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

Chaoyi Zhang Ryusuke Egawa Hiroyuki Takizawa

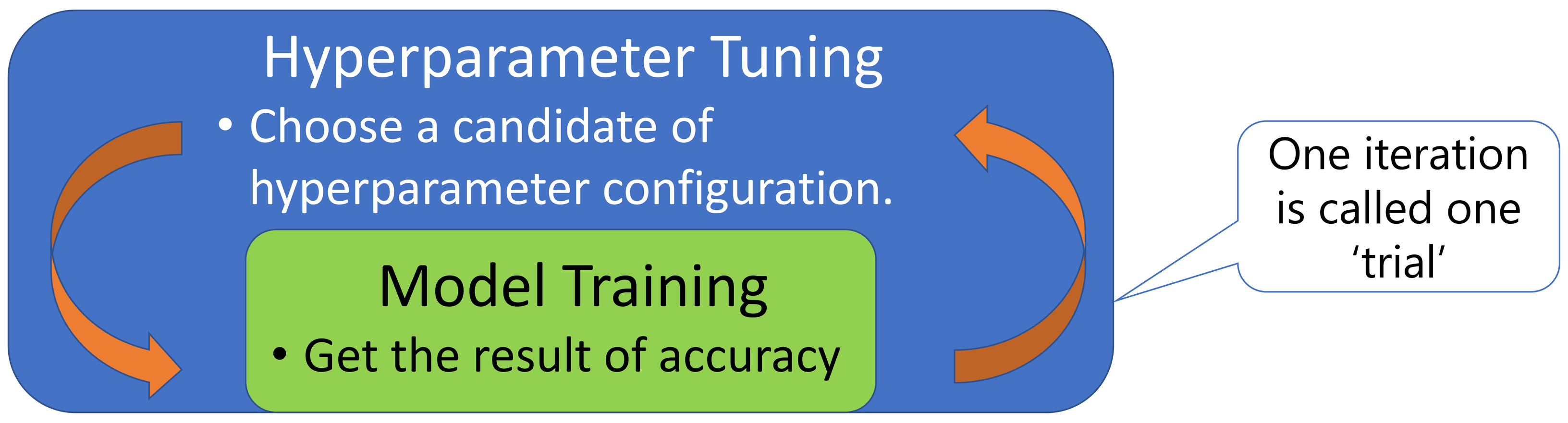
TOHOKU UNIVERSITY

Cyberscience Center

Background

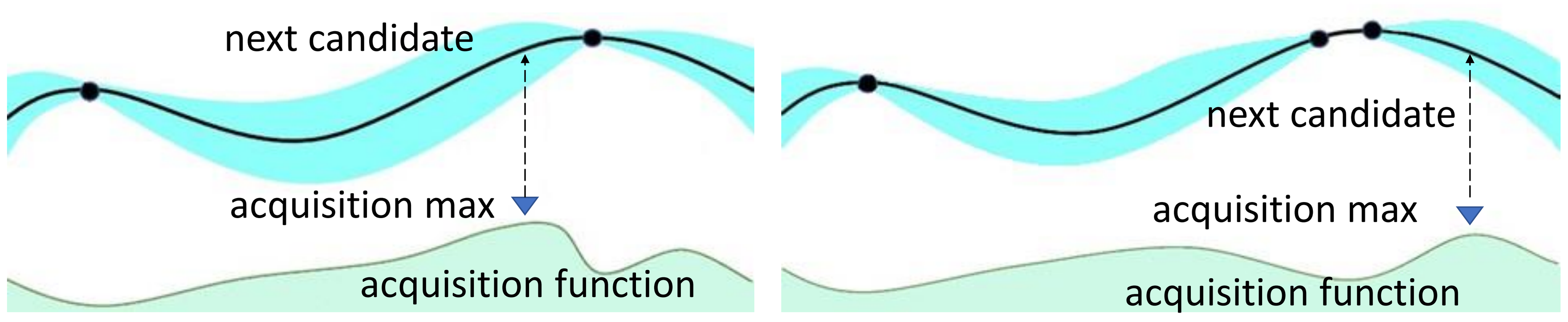
Hyperparameter tuning.

- **One significant problem** in machine learning field → **hyperparameter tuning**.
  - ➢ Hyperparameters: a kind of parameters not tuned during the model training process.
  - ➢ Appropriate hyper-parameters setting is important.
  - ➢ Hyperparameter tuning: the process of finding an optimal configuration of hyperparameters.



Bayesian Optimization

- Bayesian Optimization (BO): a method that can find the optimal configuration with a smaller number of trials.
- Two Steps before performing BO:
  - ➢ Find a **prior function** (Gaussian process prior) over the objective function.
  - ➢ Use an **acquisition function** to predict the next optimal candidate through the previous observations.
- One iteration:
  - ➢ Find the **maximum** of the acquisition function to **determine the next candidate.**
  - ➢ Evaluate the objective function at the point of the candidate.
  - ➢ Update the Gaussian process prior function and the acquisition function.



Problem & Objective

Problem

- The hyperparameter tuning with BO can be really **time-consuming**.
  - ➢ 1) One trial can be time-consuming.
  - ➢ 2) A large number of trials need to be performed sequentially.

Objective

- Accelerate the process of hyperparameter tuning with BO.
  - ➢ 1) Use a **time-constraint method** to help BO choose the hyperparameter configuration that needs a **shorter execution time** for model training.
  - ➢ 2) Achieve the **parallelization** of BO.

Proposed Approaches

Time-constraint Approach

- Time-constraint approach considers **both the execution time and accuracy** in the cost function.

I. In the first n trials, the proposed method uses observations based on the cost function in Equation (1) for BO to select the candidates that **need a shorter execution time**.

II. After n trials, the proposed method uses observations based on the cost function in Equation (2) to get better accuracy.

$$\begin{cases} Z = L + b * T & ,0<n. & (1) \\ Z = L & ,n \le n\_max. & (2) \end{cases}$$

- Here, Z means the value of cost function, L means the loss, b is a constant parameter, T means the execution time, n means the trial numbers and n_max means the max trial numbers.

Parallelization Approach

I. Send one random configuration to each node.

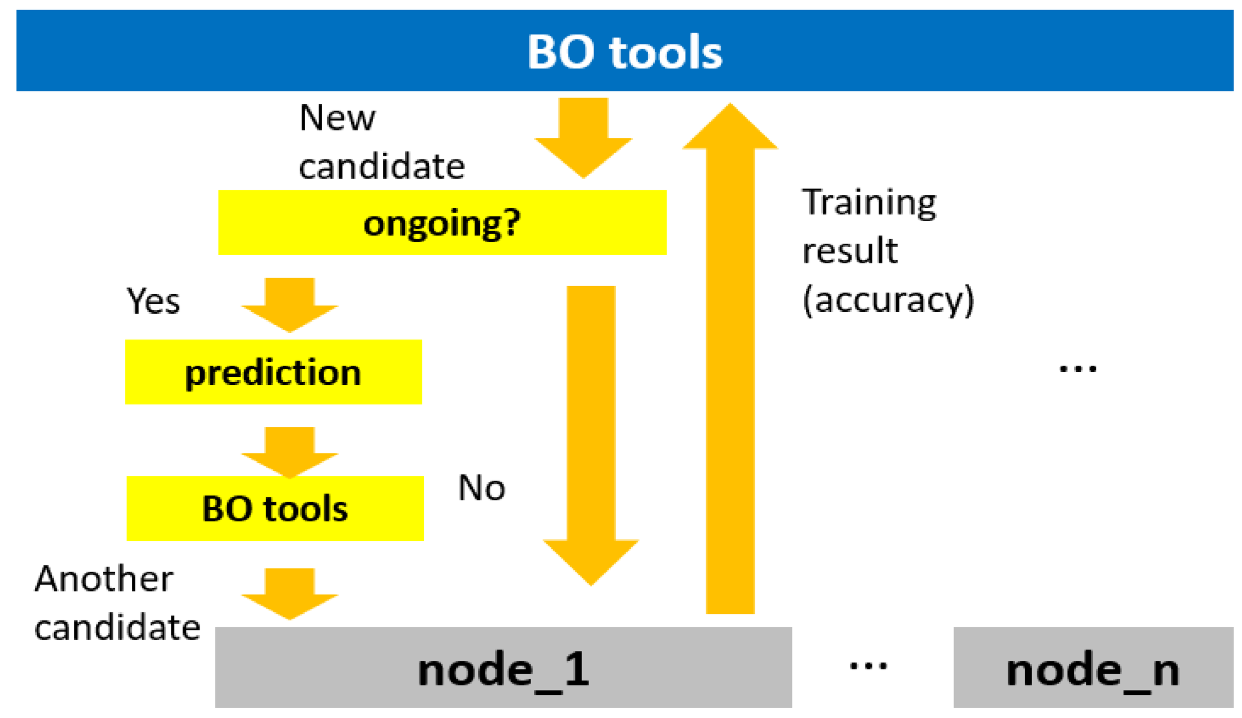
II. When a node completes a trial, BO chooses the next candidate.

III. If BO chooses a candidate that is the same as the ongoing candidate, the proposed method **predicts the result of the ongoing candidate**, and BO chooses another candidate.

IV. Send the new candidate to the node.

V. Repeat steps Ⅱ and Ⅲ for each node.

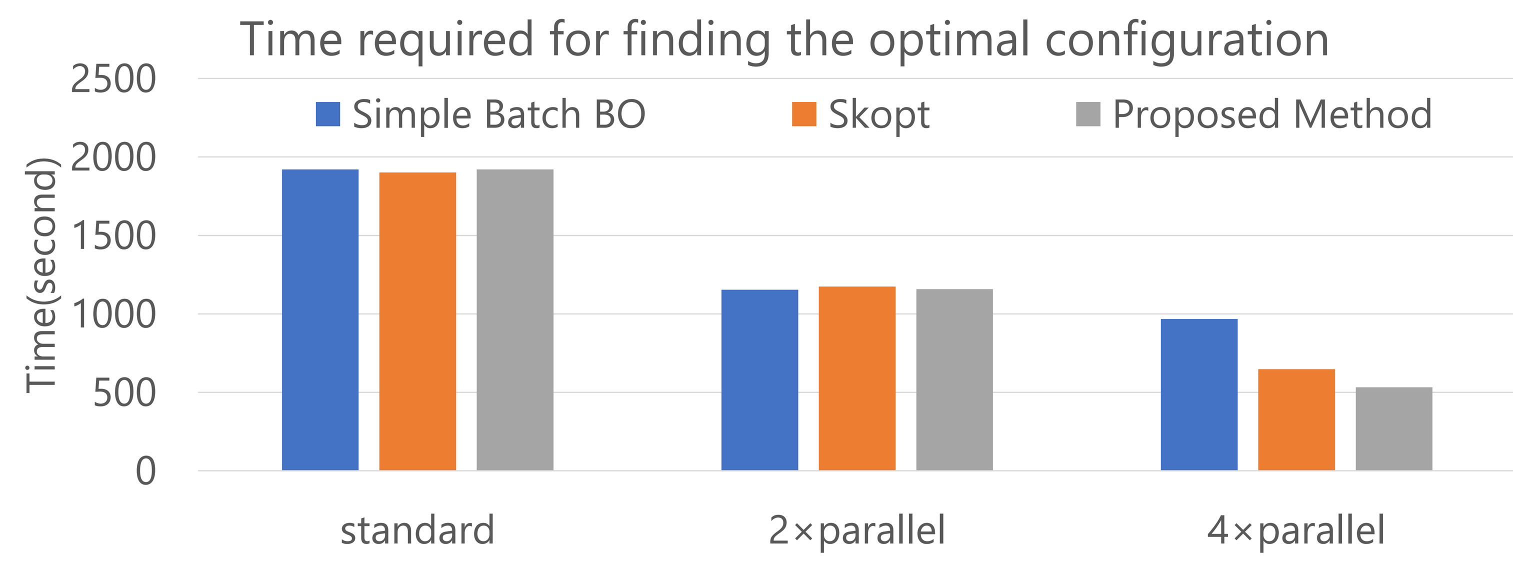
- Since the execution time of each trial is different, **each node asks for the next candidate at different points of time**.
- Step Ⅲ can **prevent the proposed method from choosing the same hyperparameter configuration for each node**.



Evaluation

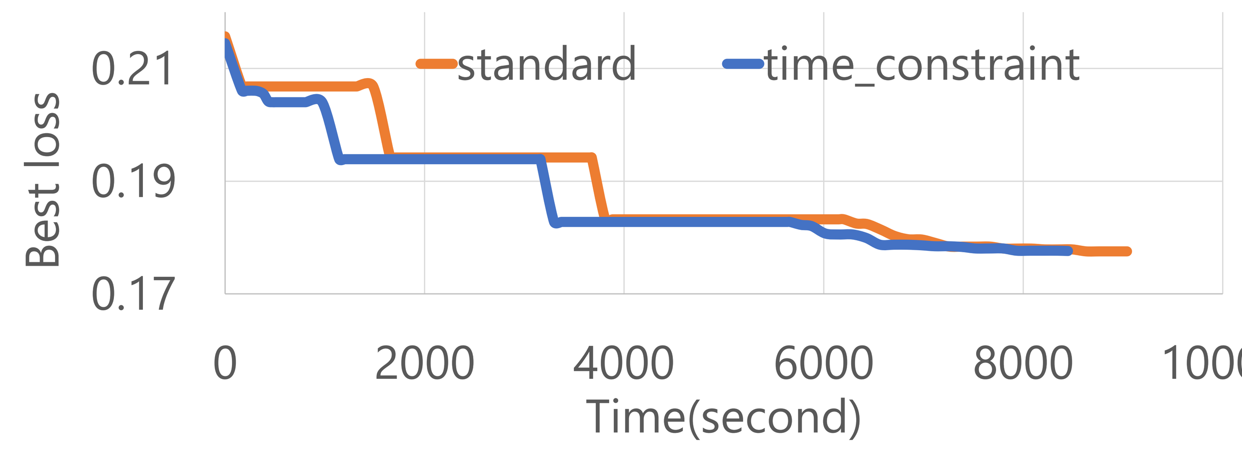
- As the evaluation, hyperparameters of a **CNN** model are auto-tuned for an object classification problem, called **CIFAR-10**.

Parallelization Approach



- In the figure, 'standard' is conventional BO and 'N × parallel' is the parallelized BO with N nodes.
- Proposed method is **faster in the case of using 4 nodes**.

Time-constraints Approach



- 'standard' is conventional BO and 'time_constraint' is time-constraint BO.
- The total **execution time is reduced** time-constraint approach, while the **best accuracy** among the trained models is kept statistically **unchanged**.

Conclusions

- To accelerate the hyperparameter tuning with BO, parallelization method and time-constraint method are proposed.
- Both of the proposed methods can **reduce the execution time, while the best accuracy is statistically unchanged.**

Acknowledgement

- This work is partially supported by MEXT Next Generation High-Performance Computing Infrastructures and Applications R&D Program "R&D of A Quantum-Annealing-Assisted Next Generation HPC Infrastructure and its Applications," and Grant-in-Aid for Scientific Research(B) #17H01706.