Practical learning of DSP-based motor control for engineering students

Lorsch, A., Schmilinsky, M., Vagapov, Y., Bolam, R.C. and Dianov, A.

Practical Learning of DSP-based Motor Control for Engineering Students

Andreas Lörsch
Wrexham University
Wrexham, UK

Max Schmilinsky
Wrexham University
Wrexham, UK

Yuriy Vagapov
Wrexham University
Wrexham, UK

Robert Cameron Bolam
Wrexham University
Wrexham, UK

Anton Dianov
Wenzhou Jiangcheng Automotive Parts Co., Ltd
Wenzhou, China

***Abstract*—This paper discusses a practical approach to learning the basic principles of control engineering using DSP-based motor control. Laboratory work utilising microcontrollers for signal processing usually requires students the knowledge of coding for embedded software programming. However, a variety of visual programming software offered on the market provides assistance in avoiding low-level software programming for microcontrollers. This feature can be useful for students not specialising in electrical engineering, who have to understand the basic operation of MCUs and do not need to study them in detail. In order to focus the attention of those students only on the main MCU features, this paper presents an example of embedded software developed using visual programming in the Matlab/Simulink environment for studying DSP-based motor control. This approach can be implemented in the laboratory curriculum for non-electrical/electronic students to provide effective learning of the principles of motor control and basic control engineering. The educational laboratory setup consists of Texas Instrument hardware and a low-power 3-phase BLDC motor.**

Keywords—microcontroller, motor control, Matlab/Simulink, BLDC motor, encoder

I. Introduction

Practical assignments, commonly known as laboratory work, are an important component of modern engineering education at the university and college levels. These hands-on activities play a significant role in reinforcing the students' understanding of the engineering principles learnt through lectures and tutorials. Laboratory works, organised usually as teamwork, provide an opportunity for the students to develop communication and coordination skills essential for professional activity in the area of engineering [1]-[3]. Laboratory works are widely used to study various engineering subjects including control engineering and particularly motor control and electric drives [4].

Electric motors are extensively utilised in a large variety of industrial applications, consuming nearly half of the electrical energy produced worldwide. Motor control and electric drive applications are essential in various industries such as mechanical manufacturing, chemical processing, transportation etc. Understanding and knowledge of motor control principles are important for electrical, mechanical, and chemical engineers providing operation and maintenance of processing equipment in these industrial areas [5]-[8]. However, motor control is admitted to be quite a complicated engineering topic that is challenging to learn, understand, and implement; it requires knowledge of a variety of engineering subjects including mechanical engineering, electrical engineering, analogue and digital electronics, and embedded software programming [9].

Modern motor control methods and algorithms are based on microcontroller programming, which demands expertise and practical skills in software code development. Such embedded programming is usually taught to students studying electronic/software engineering, where microcontroller programming is part of the syllabus. Therefore, software programming skills become a prerequisite for students learning control engineering principles applied to microcontroller-based motor control. On the other hand, it could be a problem for non-electronic engineering students studying, for example, industrial engineering, processing engineering etc. and are not involved in the learning of embedded programming.

There are some solutions helping to avoid low-level programming in the design of embedded systems. The most popular solution is based on Matlab/Simulink which is widely used in engineering education and becomes now a standard software package for students in the educational process [10]-[15]. Students use Matlab/Simulink to study various subjects such as engineering mathematics, mechanical engineering principles, electrical and electronic engineering, etc. Today, engineering students study how to use Matlab/Simulink at the entry/low level of education to form a prerequisite basis required for further studies [16]-[18].

In order to enhance the time efficiency and to simplify the development of the software for microcontrollers, the Matlab/Simulink manufacturer MathWorks offers various software tools to provide interaction of code development software with Matlab/Simulink environment. This means that there is no need to directly develop code for the microcontroller. These tools can compile block diagrams developed in Matlab/Simulink environment into the code for subsequent uploading into the microcontroller. Using these tools, there is no need for detailed knowledge of microcontroller code development.

This paper discusses the implementation of Matlab/Simulink-based approach to the software development for Texas Instruments (TI) microcontroller to control an electric motor under various control algorithms. The software development and compilation procedure as well as an appropriate hardware setup are explained in detail. An example of a student practical experiment is provided to demonstrate the efficiency of the proposed learning and teaching approach.

979-8-3503-7973-0/24/$31.00 ©2024 IEEE

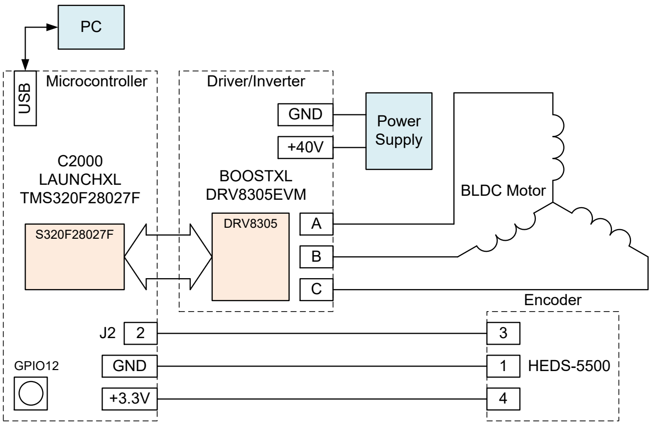

Fig. 1. The structure of the educational hardware setup.

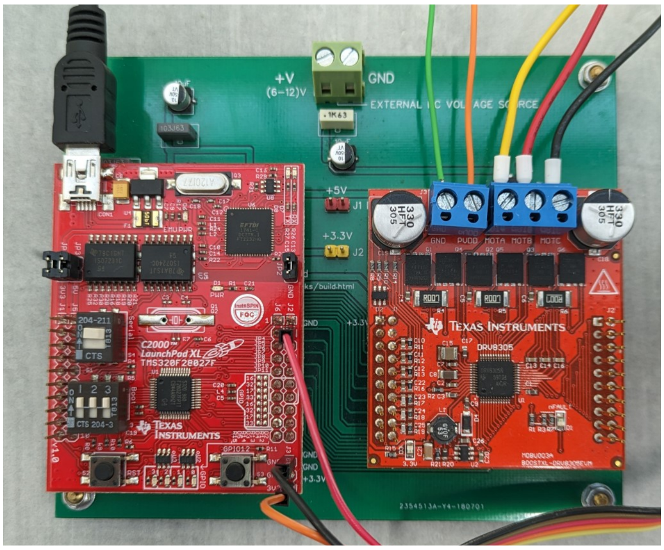

Fig. 2. Microcontroller and inverter.

(a) (b)

Fig. 3. (a) Encoder HEDS-5500 [22], (b) BLDC motor [19].

II. Hardware

The hardware for the student practical experiment is built using standard and widely available components. The hardware setup is designed for educational purposes; the structure of the hardware setup is shown in Fig. 1. The setup consists of the following components: a motor, an encoder, a 3-phase driver, and TI microcontroller. The motor is a low voltage 3-phase BLDC motor used in this experimental setup as a permanent magnet synchronous motor. The parameters of the BLDC motor are given in Table 1 [19]. The stator winding of the motor is energised by a 3-phase driver/inverter. The inverter is BOOSTXL-DRV8305 evaluation board manufactured by TI [20] and designed to operate under the control of a TI microcontroller MS320F28027F via the development board LAUNCHXL [21]. The inverter board supports an external bus power supply from 4.4V to 45V and a drive current of up to 15A RMS (20A peak). The microcontroller integrated into LAUNCHXL is a member of the C2000 series of microcontrollers developed by TI. This series is specifically designed for real-time control applications. The TMS320F28027F integrates a powerful 32-bit CPU based on the C28x core architecture with a rich set of peripheral functions and memory resources. The motor shaft is connected to the encoder HEDS-5500 [22] to obtain feedback on the motor rotational speed required for a closed-loop control [23]. The encoder has two channels and provides a resolution of 1000 counts per revolution. It can be supplied by a voltage in the range from 0.5V to 7V.

Fig. 2 shows the combination of microcontroller and inverter boards connected using a bus board. Fig. 3a shows the encoder HEDS-5500 and Fig. 3b demonstrates the BLDC motor. The proposed combination of low-power BLDC motor and HEDS-5500 encoder is quite similar to the approach suggested in [24]

Table I. BLDC Motor Parameters [19]

Parameter	Value
Number of Poles	4
Number of Phase	3
Rated Voltage	36V
Rated Speed	4000 RPM
Rated Torque	0.22Nm
Max Peak Torque	0.68Nm
Torque Constant	0.063Nm/A
Line to Line Resistance	0.65Ω
Line to Line Inductance	2.1mH
Max Peak Current	9.8A
Length	74mm
Rotor Inertia	119 g·cm^2
Weight	0.75kg

III. Software

The program to control the 3-phase BLDC motor is created in Matlab/Simulink software environment using a visual programming approach not requiring coding. The programme is divided into two main functions: motor control and speed control. The motor control is responsible for energising three-phase stator windings by generating ac voltage under PWM arrangement, whereas the speed control

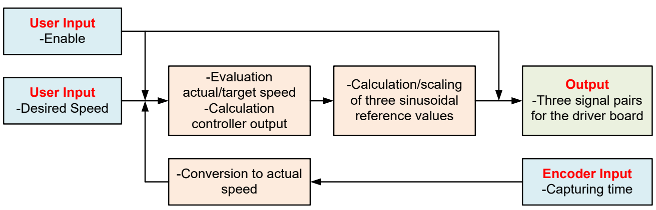

Fig. 4. Software flowchart.

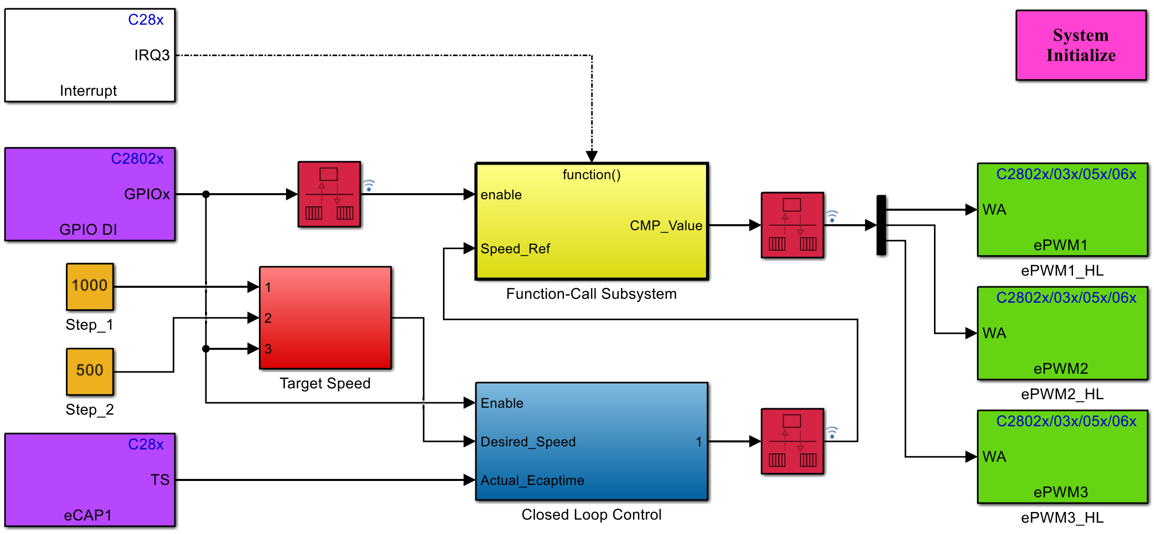

Fig. 5. Simulink block-diagram.

deals with the processing of the feedback signal corresponding to the motor shaft speed. Three basic requirements were introduced to develop the software: (1) the control should be as easy to understand as possible, and (2) the motor should be energised with sinusoidal voltages to ensure smooth running, (3) the system should operate under closed-loop control. Following the closed-loop control method, the actual rotational speed of the motor shaft is continuously recorded by the control algorithm and compared with the target speed.

The simplest motor control algorithm is based on the V/f scalar method. Since the motor shaft is not subject to dynamic mechanical loads in the planned laboratory test, this method is most suitable for practical implementation. The optical encoder already installed on the motor shaft provides a recording of the rotational speed for further processing in the microcontroller. A conventional PI controller is used in the software structure to achieve the desired speed as best as possible. The use of a simpler controller type has the advantage that the selection of the appropriate control parameters is simplified to just two values. The structure of the programme is shown in Fig. 4.

IV. Example

A practical exercise to build and compile software for the microcontroller to provide a PI control of BLDC motor is presented here as an example. This example can be used as a typical laboratory work for non-electrical/electronic students studying control engineering or basic principles of motor control.

The block diagram of the motor control example developed in Matlab/Simulink software environment using visual programming is shown in Fig. 5. It can be seen that the program consists of three sub-programs with varying sampling times. The motor is controlled using the “Function -Call Subsystem” (yellow), while speed control is provided by the subsystem “Closed Loop Control” (blue). The ePWM modules (green) for the three phases of the BLDC motor are located on the right side and are synchronized with each other. Simultaneously, the module (white) initiates the interrupt for the function call subsystem. The ePWM modules operate at a sampling frequency of 4 kHz, resulting in the execution of the function call subsystem at the same frequency. The remaining program is executed at 400 Hz. The digital input (purple) of the GPIO12, which is the programmable button on the launchpad, is located on the left side. It is utilised to enable various functions. The two target speeds (ochre) are applied as constants providing the desired speed values to the “Target Speed” sub-system.

The procedure of the practical experiment is as follows:

(1) Install Matlab/Simulink including C2000 Blockset package.

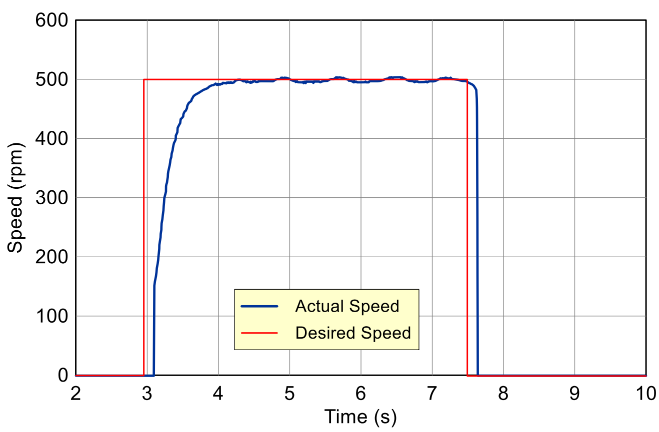

Fig. 6. Rotational speed transient at P = 0.3 and I = 6.4.

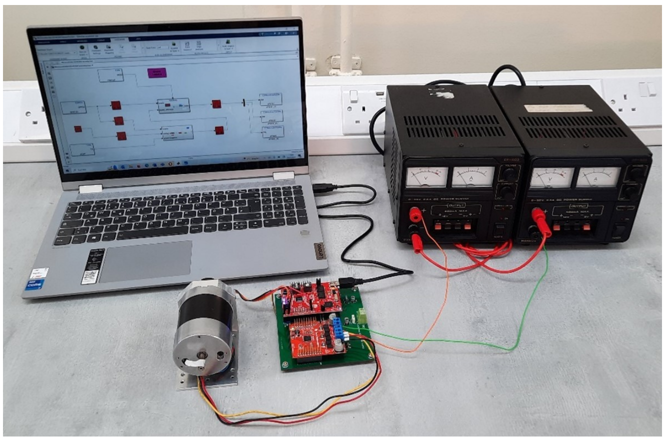

Fig. 7. The laboratory hardware setup.

(2) Use the Hardware setup window to check which third-party software needs to be downloaded and installed (TI controlSIUTE, TI Code Composer Studio, TI C2000 Ware).

(3) Open Sumulink settings of the specific microcontroller under Model/Hardware settings.

(4) Creating programming using the blocks from the library.

(5) Connect the hardware to the computer and transfer the compiled software into the microcontroller using the Build deploy and Start button.

(6) Set up and start the host model on the computer.

(7) Run and test the software uploaded to the microcontroller.

Once the software is developed and compiled into the microcontroller, the system can start operating under the created algorithm. Fig. 6 shows the result of the compiled software execution where the desired rotational speed was installed at 500 rpm and the proportional gain was P = 0.3 while the integral gain was I = 6.4. By varying both P and I gains a student can achieve different transients of the motor speed and learn how these parameters affect the system dynamic performance.

The hardware setup for the student laboratory investigation is shown in Fig. 7. The BLDC motor is supplied from the driver/inverter and the encoder is connected to the microcontroller development board. The developed software is uploaded into the microcontroller using a USB link.

The speed readings can be observed using an additional software installed in the microcontroller to record it and display it on the screen of the same PC which was employed for the programming. As an alternative option, external speed-reading hardware can also be implemented to collect speed data directly from the encoder, store it, and transfer it to a PC for further processing and visualisation.

V. Discussion

As can be seen from the provided example, the hardware part consists of the minimum number of parts and wires, thus it can be easily assembled by students who do not specialise in electrical engineering. The software part is also quite simple: it contains several blocks, which represent the control system and must be studied, and several blocks responsible for the low-level settings, which can be used without investigation. Therefore, the main purpose of focusing students' efforts on the control system itself and the exclusion of low-level programming is fully achieved.

VI. Conclusion

In a large variety of industrial and domestic applications, electric motors are often controlled by microcontrollers. In general, the development of embedded software for microcontrollers requires text-based programming coding. However, not all students studying engineering have in-depth knowledge of appropriate low-level programming coding. Alternative methods of programming such as visual programming languages significantly simplify the development of software for embedded applications including DSP-based microcontroller motor control.

This paper discusses an example of the development and compilation of the software for TI microcontroller in Matlab/Simulink environment. This approach is suitable for students studying engineering without an electrical/ electronic background. A simple hardware setup was built using two TI development boards and a low-power 3-phase BLDC motor. One board is the drive/inverter to energise the motor while another one consists of the microcontroller and appropriate peripherals. An optical encoder is installed on the motor shaft to provide the rotational speed feedback for closed-loop control.

A typical procedure of the software development for the proposed hardware setup is described. An example of the software execution showing a transient of the motor speed from rest to 500 rpm is discussed to demonstrate the capability of the hardware system to be used for educational purposes. This DSP-based approach can provide support for engineering students in learning the basic principles of control engineering and motor control motor.

References

[1] A. Anuchin and Y. Vagapov, "Instructional laboratory for practical investigation of electric drive control," *IET Circuits, Devices and Systems*, vol. 11, no. 4, pp. 344-351, 2017, doi: 10.1049/iet-cds.2016.0400

[2] A. Anuchin and M. Bychkov, "The modern electric drives — Using of information technologies and the problems of education," in *Proc. 58th IEEE Int. Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON)*, Riga, Latvia, 12-13 Oct. 2017, pp. 1-6, doi: 10.1109/RTUCON.2017.8124802

[3] M. Dal, "Teaching electric drives control course: Incorporation of active learning into the classroom," *IEEE Transactions on Education*, vol. 56, no. 4, pp. 459-469, Nov. 2013, doi: 10.1109/ TE.2013.2256424

[4] A. Anuchin, Y. Vagapov, and C. Belloc, "Development of curriculum for a postgraduate course on electric drives and motion control," in *Proc. 50th Int. Universities Power Engineering Conf. (UPEC)*, Stoke on Trent, UK, 1-4 Sept. 2015, pp. 1-4, doi: 10.1109/ UPEC.2015.7339849

[5] A. Anuchin, D. Savkin, Y. Khanova, and D. Grishchuk, "Real-time model for motor control coursework," in *Proc. 5th IEEE Int. Conf. on Power Engineering, Energy and Electrical Drives*, Riga, Latvia, 11-13 May 2015, pp. 427-430, doi: 10.1109/PowerEng.2015.7266355

[6] A. Anuchin and Y. Vagapov, "Configurable control systems of power converters for instructional laboratories," in *Proc. Int. Conf. Internet Technologies and Applications*, Wrexham, UK, 8-11 Sept. 2015, pp. 18-22, doi: 10.1109/ITechA.2015.7317362

[7] S. Wang, F. Zhang, Q. Tang, X. Zhang, and R. Zhao, "A take-home motor control teaching experiment platform for control engineering-related courses," *IEEE Transactions on Education*, vol. 65, no. 2, pp. 115-123, May 2022, doi: 10.1109/TE.2021.3094981

[8] C. Buiu, "Design and evaluation of an integrated online motion control training package," *IEEE Transactions on Education*, vol. 52, no. 3, pp. 385-393, Aug. 2009, doi: 10.1109/TE.2008.930090

[9] F. Sanfilippo, M. Okter, T. Eie, and M. Ottestad, "Teaching motion control in mechatronics education using an open framework based on the elevator model," *Machines*, vol. 10, no. 10, Oct. 2022, Art no. 945, doi: 10.3390/machines10100945

[10] A. Saghafinia, H. W. Ping, M. N. Uddin, and A. Amindoust, "Teaching of simulation an adjustable speed drive of induction motor using MATLAB/Simulink in advanced electrical machine laboratory," *Procedia Social and Behavioral Sciences*, vol. 103, pp. 912-921, 2013, doi: 10.1016/j.sbspro.2013.10.413

[11] W. Hao, H. Liu, and Y. Wang, "A DSP teaching example based on Simulink for undergraduates of electrical engineering," in *Proc. 20th Int. Conf. on Electrical Machines and Systems (ICEMS)*, Sydney, Australia, 11-14 Aug. 2017, pp. 1-4, doi: 10.1109/ ICEMS.2017.8056388

[12] L. Aamiovuori, H. Karkkainen, M. Niemela, J. Pyrhonen, B. Alalibo, and W. Cao, "Modelling a vector controlled induction motor in Simulink," in *Proc. X Int. Conf. on Electrical Power Drive Systems*, Novocherkassk, Russia, 3-6 Oct. 2018, pp. 1-8, doi: 10.1109/ ICEPDS.2018.8571659

[13] K. A. Makinde, M. S. Bakare, B. O. Akinloye, A. O. Amole, O. B. Adewuyi, U. O. Zubair, and W. O. Owonikoko, “Simulation based testing and performance investigation of induction motor drives using matlab simulink,” *SN Applied Sciences*, vol. 5, Feb. 2023, Art no. 73, doi: 10.1007/s42452-023-05296-w

[14] J. Sheng, “Teaching devices and controls for computer engineering and systems students using arduino and MATLAB/Simulink,” in *Proc. 14th IEEE Int. Conf. on Control and Automation (ICCA)*, Anchorage, AK, USA, 12-15 June 2018, pp. 318-323, doi: 10.1109/ICCA.2018.8444192

[15] B. Gecer, N. F. O. Serteller, and A. Ak, “Understanding a switched reluctance motor control and analysis methods using Matlab/Simulink,” in *Proc. IEEE World Conf. on Engineering Education*, Guatemala City, Guatemala, 14-17 March 2021, pp. 1-6, doi: 10.1109/EDUNINE51952.2021.9429155

[16] M. A. Vogelsberger, P. Macheiner, P. Bauer, and T. M. Wolbank, “Drives and motion control teaching based on distance laboratory and remote experiments,” *Journal of Power Electronics*, vol. 10, no. 6, pp. 579-586, Nov. 2010, doi: 10.6113/JPE.2010.10.6.579

[17] T. Matsuzaki, O. Elfadil, K. Horiuchi, H. Shiratsuchi, and K. Mashiko, “Embedded controller based learning system for DC motor control,” *International Journal of Innovative Computing, Information and Control*, vol. 15, no. 3, pp. 997-1007, June 2019, doi: 10.24507/ijicic.15.03.997

[18] M. D. Cook, J. L. Bonniwell, L. A. Rodriguez, D. W. Williams, and J. Pribbernow, “Low-cost DC motor system for teaching automatic controls,” in *Proc. American Control Conference (ACC)*, Denver, CO, USA, 1-3 July 2020, pp. 4283-4288, doi: 10.23919/ACC45564.2020.9147781

[19] RS PRO Brushless DC Motor [Online]. Available at: https://uk.rs-online.com/web/p/dc-motors/5366046

[20] BOOSTXL-DRV8305 [Online]. Available at: https://www.ti.com/tool/BOOSTXL-DRV8305EVM

[21] LAUNCHXL [Online]. Available at: https://www.ti.com/tool/LAUNCHXL-F2802x

[22] HEDS 5500 [Online]. Available at: https://www.faulhaber.com/en/products/series/heds-5500

[23] A. Dianov, “Comparison of feedback field-weakening techniques for synchronous machines with permanent magnets”, *Vehicles*, vol. 5, no. 4, pp. 1671-1691, 2023. doi: 10.3390/vehicles5040091

[24] A. Anuchin, A. Dianov, and F. Briz, “Synchronous constant elapsed time speed estimation using incremental encoders,” *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 4, pp. 1893-1901, Aug. 2019, doi: 10.1109/TMECH.2019.2928950