# Acceleration of Hyper-Parameter Auto-Tuning with Parallelization and Time Constraints


Chaoyi Zhang
Graduate School of Information Sciences
Tohoku University
Japan
zhang.chaoyi.q4@dc.tohoku.ac.jp

Ryusuke Egawa
Cyberscience Center
Tohoku University
Japan
egawa@tohoku.ac.jp

Hiroyuki Takizawa
Cyberscience Center
Tohoku University
Japan
takizawa@tohoku.ac.jp


## 1 Introduction

Machine learning is now playing an important role in many fields. One significant problem that still confuses machine learning researchers is how to find an optimal hyperparameter configuration quickly and accurately.

The process of finding an optimal configuration of hyper-parameters is called hyper-parameter tuning. In hyper-parameter tuning, an appropriate hyper-parameter configuration is explored in a trial-and-error fashion. In one trial, the target machine learning model is trained with a selected configuration. After that, in the case of poor performance in terms of accuracy, another hyper-parameter configuration is selected. These steps are repeated until a hyper-parameter configuration with acceptable performance is found. Since the training and testing steps are themselves time-consuming and repeated many times, the hyper-parameter tuning could need very long time.

## 2 Problem Description

This work considers the above problem through Bayesian optimization (BO), which has shown excellent performance in hyper-parameter tuning. BO selects a prior (normally the Gaussian process prior) over hyper-parameter configurations. Then, it uses previous observations to predict the most promising candidate for the next trial, which allows BO can find the optimal configuration with a small number of trials. However, BO still needs a long time in practical use, and thus there is demand for further acceleration.

Wang et al. [1] proposed an approach that takes the execution time of each trial into account in the cost function. (instead of considering only the accuracy in conventional approaches) during the hyper-parameter tuning process. However, their time-constraint method has a negative effect on the model accuracy achieved with the obtained hyperparameter configuration.

## 3 Proposed Method

The goal of this work is to minimize the execution time of BO process. To achieve this goal, we proposed two methods.

The first is a time constraints method that considers both the execution time and accuracy in the cost function:

$$Z = L + b * T \,. \tag{1}$$

Here, 'Z' means the value of cost function, 'L' means the loss, 'T' means the execution time, and 'b' is a constant parameter. We use this cost function in the beginning of the tuning process to reduce the total execution time of hyperparameter tuning by consciously selecting candidates with shorter execution time. Then, at a certain point of time, the cost function is changed back to the standard one:

$$Z = L \,, \tag{2}$$

to ensure the accuracy.

The second is to achieve the parallelization of BO. Since BO needs the results of previous trials, a new trial cannot start before its previous one finishes. Thus, the original Bayesian Optimization sequentially performs trials one by one. The current parallelization methods [2] give new candidates before the previous trials finish through predicting the approximate results of the pending trials. This work applies a parallelization method without any prediction. The basic idea is that, when a node of a parallel computing system completes a trial, we use the results to get the next candidate and send it back to that node. Since the execution time of each trial is different, if we send one random configuration to each node as the first candidate, all the nodes can run trials in parallel.

## 4 Preliminary Evaluation

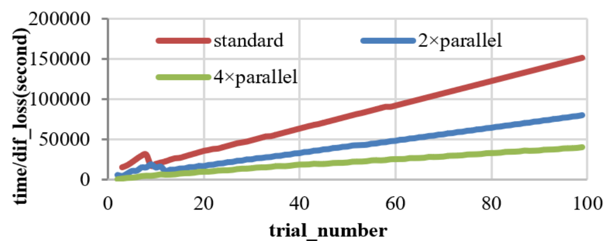


**Figure 1: comparison evaluation between parallelized BO and standard BO.**

Figure 1 shows the performance evaluation results to compare original and parallelized Bayesian Optimization methods, in which a convolutional neural network is trained with the CIFAR-10 dataset. The y-axis is the ratio between time and the difference of loss. If the ratio is higher, it means the method need more time to reduce loss. The result shows that our method works well in parallelization.

This work also compares the standard and the time-constraint BO methods. The total execution time is reduced by approximately 5% by using the proposed time-constraint BO, while the best accuracy among the trained models is kept statistically unchanged.

## REFERENCES


[1] Wang et al., "Automatic Hyperparameter Tuning of Machine Learning Models under Time Constraints," IEEE International Conference on Big Data, 2018, pp.4967-4973.

[2] Snoek et al., "Practical Bayesian Optimization of Machine Learning Algorithms," in Advances in neural information processing systems, 2012, pp. 2951–2959.