An Autotuning-based HPO Framework for Mixed-kernel SVM Classifications in Science and Engineering on Aurora

Xingfu Wu, Tupendra Oli, Valerie Taylor
Argonne National Laboratory

Justin H. Qian, Mark C. Hersam, and Vinod K. Sangwan
Dept. of Materials Science and Engineering, Northwestern University

Scalable Tools Workshop, Madison, WI, July 26-30, 2026

Mixed-kernel Support Vector Machine (SVM)

- A popular classification method widely used in science and engineering

- Handles both linear and nonlinear classifications
- When data is not linearly separable, kernel functions are used as preprocessing methods to be applied to the training data to transform it to higher-dimensional space in order to make the data linearly separable.

- Some popular nonlinear kernel functions: polynomial, Gaussian, and Sigmoid
  - In general, the Gaussian outperforms the polynomial in both accuracy and convergence time
  - The Gaussian has interpolation ability and is effective at identifying local properties
  - The Sigmoid is better suited for identifying global characteristics but has a relatively weak interpolation ability

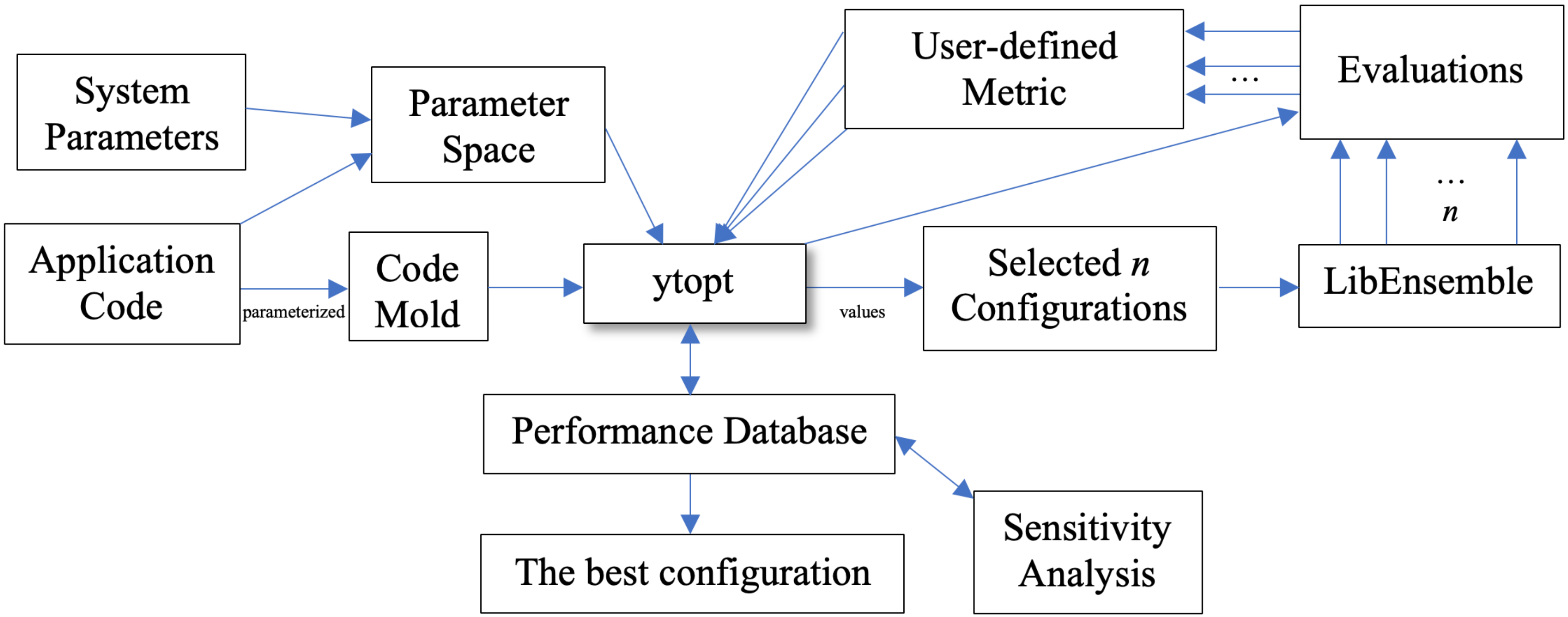
- The mixed kernel between the Gaussian and Sigmoid in SVMs takes the advantages of both kernels.

Background, Motivations, and Our Approach

- In our DOE ASCR-funded Microelectronics projects Threadwork and BIA,
  - Mixed-kernel heterojunction transistor device was developed by Northwestern U.
  - Application driver focuses on HEP detector datasets

- **Approach**: Use mixed kernel SVMs to process HEP detector datasets
  - Utilize *SVC* from scikit-learn to implement a mixed-kernel SVM model
    - *SVC(C=C, kernel=lambda x, y: ExperimentalKernel().mixed_kernel(x, y, mixed_ratio, sigmoid_ratio, coef0, gaussian_ratio))*
  - In HEP, commonly use grid search to conduct initial hyperparameter optimization with the focus on the mixed ratio, sigmoid ratio, and gaussian ratio and obtained the accuracy 83.4% as the baseline
  - Use our autotuning tool ytopt to conduct hyperparameter optimization for five hyperparameters: the mixed ratio, sigmoid ratio, gaussian ratio, the regularization parameter C, and the parameter coef0 in SVC.

ytopt-libe: Asynchronous Autotuning Framework



https://journals.sagepub.com/doi/10.1177/10943420241286476

https://github.com/ytopt-team/ytopt/tree/main/ytopt-libe

Methodology of the Autotuning Framework ytopt

- The search method within ytopt uses Bayesian optimization, where a dynamically updated Random Forest surrogate model that learns the relationship between the configurations and the performance metric, is used to balance exploration and exploitation of the search space.

  - In the exploration phase, the search evaluates parameter configurations that improve the quality of the surrogate model,
  - In the exploitation phase, the search evaluates parameter configurations that are closer to the previously found high-performing parameter configurations.
  - The balance is achieved through the use of the lower confidence bound (LCB) acquisition function that uses
    - the surrogate model' predicted values of the unevaluated parameter configurations
    - and the corresponding uncertainty values (standard deviation).

Two Asynchronous Aspects of the ytopt-libe

- *The asynchronous aspect of the search* allows the search to avoid waiting for all the evaluation results before proceeding to the next iteration.
  - As soon as an evaluation is finished, the data is used to retrain the surrogate model, which is then used to bias the search toward the previous promising configurations.

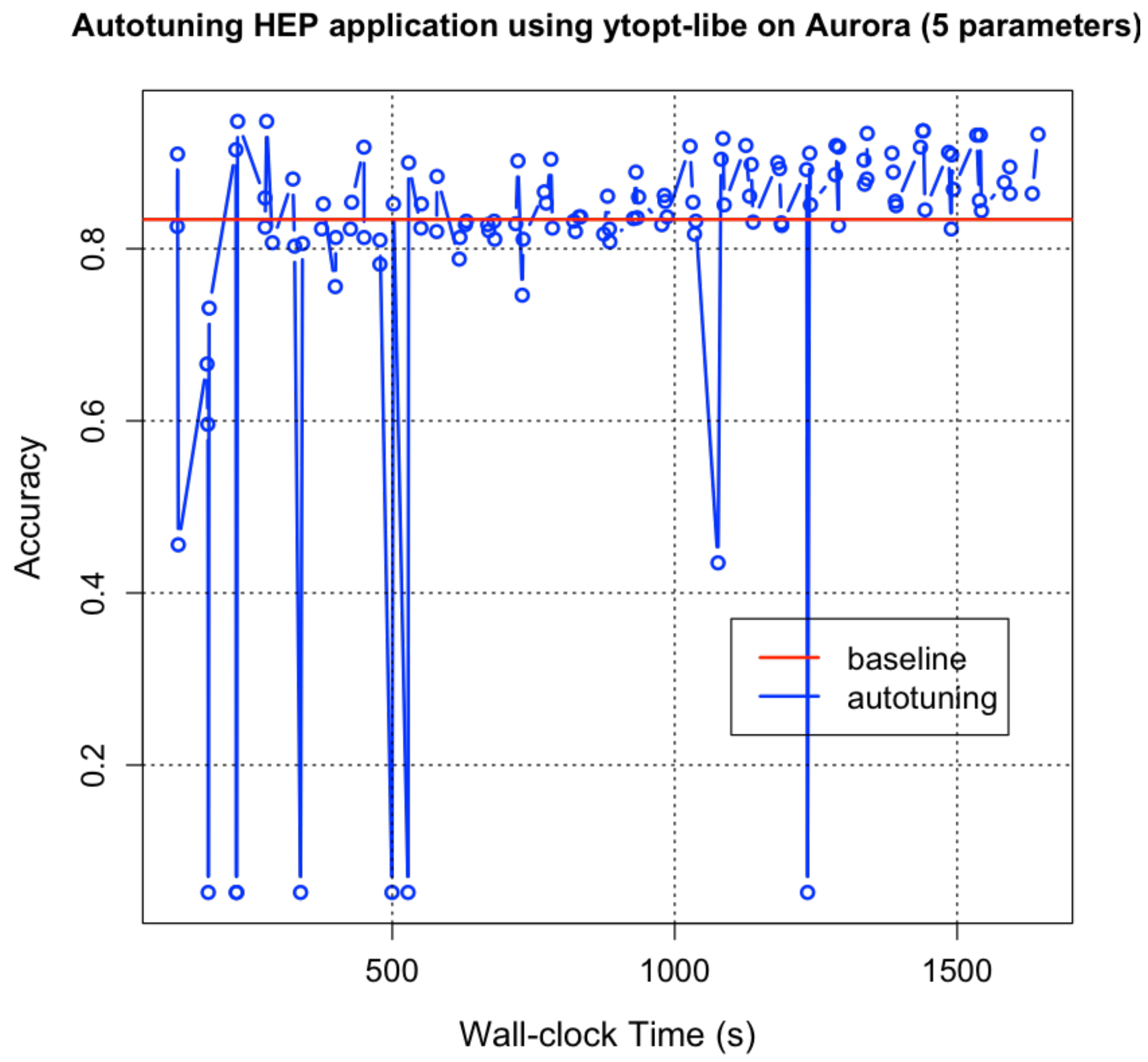
- *The asynchronous aspect of the evaluation* allows ytopt-libe to evaluate multiple selected parameter configurations in parallel by using the asynchronous and dynamic task manager libEnsemble.

Mixed Kernel SVMs for HEP Detectors

Tunable mixed-kernel SVM enables efficient data filtering for tracking systems and smart pixel HEP detectors

- Use Smart Pixel Datasets generated from Fermi Lab
  - This dataset comprises the charge deposited in each pixel per time slice upon pion interaction with the silicon pixel detectors, along with the truth information of each interacting pion.
- Utilize *SVC* from popular scikit-learn to implement a mixed-kernel SVM model to distinguish between low and high transverse momentum tracks
  - `SVC(C=C, kernel=lambda x, y: ExperimentalKernel().mixed_kernel(x, y, mixed_ratio, sigmoid_ratio, coef0, gaussian_ratio))`
- Use ytopt to conduct hyperparameter optimization for five hyperparameters: the mixed ratio, sigmoid ratio, gaussian ratio, the regularization parameter C, and the parameter coef0 in SVC.

Autotuning the HEP Application with 5 parameters



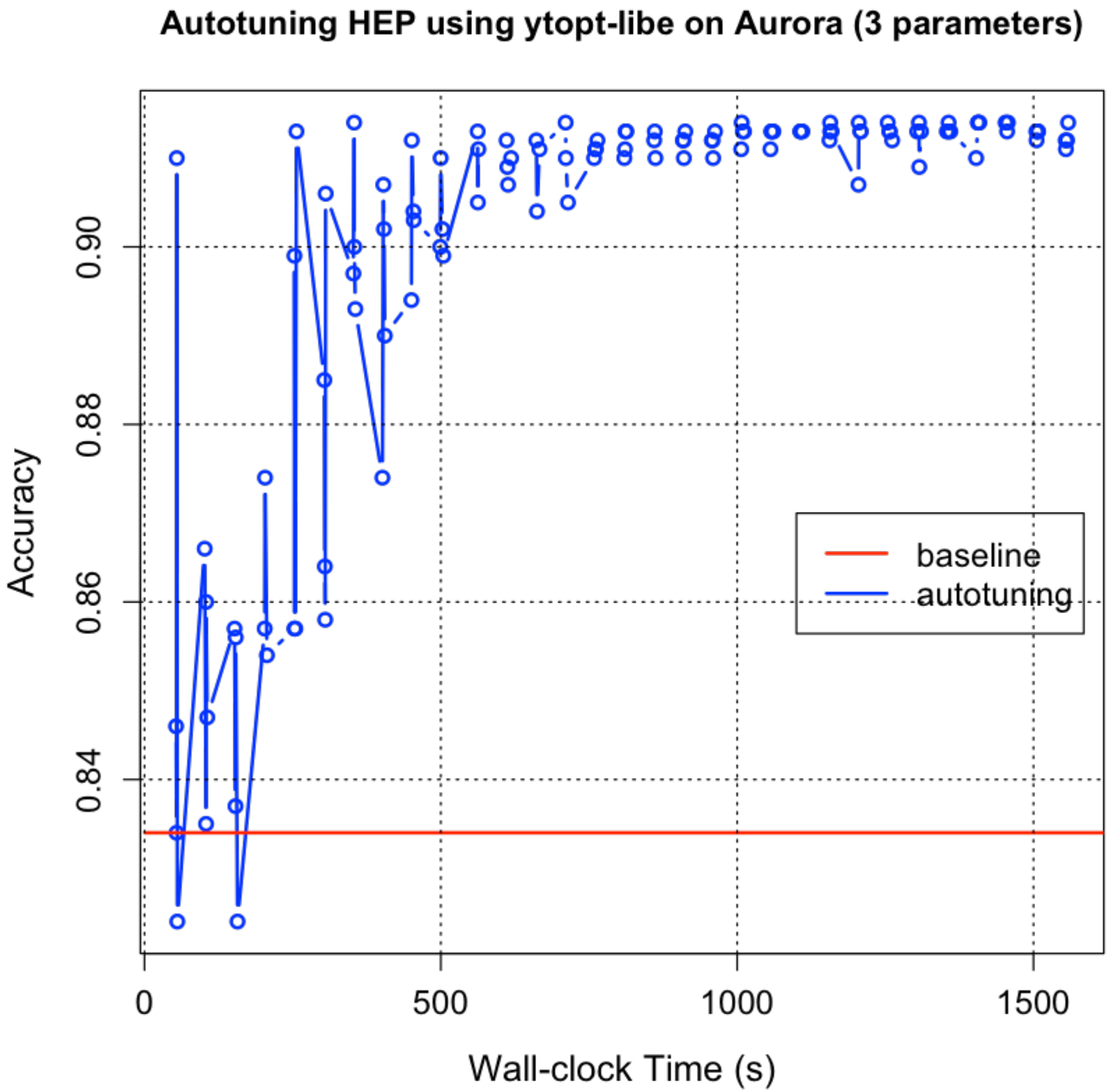
What We Observed

- SVC from scikit-learn:
  - Keep the ranges the same for the mixed ratio, sigmoid ratio, and gaussian ratio
  - the regularization parameter C
    - Common to all SVMs
    - The default value 1.0
    - The strength of regularization for training data is inversely proportional to C
      - If the data has a lot of noisy observations, decrease C (more regularization)
    - Must be positive real number
    - Set the large range [0.1, 100]

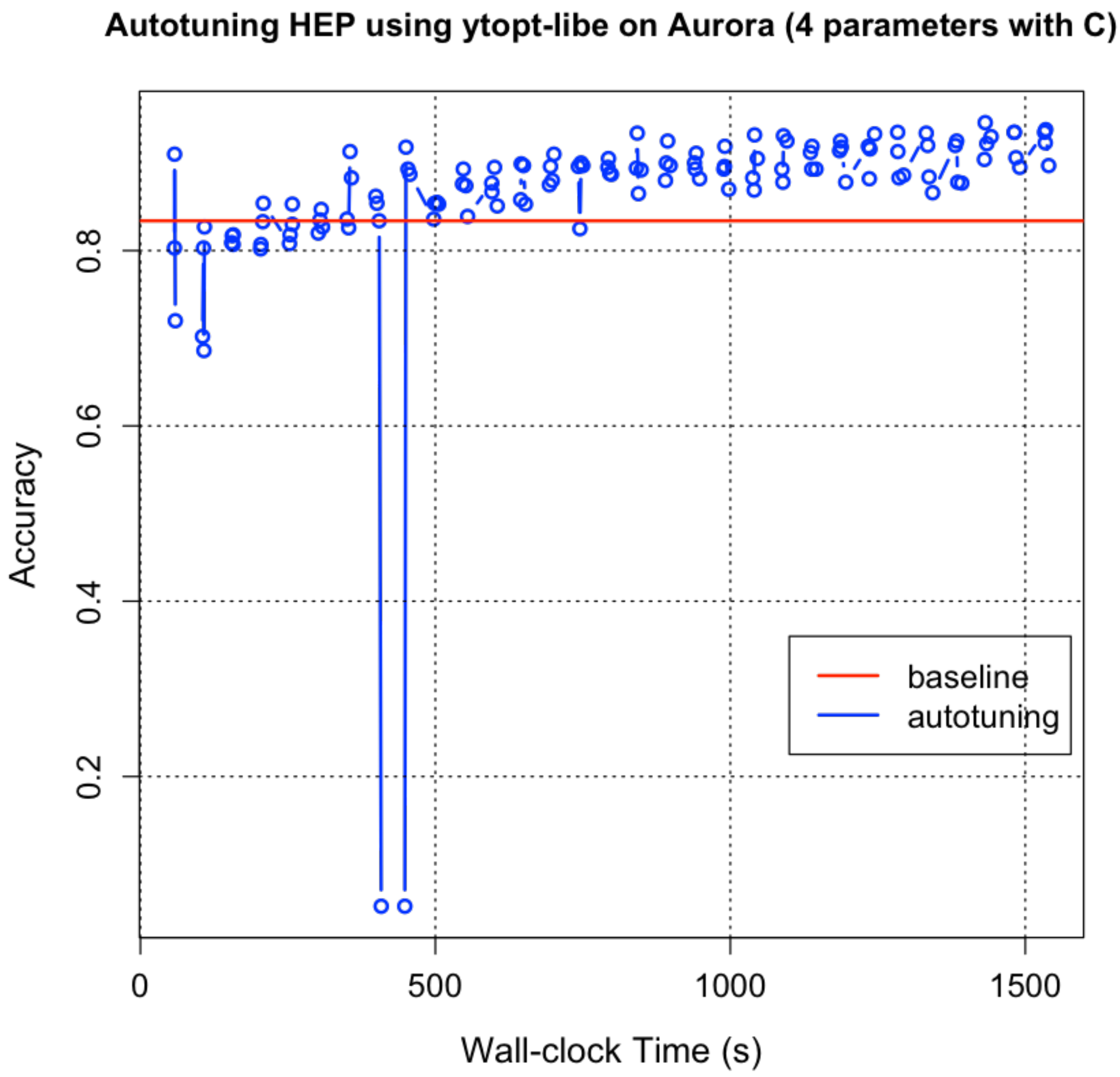
  - the parameter coef0
    - Significant in the sigmoid kernel
    - The default value 0.0
    - Positive or negative real number
    - Set the large range [-15, 15]

- Some abnormal cases with the very low accuracy 5.2%
- Why? This motivates us to investigate what really happened here

Autotuning the HEP Application with 3 parameters with C=1.0 and coef0=0.0

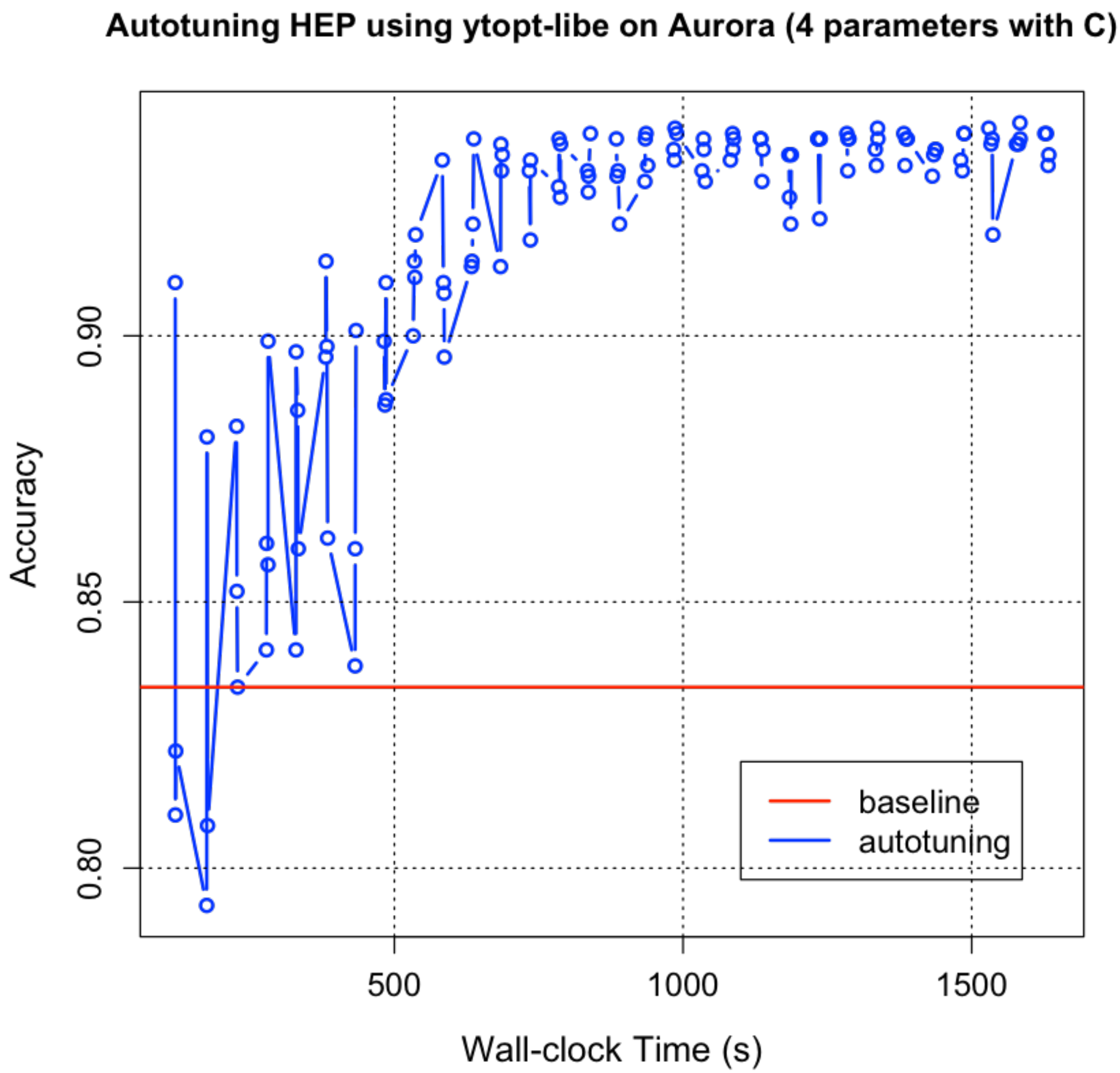


Autotuning the HEP Application with 4 parameters with C=[0.1, 100] and coef0=0.0



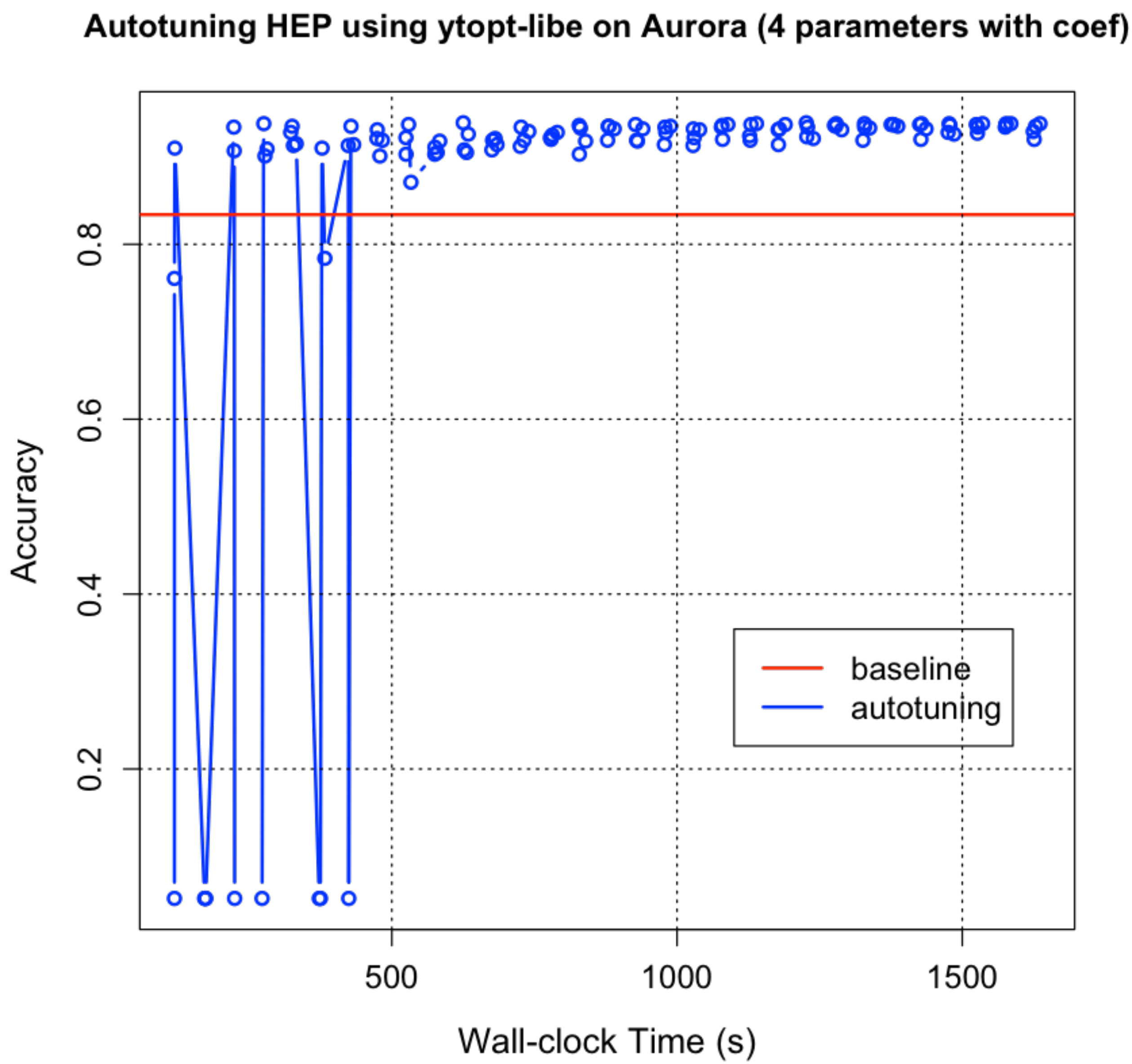
Argonne NATIONAL LABORATORY

Autotuning the HEP Application with 4 parameters with C=[0.37, 10] and coef0=0.0

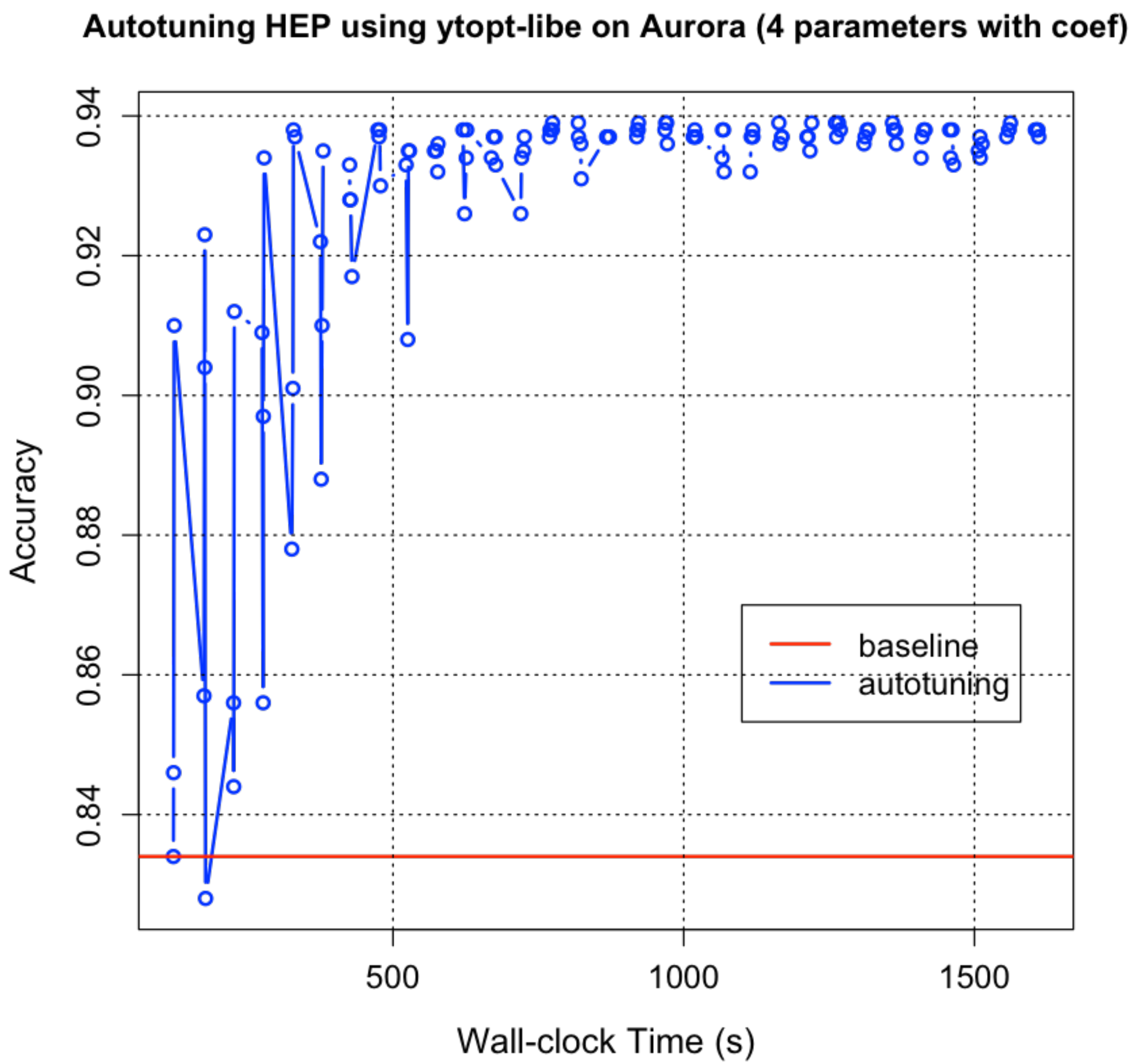


Argonne NATIONAL LABORATORY

Autotuning the HEP Application with 4 parameters with C=1.0 and coef0=[-15, 15]

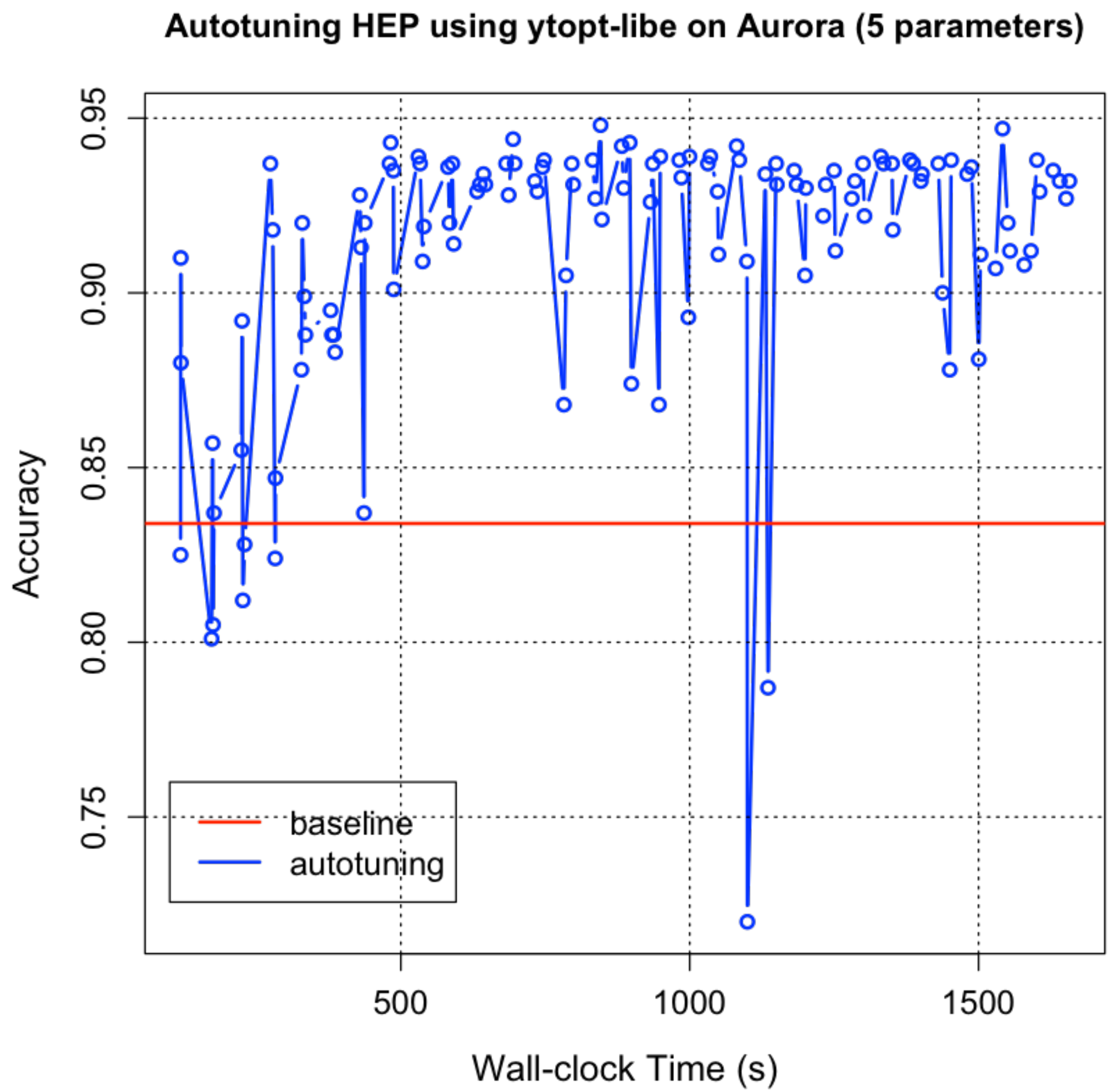


Autotuning the HEP Application with 4 parameters with C=1.0 and coef0=[-1, 1]



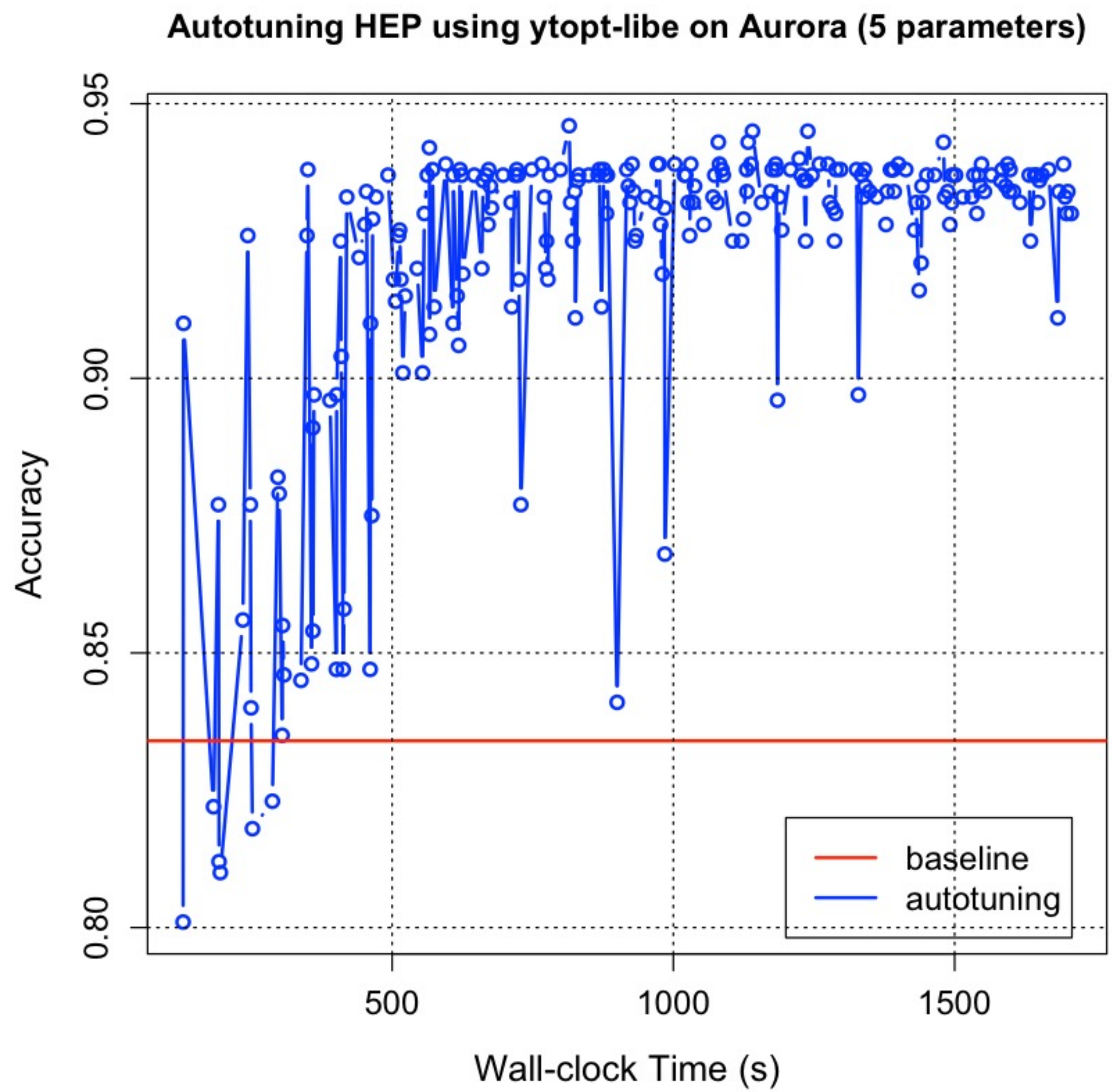
Argonne NATIONAL LABORATORY

Autotuning the HEP Application with 5 parameters with C=[0.37, 10] and coef0=[-1, 1] on 4 Aurora nodes



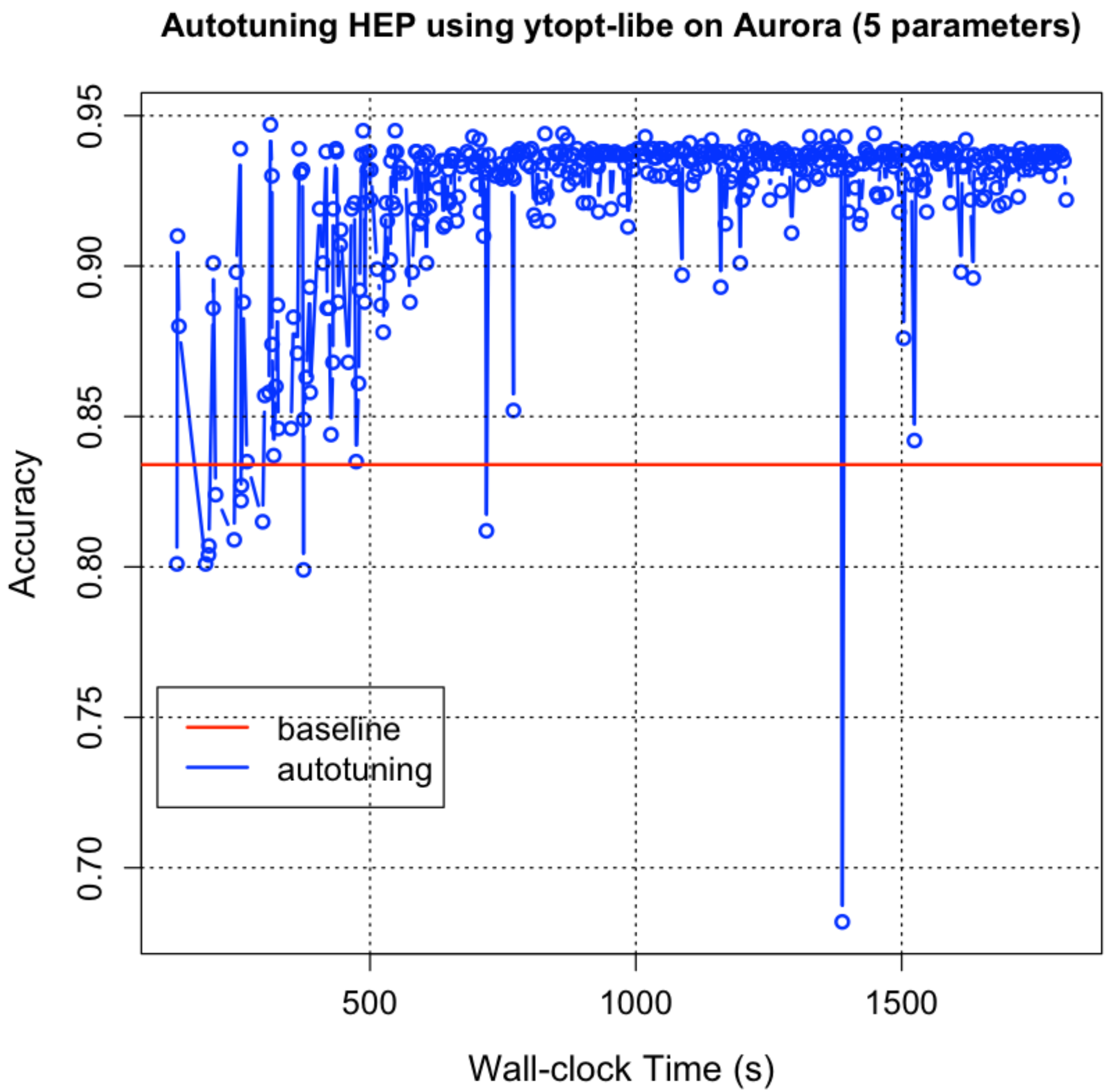
Accuracy:
94.8%

Autotuning the HEP Application with 5 parameters with C=[0.37, 10] and coef0=[-1, 1] on 8 nodes



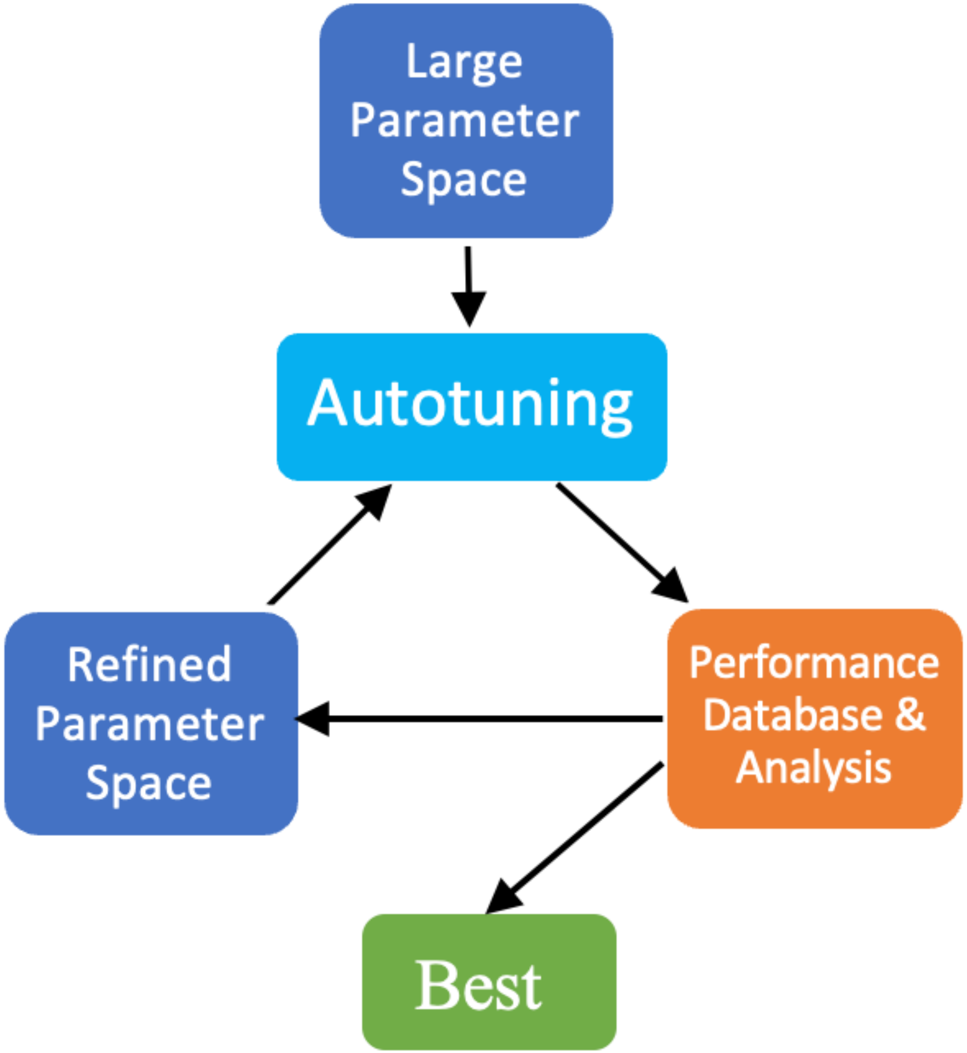
Accuracy: 94.8%

Autotuning the HEP Application with 5 parameters with C=[0.37, 10] and coef0=[-1, 1] on 16 nodes



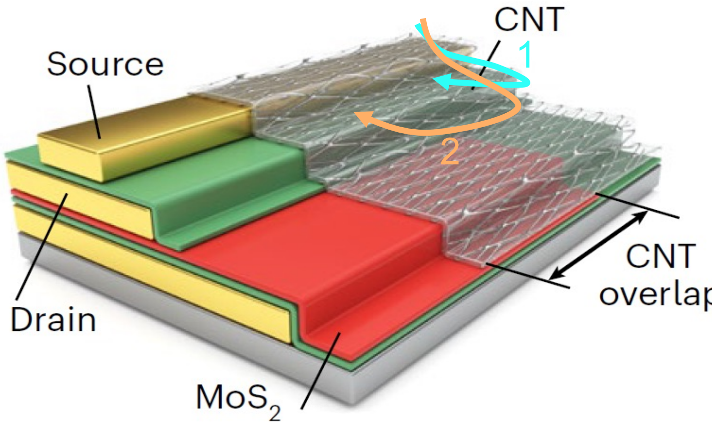
Accuracy: 94.8%

Autotuning-based HPO Framework



Mixed Kernel Heterojunction (MKH) Transistor

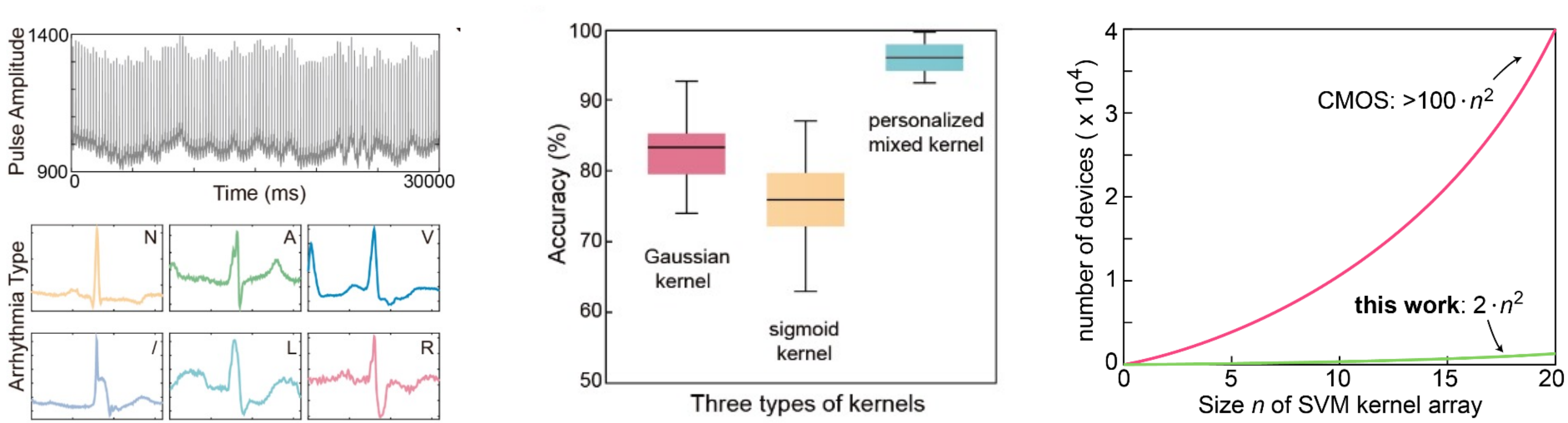
- CNT-MoS_2 vdW p-n heterojunctions with *lateral offsets*
- Dual-gated structure achieves tunable Gaussian and sigmoid kernel functions for efficient SVM classification



*Yan, Qian, Sangwan, Hersam et al., Nature Electronics **6**, 862 (2023)*

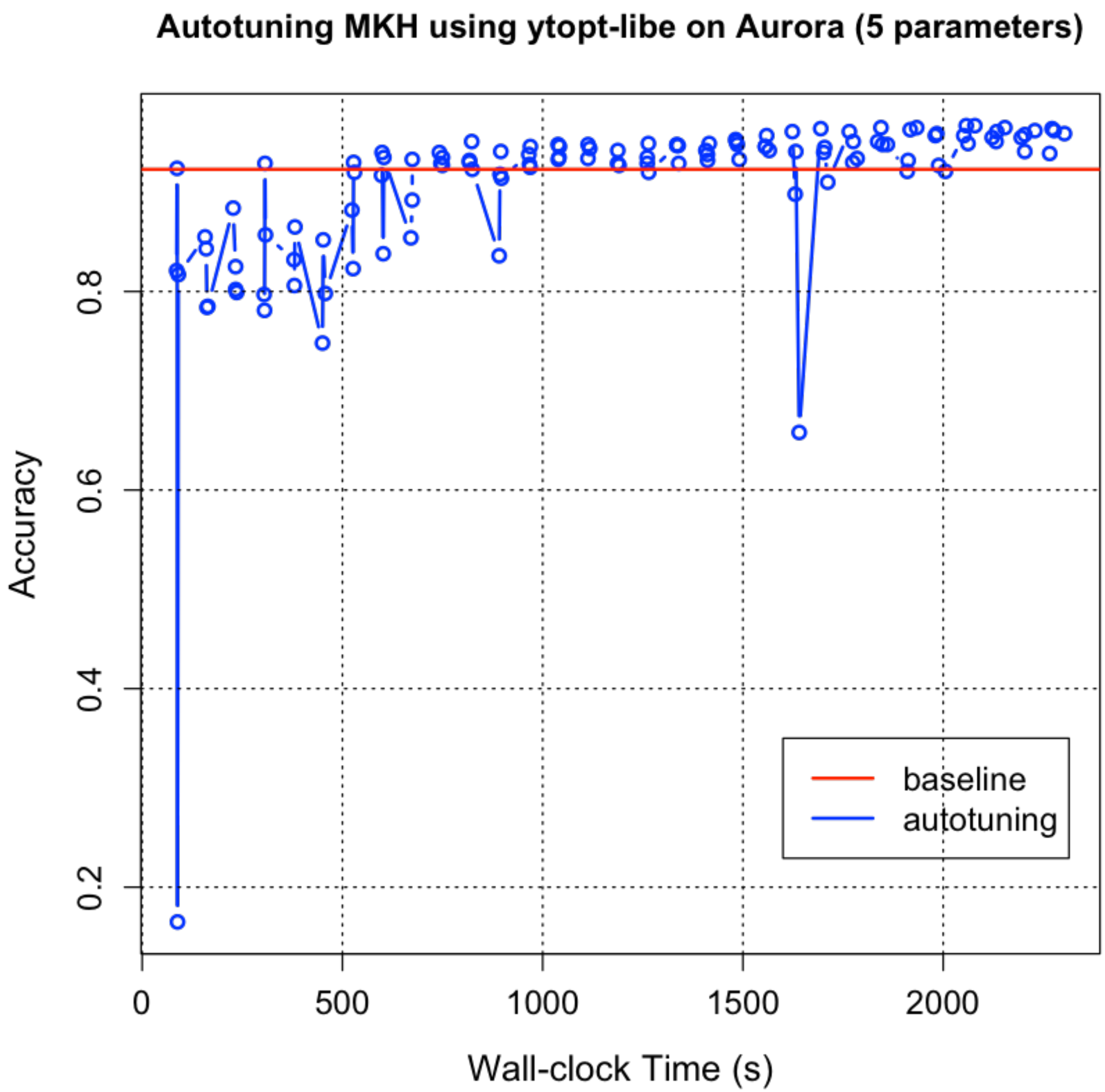
Heterojunction Transistor—Applications

- Two devices → full tunability of all hyperparameters
- SVM with mixed kernels → efficient arrhythmia detection in electrocardiogram (ECG) signal

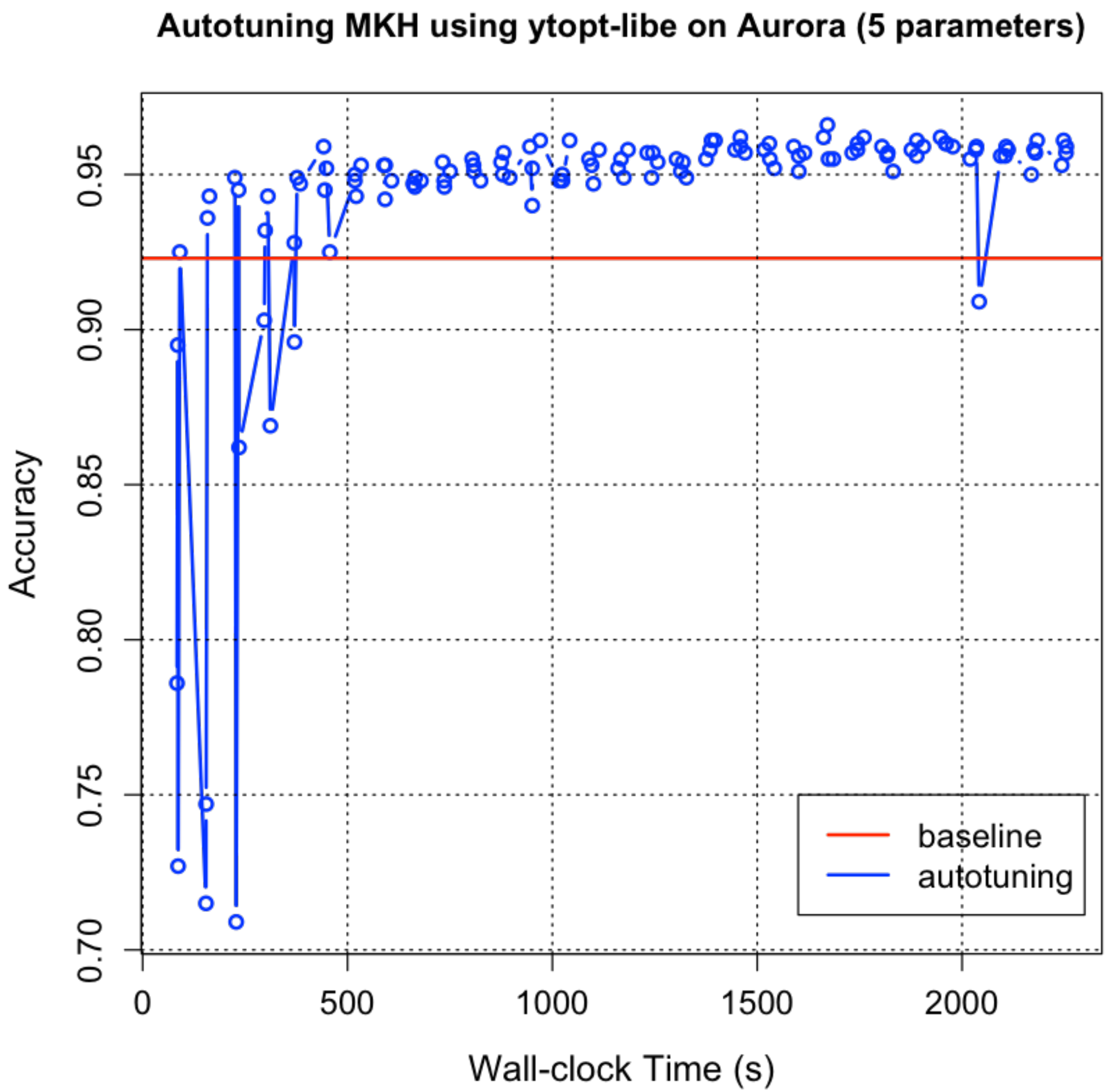


*Yan, Qian, Sangwan, Hersam et al., Nature Electronics **6**, 862 (2023) with **less than 95% average accuracy***

Autotuning the MKH Application with 5 parameters with C=[0.1, 100] and coef0=[-15, 15] on 4 nodes

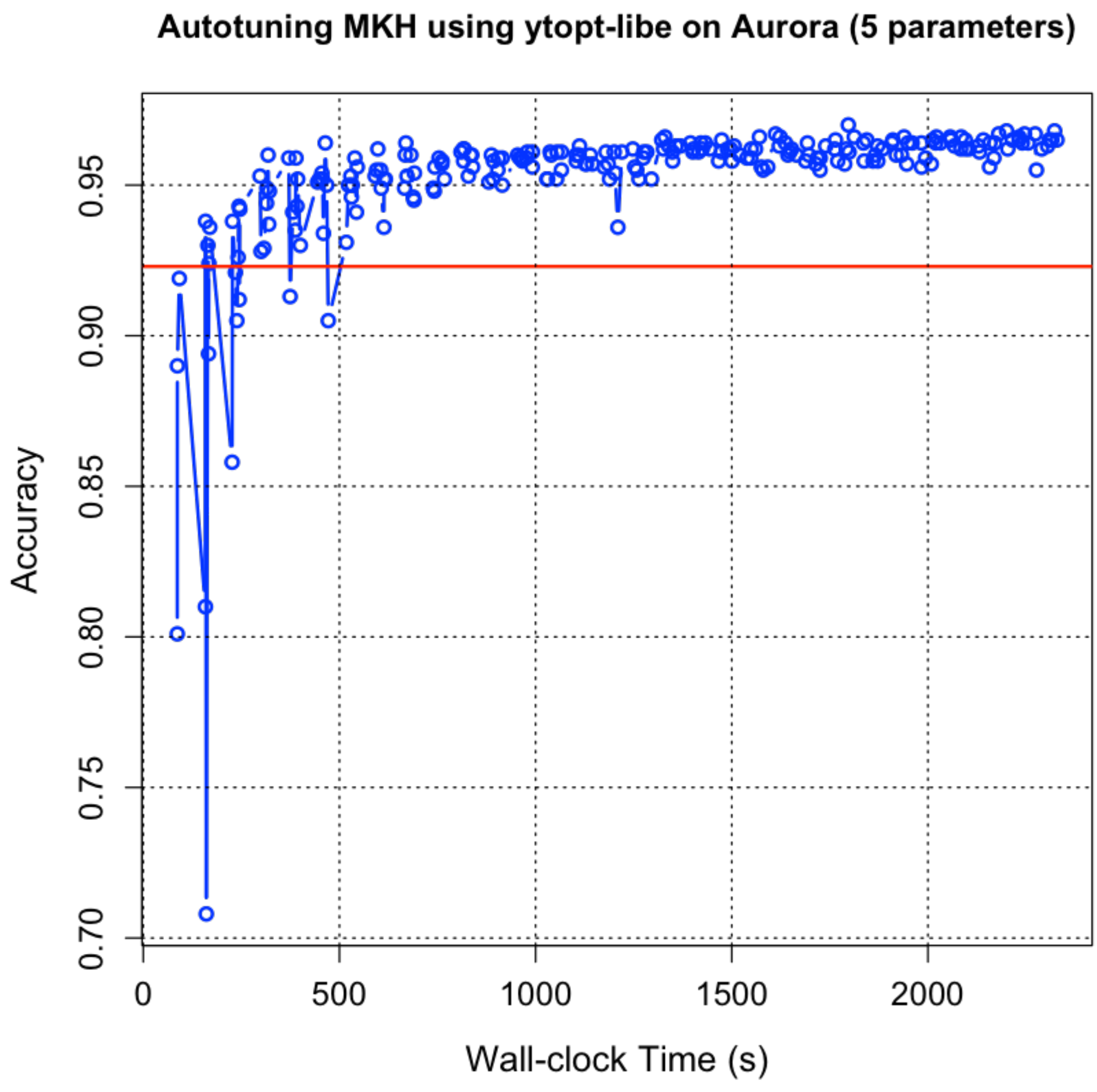


Autotuning the MKH Application with 5 parameters with C=[0.37, 10] and coef0=[-1, 1] on 4 nodes



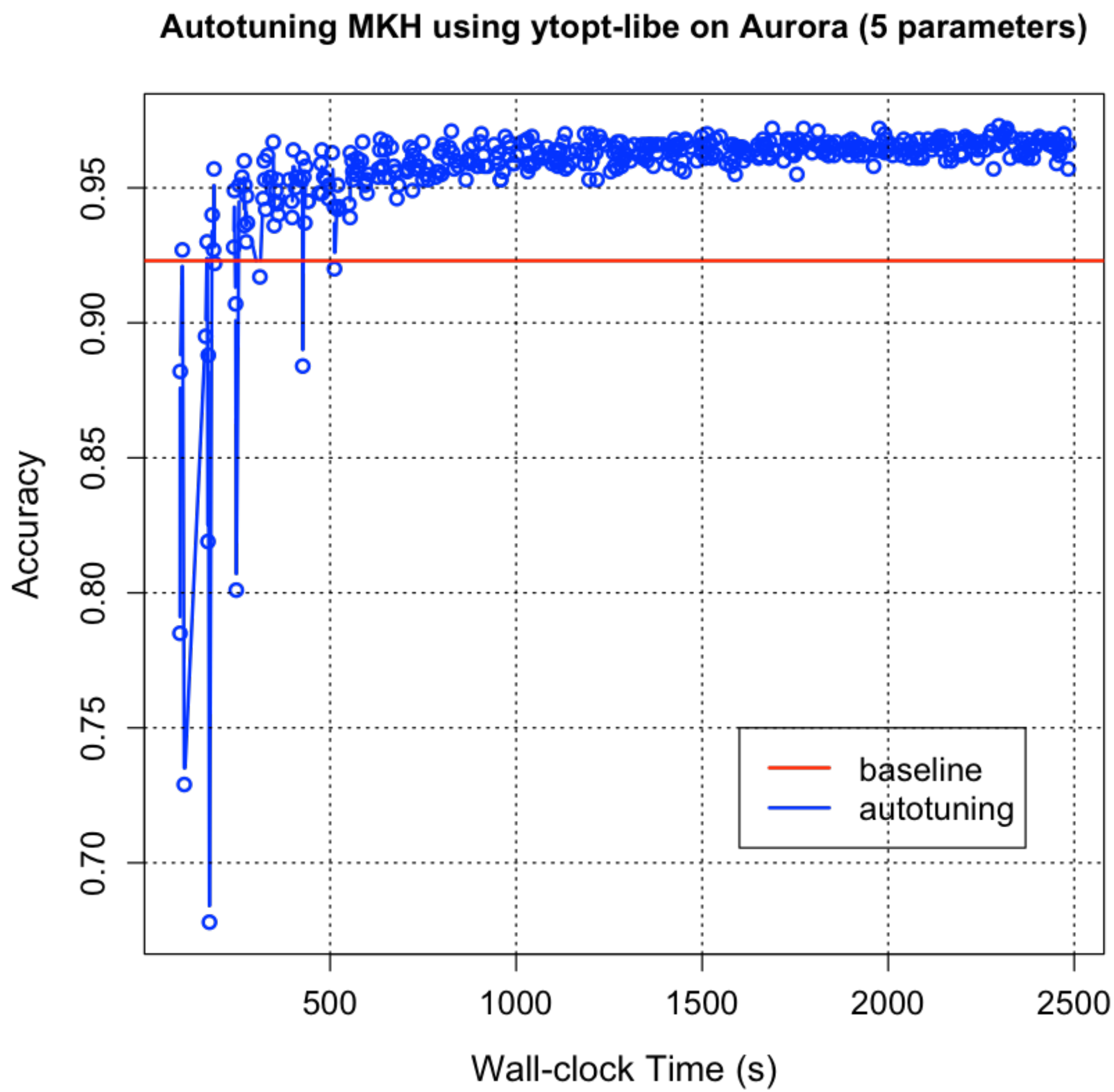
Accuracy: 96.6%

Autotuning the MKH Application with 5 parameters with C=[0.37, 10] and coef0=[-1, 1] on 8 nodes



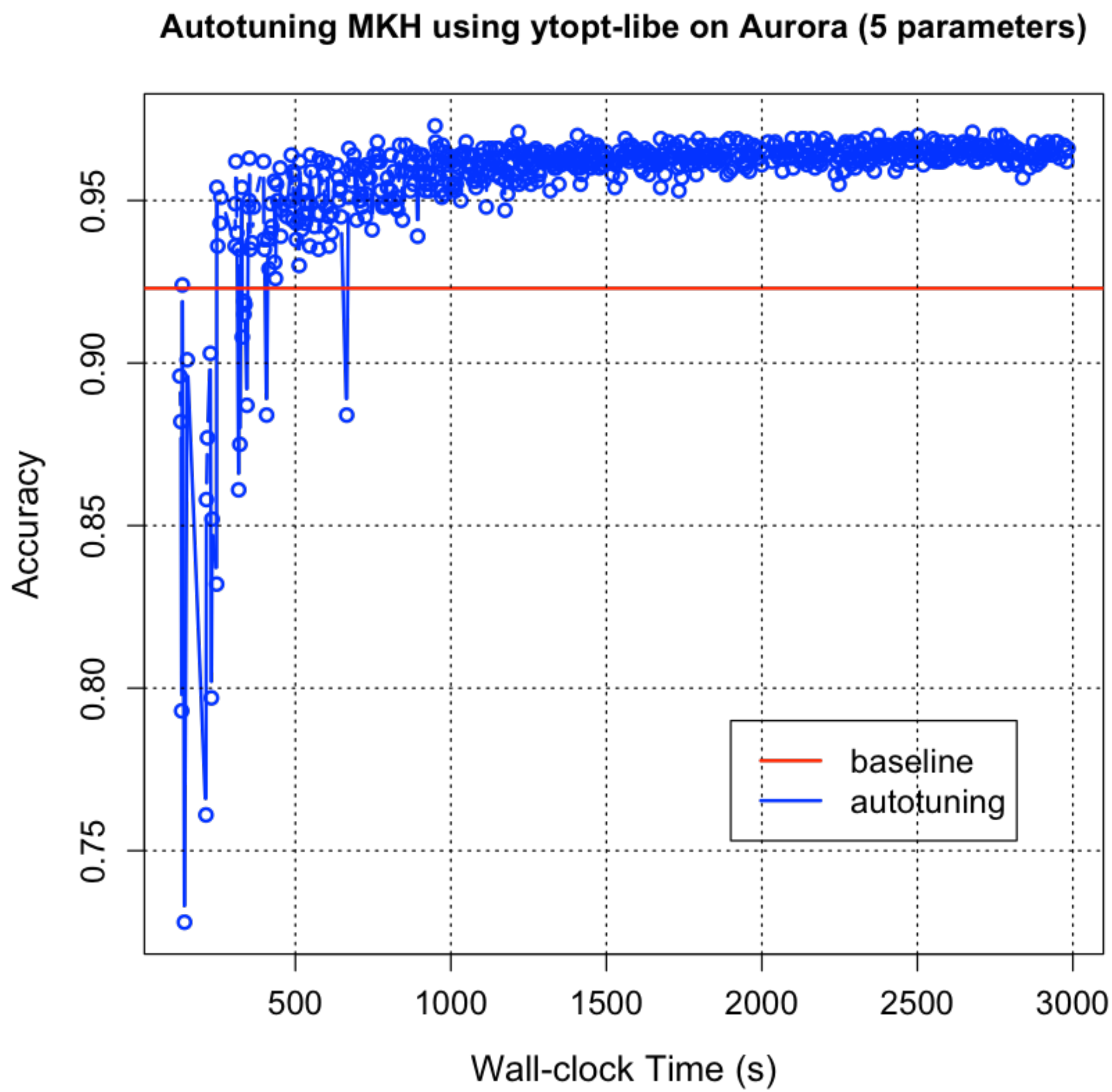
Accuracy: 97%

Autotuning the MKH Application with 5 parameters with C=[0.37, 10] and coef0=[-1, 1] on 16 nodes



Accuracy: 97.3%

Autotuning the MKH Application with 5 parameters with C=[0.37, 10] and coef0=[-1, 1] on 32 nodes



Accuracy: 97.3%

Summary

- We proposed the autotuning-based HPO framework for the mixed-kernel SVM applications in science and engineering on Aurora

- Our experimental results on Aurora show that
  - the optimal selection of hyperparameters in SVMs greatly varies for different applications and datasets
  - Uninformed choices of hyperparameters C and coef0 in the mixed-kernel SVMs resulted in severely low accuracy (5.2%)
  - The proposed framework achieved the highest accuracy for both mixed-kernel SVM applications.

- The implementation codes for both applications are available on our GitHub repo (https://github.com/ytopt-team/ytopt/tree/main/ytopt-libe/svms)

- Futher work: LLM-based HPO (Can AI replace our tools?)

Acknowledgements

This work was supported in part by

– DoE ASCR Microelectronics projects Threadwork and BIA

– DoE ASCR SciDAC RAPIDS3