(x_{cal} - x_{gap}) \\ & - C_{cal}\dot{x}_{cal} - \mu_s N_{cal} \\ & - \mu_d N_{cal} tanh(\dot{x}_{cal}), \end{aligned} \tag{13}$$

where M_{cal}, K_{cal}, and C_{cal} represent, respectively, the brake pad mass, stiffness, and damping coefficient. $\mu_s N_{cal}$ and $\mu_d N_{cal} tanh(\dot{x}_{cal})$ indicate the static and dynamic friction forces applied to the pad, respectively. $\dot{x}_{cal}$ and x_{cal} are calculated from Eq. (13) and substituted into Eq. (12) to obtain P_{cal}, which will be used again in Eq. (13). Finally, the brake clamping force ($P_{cal}A_{cal}$) can be determined from Eq. (13).

When using a model-based, synthetic data generation scheme, the dynamic models are run by software programs such as MATLAB/ Simulink. The corresponding solver often keeps step sizes small enough to maintain a high solution accuracy, which leads to the generation of excessive redundant data to be eliminated to reduce the data volume.

ode23s is the selected type of solver used to solve stiff differential equations based on a modified Rosenbrock formula of order 2. In addition, the variable step size is selected for the numerical solution of ordinary differential equations. The reason for this is to control the errors of the method and to ensure stability properties. The final time of the simulation is set to 7080 (1180*6), which is equal to the time that six consecutive NEDC cycles would take.

4. Optimal K-means clustering

Clustering refers to the idea of dividing unlabeled data samples into some clusters, such that the samples in the same cluster are distributed as densely as possible and those in different clusters as sparsely as possible.

4.1. Clustering validity indexes

The clustering validity index (CVI) is a quantitative measure of how well a given dataset is clustered. In other words, it provides a quantitative evaluation of the discrimination ability of a given clustering scheme. Ideally, this index is responsible for maintaining *cohesion* within-cluster data points and *separation* between cluster centers. This article has used the Davies-Bouldin (DB) and Chou-Su (CS) clustering validity indices. Generally, CVIs are represented by ratios in which the numerator indicates the scatter of data points within a cluster, and the denominator specifies the separation between cluster centers. Thus, from an optimization standpoint, each CVI can be thought of as a cost function, which may minimize the desired level of cohesion among data points and the optimal separation distance between cluster centers. The scatter of data points within the i^{th} cluster ($S_{i,q}$) and the distance between the i^{th} and the j^{th} cluster centers ($d_{ij,t}$) are determined as:

$$S_{i,q} = \sqrt[q]{\frac{1}{N_i}\sum_{\vec{X}\in C_i}\left\|\vec{X}-\vec{m}_i\right\|_2^{\,q}}, \tag{14}$$

$$d_{ij,t} = \sqrt[t]{\sum_{p=1}^{d}\left|m_{i,p}-m_{j,p}\right|^t} = \left\|\vec{m}_i-\vec{m}_j\right\|_t, \tag{15}$$

where $\vec{m}_i$ denotes the i^{th} cluster center; $q,t \geq 1$. The integer q specifies the norm type, and $q = 2$ (or the Euclidean norm) is used in this study. N_i represents the number of data samples in the i^{th} cluster C_i. The DB criterion is defined as:

$$\mathrm{DB} = \frac{1}{K}\sum_{i=1}^{K} R_{i,qt}\,, \tag{16}$$

where

$$R_{i,qt} = \max_{j\in K, j\neq i}\left\{\frac{S_{i,q}+S_{j,q}}{d_{ij,t}}\right\}. \tag{17}$$

Fig. 7 delineates the flowchart of the APTFPSO- based optimal clustering algorithm with the DB validity index for finding an initial, random solution with five cluster centroids. As can be seen in Fig. 7, first, the setting parameters of the algorithm are defined. The next step is initialization, where random solutions are produced. The cost function is then evaluated with respect to the generated solutions, and personal and global best solutions are specified. The fuzzy inference system (FIS) calculates the personal and global learning coefficients based on the particles' fitness vectors and a number of iterations. The particles' positions are updated and, then, limited.

Finally, the algorithm goes to the cost function evaluation phase until the termination criterion (reaching a maximum number of iterations of 100) is met. In addition to the DB clustering validity index, Chou et al. [37] introduced the following clustering validity index written in Eq. (18).

$$\mathrm{CS} = \frac{\sum_{i=1}^{K}\left[\frac{1}{N_i}\sum_{\vec{X}_i\in C_i}\max_{\vec{X}_q\in C_i}\left\{d\left(\vec{X}_i,\vec{X}_q\right)\right\}\right]}{\sum_{i=1}^{K}\left[\min_{j\in K, j\neq i}\left\{d\left(\vec{m}_i,\vec{m}_q\right)\right\}\right]} \tag{18}$$

In this equation, $\vec{m}_i = \frac{1}{N_i}\sum_{x_j\in C_i}\vec{x}_j$ is the mean value of the data within the i^{th} cluster C_i. Additionally, $d\left(\vec{X}_i,\vec{X}_q\right)$ denotes the distance matrix between data points $\vec{X}_i$ and $\vec{X}_q$. It can be seen that the minimization of both the DB and CS validity indices results in a lower datapoint scatter within the cluster and a greater separation distance between cluster centers.

Chou et al.'s findings [37] revealed that the CS is more efficient than other validity indices, albeit at a higher computational cost [11, 38].

4.2. APT-FPSO-based optimal clustering

The adaptive particularly tunable fuzzy particle swarm optimization (APT-FPSO) algorithm is a novel variant of the PSO algorithm with improved exploitation ability [39]. The algorithm takes advantage of fuzzy membership functions to individually update, at each iteration, the global and personal learning coefficients [12].

The parameters of the APT-FPSO algorithm are set as follows. The number of population individuals and the maximum number of iterations are both set to 100. The initial inertia weight of particles is equal to one, but it is multiplied by a damping factor of 0.99 at the end of each iteration. For more detailed information on metaheuristic clustering, please refer to [40].

5. Results and discussion

In this work, there are n (= 5115), D (= 2) dimensional data points, and the maximum number of clusters $k = 10$. Thus, the position of each particle would constitute a $k + k \times d = 30$ dimensional vector. The first k is a floating number in the range of [0, 1], and $k \times d$ corresponds to k cluster centers. In other words, the position of each particle is a 10×3 matrix in which the elements of the third column are responsible for the activation of their corresponding row (cluster centroids). This process is conducted so that for each element above 0.5, a corresponding centroid is activated, and vice versa [11].

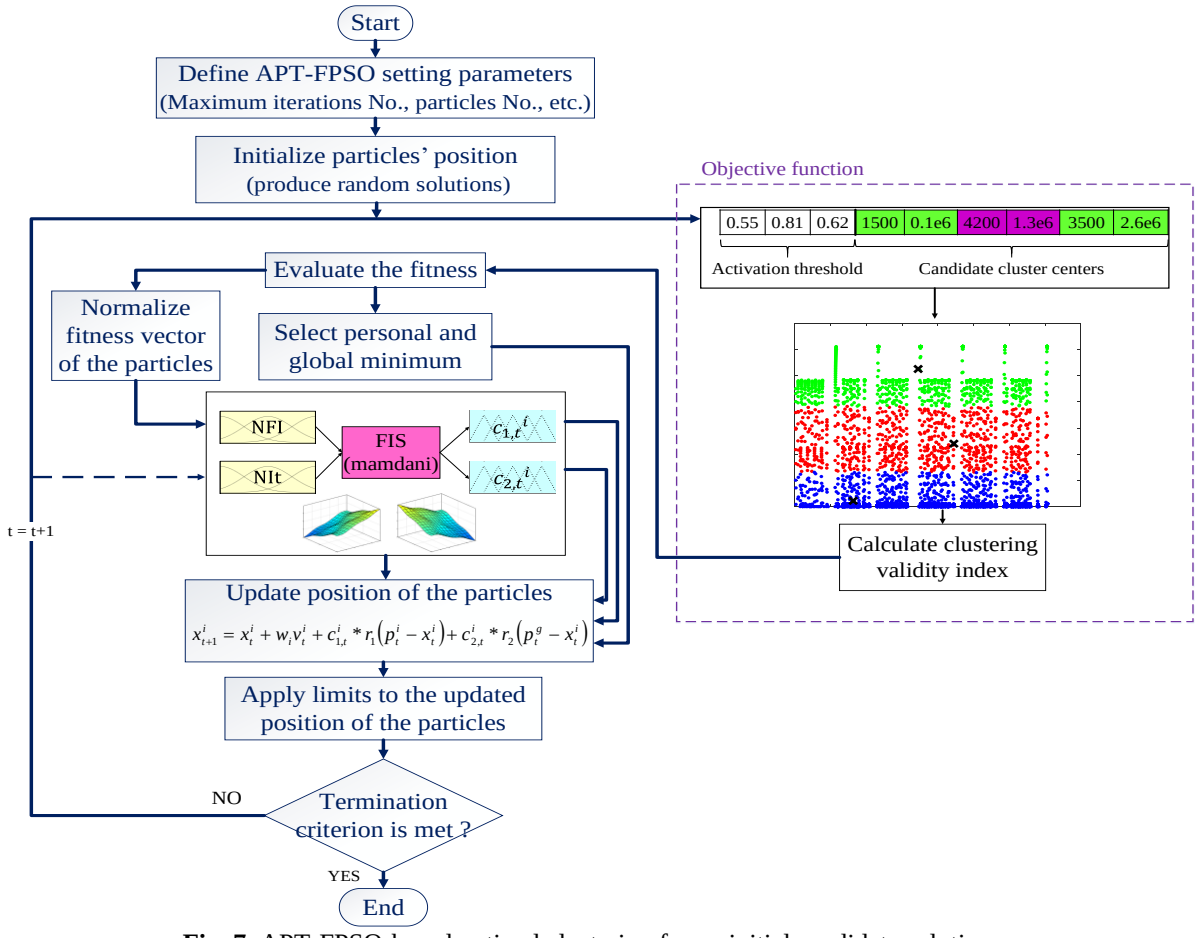

Fig. 7. APT-FPSO-based optimal clustering for an initial candidate solution.

The subsequent computations were performed on a laptop computer with Intel® Core™2 Duo CPU @ 2.50 GHz and 12.00 GB of RAM. It took 16.071 and 6602.073 seconds of run time to execute the algorithm with the DB and the CS validity indexes, respectively. Ezugwu has pointed out that it is better to judge metaheuristic clustering algorithms based on the number of fitness functions they evaluate rather than the number of iterations [10]. Table 3 lists the global best position found by the APT-FPSO algorithm with the CS validity index, at a global cost of 0.5597. Two cluster centroids with coordinates of [0.0047E6, 3.0597E6] and [0.0059E6, 3.0288E6] are found by this algorithm. Accordingly, this position is a 10×3 matrix in which the first two columns are cluster centers, and the third column includes the activation values for each row (cluster centers). The activation thresholds above 0.5 are bolded, and their corresponding cluster centers are activated.

Fig. 8 depicts the convergence of global best cost values versus the number of objective functions with the CS index. As is observed, the algorithm converges to the global cost of 0.5597 after evaluating 10100 times of the objective function.

Fig. 9 demonstrates that the obtained dataset can be best clustered into two intensive and moderate braking clusters.

However, it is obvious from the figure that the algorithm with the CS validity index has performed poorly and failed to cluster the dataset correctly. Most likely, this is due to the high sensitivity of automatic clustering to the initial positions of cluster centroids, which makes the optimization algorithm prone to being trapped in local minima [41]. Repeated executions of the APT-FPSO-based algorithm with both the CS and DB validity indices have revealed that, for this problem, the above-mentioned flaw can be rectified by using the algorithm with the DB index.

Similarly,

Table ***4*** tabulates the global best position found by the APT-FPSO-based optimal clustering algorithm with the DB validity index. The moderate and intensive braking cluster centroids are positioned at [3487, 0.0979E6] and [3526, 2.485E6], respectively.

Fig. 10 shows the convergence diagram of global best cost values versus the number of objective functions evaluated by the APT-FPSO clustering algorithm with the DB validity index.

Table 3. Global best position found by APT-FPSO algorithm with CS validity index and fitness value of 0.559710978219709.

Cluster centroids (*1e6)		Activation threshold
0. 001476730455	1. 01798905886	0. 140745
0. 001956470687	2. 04326686975	0. 382383
0. 005405112725	1. 63246475301	0. 484018
0. 003156021945	2. 78071619015	0. 358828
0. 004001458180	1. 42496190681	0. 224604
0. 004716198794	**3. 05973551743**	**0. 556690**
0. 004451404928	0. 72963973787	0. 407020
0. 003316182008	0. 63138054738	0. 354769
0. 002873009563	1. 69800830197	0. 235816
0. 005918633118	**3. 02880037165**	**0. 779897**

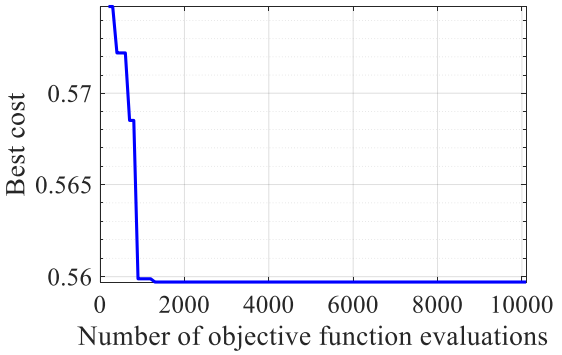

Fig. 8. Convergence diagram of the APT-FPSO-based optimal clustering algorithm with CS validity index.

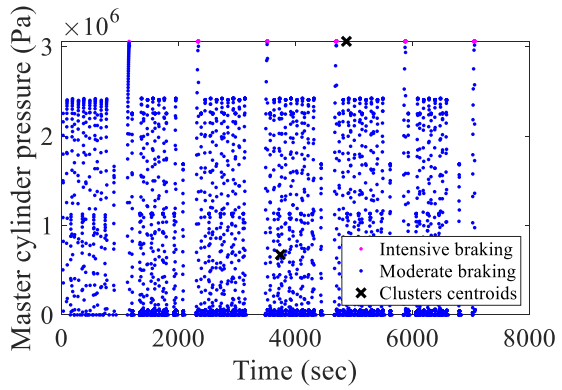

Fig. 9. Final results of the APT-FPSO-based optimal clustering algorithm with CS validity index.

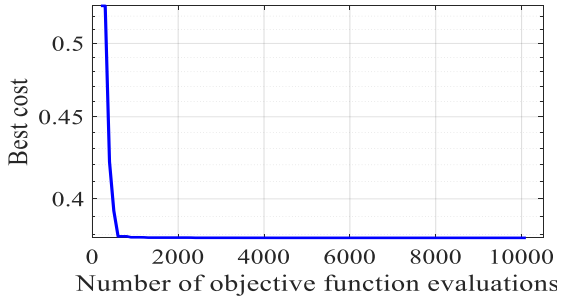

Fig. 10. Convergence diagram of the APT-FPSO-based optimal clustering algorithm with DB validity index.

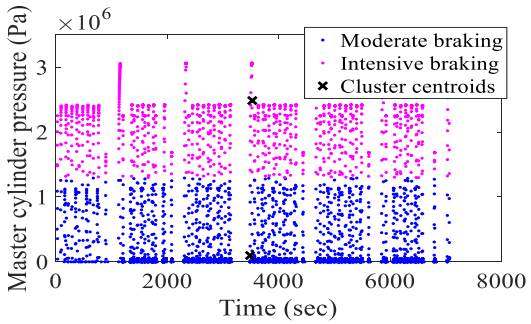

Fig. 11. Final results of the APT-FPSO-based optimal clustering algorithm with DB validity index.

Table 4. Global best position found by APT-FPSO algorithm with DB validity index and fitness value of 0.378236535796813.

Cluster centroids (*1e6)		Activation threshold
0.0041502239	1.20441068818	0.118583
0.0021449900	1.78058173302	0.257461
0.0030604892	2.12802088065	0.268538
0.0035961987	1.32368268307	0.414962
0.0034868318	**0.09790430184**	**0.703775**
0.0055365881	0.97742164925	0.415209
0.0047015273	1.81183429066	0.226196
0.0035263679	**2.48543309092**	**0.687508**
0.0020497855	2.07372070641	0.384973
0.0055383731	0.79104425001	0.377243

As can be seen, the algorithm converges to the global cost of 0.3782 after evaluating 10100 times the objective function.

Fig. 11 demonstrates a great improvement in the performance of the clustering algorithm with the DB validity index. The dataset is partitioned into two clusters with the centers given in Table ***4***.

As for future research works in line with the current topic, and for the sake of generalization, this research could be replicated based on other driving cycles. In that case, a primary driving cycle recognition system may have to be used first to detect the driving cycle, and then a driver behavior recognizer would be exploited. Moreover, applications of the internet of things [42, 43], cloud computing [44], and vehicle communication [45, 46] could be investigated to facilitate the acquisition and assessment of large-scale brake pressure data generated while driving the vehicle.

6. Conclusions

In this paper, an optimal K-means clustering strategy was proposed for recognizing a driver's braking behavior in a vehicle subjected to the NEDCs. The brake pressure data were generated synthetically with the help of two closed-loop controlled dynamic models: one for the higher-level vehicle longitudinal motion and the other for the lower-level brake hydraulics. APT-FPSO metaheuristic algorithm was used to find the optimal number of clusters and the associated cluster centers. It was observed that considering the DB validity index, the optimization algorithm performs well and clusters the brake pressure data into two clusters of moderate and intensive braking. The results of this research could be reliably used in the development of electromechanical brake boosters and intelligent driver assistance systems. For instance, such a system may be used in iBoosters from Bosch to automatically adjust the driver's pedal feel based on his/her treatment with the brake pedal.

References

[1] H. R. Eftekhari and M. Ghatee, "A similarity-based neuro-fuzzy modeling for driving behavior recognition applying fusion of smartphone sensors", *J. Intell. Transp. Syst.*, Vol. 23, No. 1, pp. 72-83, (2019).

[2] C. Lu, F. Hu, D. Cao, J. Gong, Y. Xing and Z. Li, "Virtual-to-real knowledge transfer for driving behavior recognition: Framework and a case study", *IEEE Trans. Veh. Technol.*, Vol. 68, No. 7, pp. 6391-6402, (2019).

[3] S. Jia, F. Hui, S. Li, X. Zhao and A. J. Khattak, "Long short-term memory and convolutional neural network for

abnormal driving behaviour recognition", *IET Intell. Transp. Syst.*, Vol. 14, No. 5, pp. 306-312, (2019).

[4] J. Zhang, Z. Wu, F. Li, C. Xie, T. Ren, J. Chen and L. Liu, "A deep learning framework for driving behavior identification on in-vehicle CAN-BUS sensor data", *Sensors*, Vol. 19, No. 6, p. 1356, (2019).

[5] Y. Xing, C. Lv, H. Wang, H. Wang, Y. Ai, D. Cao, E. Velenis and F. Y. Wang, "Driver lane change intention inference for intelligent vehicles: framework, survey, and challenges", *IEEE Trans. Veh. Technol.*, Vol. 68, No. 5, pp. 4377-4390, (2019).

[6] Y. Xing, C. Lv, H. Wang, D. Cao, E. Velenis and F. Y. Wang, "Driver activity recognition for intelligent vehicles: a deep learning approach", *IEEE Trans. Veh. Technol.*, Vol. 68, No. 6, pp. 5379-5390, (2019).

[7] W. Bi, M. Cai, M. Liu and G. Li, "A big data clustering algorithm for mitigating the risk of customer churn", *IEEE Trans. Industr. Inform.*, Vol. 12, No. 3, pp. 1270-1281, (2016).

[8] K. Krishna and M. Narasimha Murty, "Genetic k-means algorithm", *IEEE Trans. Syst. Man Cybern. Syst., part B (cybernetics)*, Vol. 29, No. 3, pp. 433-439, (1999).

[9] D. Aloise, A. Deshpande, P. Hansen and P. Popat, "NP-hardness of Euclidean sum-of-squares clustering", *Mach. Learn.*, Vol. 75, No. 2, pp. 245-248, (2009).

[10] A. E. Ezugwu, "Nature-inspired metaheuristic techniques for automatic clustering: a survey and performance study", *SN Appl. Sci.*, Vol. 2, No. 2, p. 273, (2020).

[11] S. Das, A. Abraham and A. Konar, "Automatic clustering using an improved differential evolution algorithm", *IEEE Trans. Syst. Man Cybern. Syst.*, Vol. 38, No. 1, pp. 218-237, (2007).

[12] N. Bakhshinezhad, S. A. Mir Mohammad Sadeghi, A. R. Fathi and H. R. Mohammadi Daniali, "Adaptive particularly tunable fuzzy particle swarm optimization algorithm", *Iran. J. Fuzzy Syst.*, Vol. 17, No. 1, pp. 65-75, (2020).

[13] W. Wang, J. Xi and D. Zhao, "Learning and inferring a driver's braking action in car-following scenarios", *IEEE Trans. Veh. Technol.*, Vol. 67, No. 5, pp. 3887-3899, (2018).

[14] Q. Guo, Z. Zhao, P. Shen, X. Zhan and J. Li, "Adaptive optimal control based on driving style recognition for plug-in hybrid electric vehicle", *Energy*, Vol. 186, p. 115824, (2019).

[15] C. Lv, Y. Xing, C. Lu, Y. Liu, H. Guo, H. Gao and D. Cao, "Hybrid-learning-based classification and quantitative inference of driver braking intensity of an electrified vehicle", *IEEE Trans. Veh. Technol.*, Vol. 67, No. 7, pp. 5718-5729, (2018).

[16] K. Nikzadfar and A. H. Shamekhi, "Investigating a new model-based calibration procedure for optimizing the emissions and performance of a turbocharged diesel engine", *Fuel*, Vol. 242, pp. 455-469, (2019).

[17] K. Nikzadfar and A. H. Shamekhi, "An extended mean value model (EMVM) for control-oriented modeling of diesel engines transient performance and emissions", *Fuel*, Vol. 154, pp. 275-292, (2015).

[18] K. Nikzadfar and A. H. Shamekhi, "Investigating the relative contribution of operational parameters on performance and emissions of a common-rail diesel engine using neural network", *Fuel*, Vol. 125, pp. 116-128, (2014).

[19] K. Nikzadfar and A. H. Shamekhi, "Development of a hierarchical observer for burned gas fraction in inlet manifold of a turbocharged diesel engine", *IEEE Trans. Veh. Technol.*, Vol. 67, No. 12, pp. 11500-11510, (2018).

[20] H. Gao, Y. Li, P. Kabalyants, H. Xu and R. Martinez-Bejar, "A novel hybrid PSO-k-means clustering algorithm using gaussian estimation of distribution method and lévy flight", *IEEE access*, Vol. 8, pp. 122848-122863, (2020).

[21] A. Kunz, M. Kunz, H. Vollert and M. Förster, "Electromechanical brake booster

for all drive concepts and automated driving", *ATZ Worldw.*, Vol. 120, No. 4, pp. 58-61, (2018).

[22] T. Leiber, H. Leiber and A. van Zanten, "Brake boosters for automated driving", *ATZ Worldw.*, Vol. 121, No. 3, pp. 48-53, (2019).

[23] D. Crolla and B. Mashadi, *Vehicle powertrain systems*, John wiley & sons Inc., (2011).

[24] R. T. Sangeetha, V. Shankar, A. Bose and B. Jayaraman, "A unique approach to optimize the gear-shift map of a compact SUV to improve FE and performance", *SAE Tech. Pap.*, 2020-01-0969, (2020).

[25] L. Paulraj, S. Muthiah and S. Chidhanand, "Gear shift pattern optimization for best fuel economy, performance and emissions", *SAE Tech. Pap.*, 2020-01-1280, (2020).

[26] G. Lucente, M. Montanari and C. Rossi, "Modelling of an automated manual transmission system", *Mechatronics*, Vol. 17, No. 2-3, pp. 73-91, (2007).

[27] K. Nikzadfar, N. Bakhshinezhad S. A. MirMohammadSadeghi, H. T. Ledari, and A. Fathi, "An optimal gear shifting strategy for minimizing fuel consumption based on engine optimum operation line", *SAE Tech. Pap.*, 2019-01-5055, (2019).

[28] S. A. MirMohammad Sadeghi, K. Nikzadfar, N. Bakhshinezhad and A. Fathi, "Optimal idle speed control of a natural aspirated gasoline engine using bio-inspired meta-heuristic algorithms", *Automot. Sci. Eng.*, Vol. 8, No. 3, pp. 2792-2806, (2018).

[29] S. A. Mir Mohammad Sadeghi, S. F. Hoseini, A. Fathi and H. Mohammadi Daniali, "Experimental hysteresis identification and micro-position control of a shape-memory-alloy rod actuator", *Int. J. Eng.*, Vol. 32, No. 1, pp. 71-77, (2019).

[30] S. F. Hoseini, S.A. MirMohammad Sadeghi, A. Fathi and H.M. Daniali, "Adaptive predictive control of a novel shape memory alloy rod actuator", *Proc. Inst. Mech. Eng. I.*, Vol. 235, No. 3, pp. 291-301, (2021).

[31] M. Maghroory, A. Farhadi and P. Naderi, "Hydraulic anti-lock and anti-skid braking system using fuzzy controller", *Jour. of comput. Appl. Res. Mech. Eng.*, Vol. 6, No. 1, pp. 21–37, (2016).

[32] J. C. Gerdes and J. K. Hedrick, "Brake system modeling for simulation and control", *J. Dyn. Syst. Meas. Control*, Vol. 121, pp. 496-503, (1999).

[33] C. Lv, J. Zhang, Y. Li, D. Sun and Y. Yuan, "Hardware-in-the-loop simulation of pressure-difference-limiting modulation of the hydraulic brake for regenerative braking control of electric vehicles", *Proc. Inst. Mech. Eng. D.*, Vol. 228, No. 6, pp. 649-662, (2014).

[34] J. Zhang, C. Lv, J. Gou and D. Kong, "Cooperative control of regenerative braking and hydraulic braking of an electrified passenger car", *Proc. Inst. Mech. Eng. D.*, Vol. 226, No. 10, pp. 1289-1302, (2012).

[35] B. Moaveni and P. Barkhordari, "Modeling, identification, and controller design for hydraulic anti-slip braking system", *Proc. Inst. Mech. Eng. D.*, Vol. 233, No. 4, pp. 862-876, (2019).

[36] H. E. Merritt, *Hydraulic Control Systems*, John wiley & sons Inc., New York, (1967).

[37] C. H. Chou, M. C. Su, E. Lai, "A new cluster validity measure for clusters with different densities", *IASTED Int. Conf. Intell. Syst. Control*, pp. 276-281, (2003).

[38] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization", *IEEE Trans. Evol. Comput.*, Vol. 1, No. 1, pp. 67-82, (1997).

[39] S. A. Mirmohammad Sadeghi, N. Bakhshinezhad, A. Fathi and H. M. Daniali, "An Optimal Defect-free Synthesis of Four-bar Mechanisms by Using Constrained APT-FPSO Algorithm", *J. Comput. Robot. (JCR)*, Vol. 12, No. 2, pp. 39-48, (2019).

[40] S. Das, A. Abraham and A. Konar, *Metaheuristic clustering*, Springer, Vol. 178, (2009).

[41] M. E. Celebi, H. A. Kingravi and P. A. Vela, "A comparative study of efficient initialization methods for the k-means

clustering algorithm", *Expert Syst. Appl.*, Vol. 40, No. 1, pp. 200-210, (2013).
[42] A. Papacharalampopoulos, C. Giannoulis, P. Stavropoulos and D. Mourtzis, "A digital twin for automated root – cause search of production alarms based on KPIs aggregated from IoT", *Appl. Sci.*, Vol. 10, No. 7, p. 2377, (2020).
[43] G. Mehta, M. Singh, S. Dubey and Y. Mishra, "Design of Auto-Braking System for Accident Prevention and Accident Detection System Using IoT", *Smart Sens. Industr. Internet of Things*, pp. 101-114, (2021).
[44] S. S. Gill, S. Tuli, M. Xu, I. Singh, K.V. Singh, D. Lindsay, S. Tuli, D. Smirnova, M. Singh, U. Jain and H. Pervaiz, "Transformative effects of IoT, Blockchain and Artificial Intelligence on cloud computing: Evolution, vision, trends and open challenges", *Internet Things (IOT)*, Vol. 8, p. 100118, (2019).
[45] M. A. Rahim, M. A. Rahman, M. M. Rahman, A. T. Asyhari, M. Z. A. Bhuiyan and D. Ramasamy, "Evolution of IoT-enabled connectivity and applications in automotive industry: A review", *Veh. Commun.*, Vol. 27, p. 100285, (2020).
[46] L. Athanasopoulou, A. Papacharalampopoulos, P. Stavropoulos and D. Mourtzis, "Design and manufacturing of a smart mobility platform's context awareness and path planning module: A PSS approach", *Procedia Manufacturing*, Vol. 51, pp. 61-66, (2020).

How to cite this paper:

Ali Mirmohammad Sadeghi, Abdollah Amirkhani and Behrooz Mashadi, "Braking intensity recognition with optimal K-means clustering algorithm,", *J. Comput. Appl. Res. Mech. Eng.*, Vol. 11, No. 2, pp. 409-423, (2022).

DOI: 10.22061/JCARME.2021.7412.1987

URL: https://jcarme.sru.ac.ir/?_action=showPDF&article=1577