Speeding up of the Nelder-Mead Method by Data-driven Speculative Execution

Shuhei Watanabe *1,*2 Yoshihiko Ozaki *1,*3 Yoshiaki Bando *1 Masaki Onishi *1
*1 National Institute of Advanced Industrial Science and Technology, *2 The University of Tokyo, *3 Gree .inc

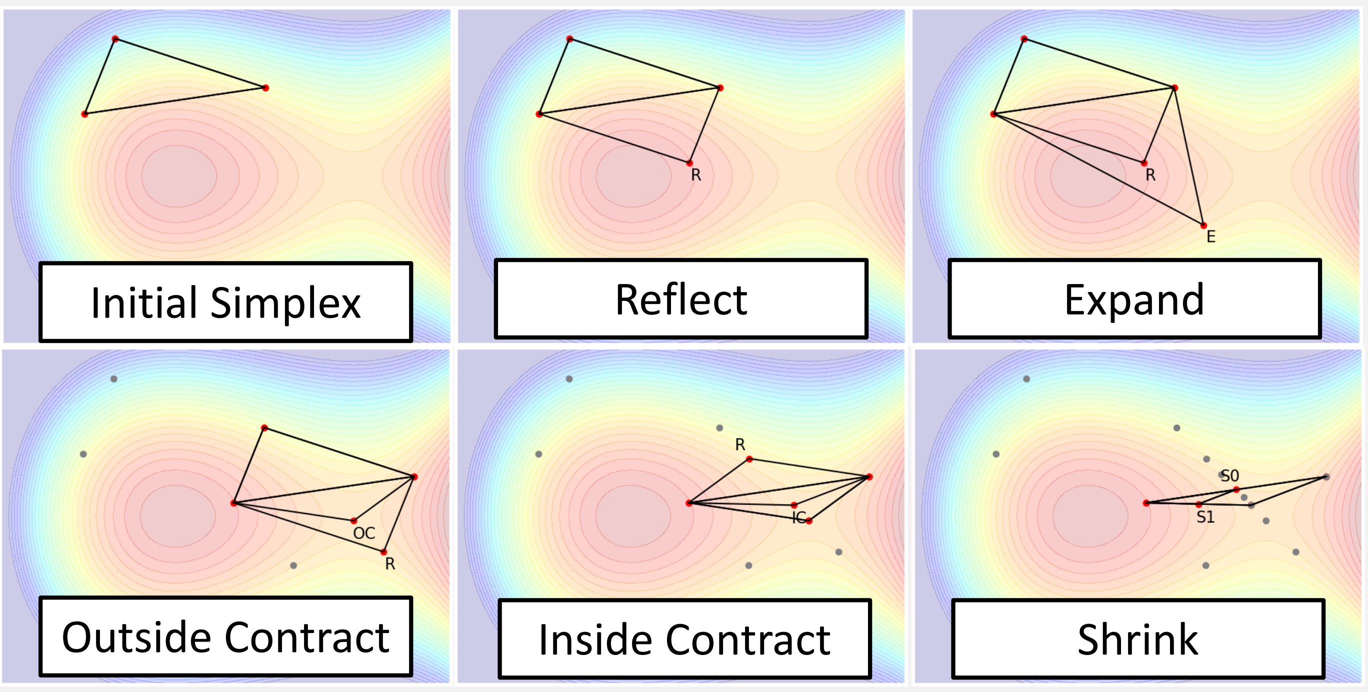
Brief Overview

Problem
1. The performance of machine learning algorithms considerably depends on the hyperparameter settings.
2. The Nelder-Mead (NM) method [2], which outperformed other promised hyperparameter optimization (HPO) methods [4] does not scale to the number of processors.

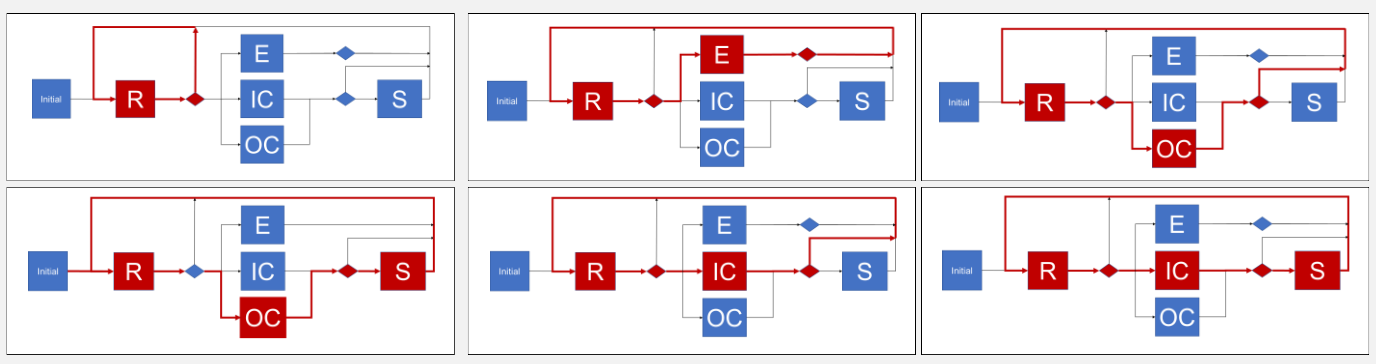
Proposed method
1. The NM method tends to take specific transitions during optimization processes, and we analyzed the traits on the experiments on benchmark functions
2. We used the statistical information obtained in the experiments above and accelerated the Nelder-Mead method.

Nelder-Mead (NM) Method

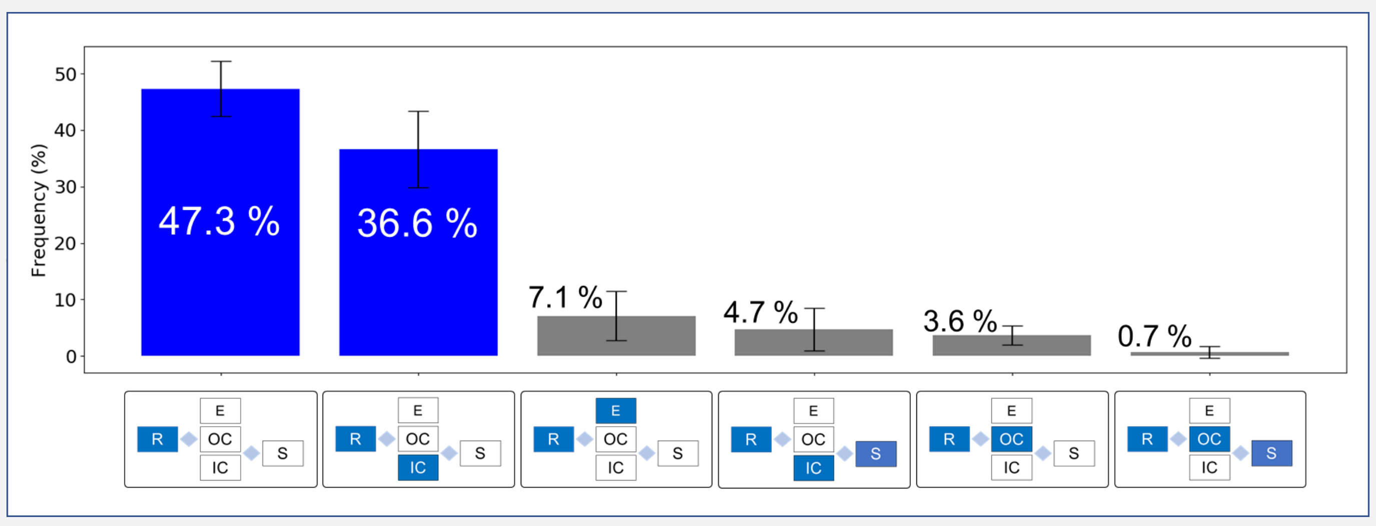
Brief description of the NM method
1. The NM method optimizes objective function using 5 operations called “Reflect” (R), “Expand” (E), “Inside Contract” (IC), “Outside Contract” (OC), “Shrink” (S).



2. Every iteration, the NM method can take 6 types of transition.



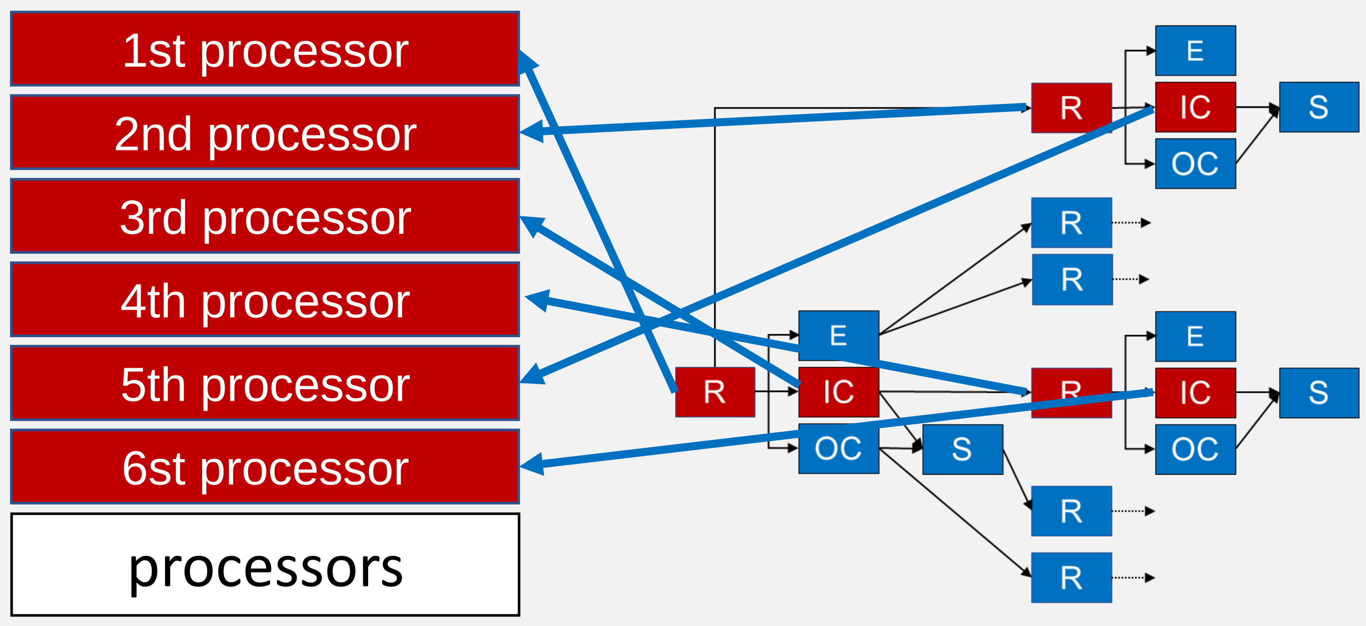
Experiments to obtain the statistics of the NM method
1. We experimentally obtained the probabilities of each transition for speculative execution of the NM method.
2. We optimized 17 kinds of 10-dimensional noisy benchmark functions by the NM method.
3. Then the probabilities of each transition were calculated based on the optimized data.



Observations
1. Each transition of the NM method does not occur evenly.
2. The figure below shows the probabilities of each transition in 1 iteration in the NM method and 2 out of 6 transitions occupies the 83 % of transitions in the optimization experiments.

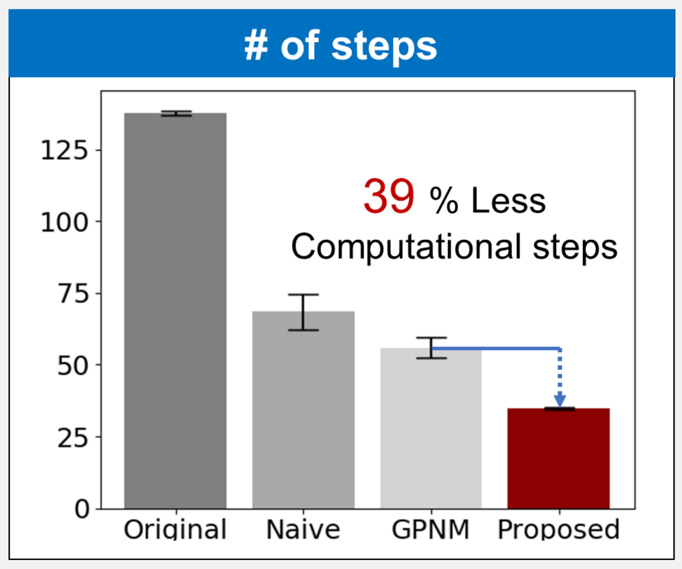
Proposed Method

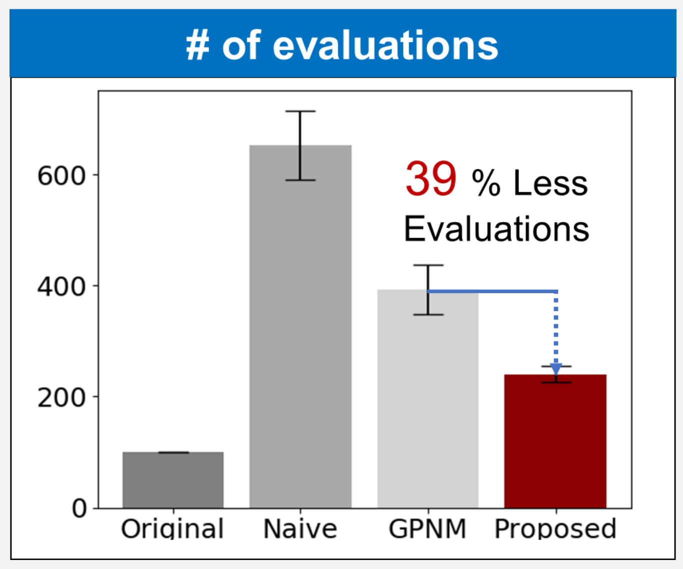
Algorithm
1. Consider all the possible points.
2. Take the points by the order of the probabilities of transition. Here we used the probabilities obtained in the previous experiments.
3. Repeat this operation as long as some processors are available



Experiments

1. We optimized wide residual neural networks (WRN28-10) [5] trained on CIFAR100 and evaluated 100 hyperparameter settings during an optimization process.
2. Our method was compared with two parallel NM methods: Naïve Parallel NM (NP-NM) [1] method and Gaussian Process based NM method (GP-NM) [3].





Conclusion

1. We successfully speeded up the NM method by 40 % comparing to the GP based parallel NM method.
2. In the experiments, we found out that if the proposed method adapted to the asynchronous method, the NM method can be accelerated furthermore. Therefore, making our method asynchronous would be the future study.

Reference

1. Dennis, J., Torczon, V.: Parallel implementation of the nelder-mead simplex algorithm for unconstrained optimization. In: High Speed Computing. Vol. 880, pp. 187-192. International Society for Optics and Photonics (1998).
2. Nelder, J.A., Mead, R.: A simplex method for function minimization. The computer journal 7(4), 308-313 (1965).
3. Ozaki, Y., Watanabe, S., Onishi, M.: Accelerating the nelder-mead method with predictive parallel evaluation. In: 6th ICML Workshop on Automated Machine Learning (2019).
4. Ozaki, Y., Yano, M., Onishi, M.: Effective hyperparameter optimization using nelder-mead method in deep learning. IPSJ Transactions on Computer Vision and Applications 9(1), 20 (2017).
5. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: Proceedings of the British Machine Vision Conference (BMVC). Pp. 87.1-87.12. BMVA Press (2016).