Parallel computing applied to auto-tuning of state feedback speed controller for PMSM drive

Rafal Szczepanski[1,*], *Tomasz* Tarczewski[1] and *Lech M.* Grzesiak[2]

[1]Department of Automatics and Measurement Systems, Nicolaus Copernicus University, Grudziadzka 5, 87-100 Torun, Poland
[2]Institute of Control and Industrial Electronics, Warsaw University of Technology, Koszykowa 75, 00-662 Warsaw, Poland

Abstract. Nowadays the simulation is inseparable part of researcher's work. Its computation time may significantly exceed the experiment time. On the other hand, multi-core processors can be used to reduce computation time by using parallel computing. The parallel computing can be employed to decrease the overall simulation time. In this paper the parallel computing is used to speed-up the auto-tuning process of state feedback speed controller for PMSM drive.

1 Introduction

Over the last few year multicore processors have become common in personal computers, but usage of them by programs is not trivial due to shared memory, scalability, synchronization, deadlock, race condition and difficult debugging process. There are many application programming interfaces (APIs) for parallel programming, the most known are: Open Multi-processing (OpenMP), Message Passing Interface (MPI), POSIX Threads (Pthreads). Each of these APIs have pros and cons, some of them are oriented for parallel computing and some are oriented for parallel tasking. One on the main disadvantage of these APIs is additional workload. Since the researcher's work requires a lot of modifications in model during tests, the most popular software used by researchers, which is MATLAB/Simulink offers Parallel Computing Toolbox. It is based on parallel CPU computing using MPI and parallel GPU computing using CUDA. This toolbox allows to use parallel computing without directly programming in these APIs. It provides functions that allows to take advantage of multicore processor or graphics card e.g. asynchronously run a Simulink model, run *for* loop parallel, parallel multiply matrices. In this paper Parallel Computing Toolbox has been used for auto-tuning space-state feedback speed controller (SFC) for PMSM drive.

2 Auto-tuning of SFC

Permanent magnet synchronous motor (PMSM) are commonly used in industrial robots, CNC machines, electrical and hybrid vehicles, air conditioning application [1, 2, 3]. The main reasons of its wide range of application are: compact construction, relatively high efficiency and high reliability [4]. The most commonly used control of PMSM drive is based on cascade of PI regulators [5], due to simple implementation and tuning process, while state-space feedback controller offers a better dynamical behavior and better disturbance compensation, but its tuning process is difficult [6]. In order to obtain optimal coefficients of SFC, that will fulfill respective objectives, the relatively new approach presented in [7] has been used. In a considered case, the main objective is to obtain a satisfactory dynamical behavior of the angular velocity and to ensure steady-state error-free operation in case of step changes of the reference value and of the load torque. Additionally, in order to maximize efficiency of the PMSM, the *d*-axis current should be zero and chattering in control signals shouldn't occur. The Artificial Bee Colony algorithm (ABC) [8] has been used to obtain optimal coefficients. Considered auto-tuning process requires several simulation of the drive during optimization.

3 Parallelization of algorithm

The Parallel Computing Toolbox for MATLAB/Simulink allows to take advantage of using both: multi-core processor (CPU) and graphical processing unit (GPU). It creates parallel pool, which is set of MATLAB workers on a compute cluster or desktop. The parallel pool can be used by functions e.g. *parfor* – parallel *for* loop, *parfeval* – asynchronous function computation, *parsim* – asynchronous Simulink model simulation. Auto-tuning process of SFC takes a several-tens of minutes. The most time takes simulation of considered plant (i.e. PMSM fed by VSI) in Simulink environment. In order to reduce computation time, all phases of ABC have been divided into three parts: (i) generate all new food sources, (ii) evaluating new food sources and (iii) comparing actual food sources with a new one. The first and third part is evaluating on single

[*] Corresponding author: szczepi@fizyka.umk.pl

thread, because they are not very time consuming, while the second part is executing with parallel pool of workers (*parsim* function).

4 Results and efficiency analysis

Used parameters of PMSM drive, Artificial Bee Colony algorithm and performance index are presented in [7]. Due to application of parallel computing the number of Food sources was increased to 40, while the number of iteration was decreased to 25.

The same problem has been solved with number of workers in parallel pool from 1 to 8, but also on three different personal computers: (i) Intel Core i3-3120M 2.5GHz (2 cores, 4 threads) supported by 4GB RAM, (ii) Intel Core i5-7500 3.4GHz (4 cores, 4 threads) supported by 16GB RAM and (iii) Intel Core i7-7700HQ 2.8GHz (4 cores, 8 threads) supported by 8GB RAM. The operating system used on each computer was Windows 10. In order to compare computation time, *speedup* term is used, which is defined as sequential program computation time divided by parallel program computation time. The graphical representation of results of computation time reduction for presented optimization problem is shown in the Fig. 1.

The results confirm that parallel computing allows decreasing computation time of auto-tuning of state feedback speed controller for PMSM drive. Speedup obtained for i5-7500 and i7-7700HQ are similar, due to the same number of physical cores. On the other hand, the maximum speedups for these processors are obtained with different number of workers, which is caused by different number of logical threads.

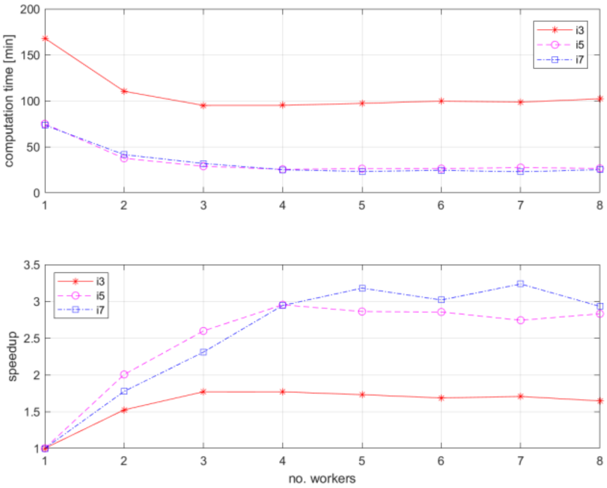

Fig. 1. Graphical representation of computation time and speedup for different number of workers.

5 Conclusion

In this paper, efficiency analysis of parallel computing applied to auto-tuning of state feedback speed controller for PMSM drive has been presented. The Artificial Bee Colony has been applied to auto-tuning process of state feedback speed controlled for PMSM drive fed by VSI. The computation time of solving described optimization problem has been examined on various parallel pool size and on three different personal computers. The results confirm a positive effect of parallel computing on the computation time. The maximum obtained speedup is equal to 1.77 with i3-3120M, 2.95 with i5-7500 and 3.23 with i7-7700HQ. The optimal number of workers depends on number of physical cores and logical threads of the processor. Increasing the number of workers above optimal value does not improve speedup and even decrease it, which is caused by race condition. It is worth to point out, the operating system has processes, that have higher priority than Matlab application. For this reason, Matlab is not able to use all physical cores during the whole simulation time. It should be noted, that by increasing the number of logical threats the management possibilities of operating system increases but the computational capability does not change.

References

1. R. Szczepanski, K. Erwinski, M. Paprocki, IEEE Intern. Conf. MMAR 2017, 431-436 (2017).
2. X. Liu, H. Chen, X. Zhao, A. Belahcen, IEEE Trans. Ind. Electron., **63**, 6 (2016).
3. L. Faa-Jeng, S. Hsin-Jang, S. Po-Huang, S. Po-Hung, IEEE Trans. Syst., Man, Cybern., **36**, 2 (2006).
4. L. Zhong, M. F. Rahman, W. Y. Hu, K. Lim, IEEE Trans. Power Electron., **12**, 3 (1997).
5. L. Jarzebowicz, Proc. IEEE Intern. Conf. EPE 2014, 901-905 (2014).
6. G. F. Franklin, J. D. Powell, A. Emami-Naeini, J. D. Powell, *Feedback control of dynamic systems*, **3** (1994).
7. T. Tarczewski, L. M. Grzesiak, IEEE Trans. Ind. Appl., **54**, 3 (2018).
8. D. Karaboga, B. Basturk, Foundations of Fuzzy Logic and Soft Computing, **4529** (2007).