# Analyzing Search Techniques for Autotuning Image-based GPU Kernels: The Impact of Sample Sizes

**Jacob O. Tørring**, Anne C. Elster

Department of Computer Science
Norwegian University of Science and Technology (NTNU)

Norwegian University of Science and Technology

# Motivation

- Heterogeneous systems: GPUs
- Optimizing portable GPU code
- Searching for the optimal configuration
- Limited budget for searching
- Which algorithm to choose, and when?

# Contributions

— Comparing metaheuristic optimization algorithms against Bayesian optimization-based search.
  - Bayesian Optimization based on Gaussian Processes
  - Bayesian Optimization based on Tree-Parzen Estimators
  - Genetic Algorithms

— Present tools to make statistically significant comparison
  - Non-parametric significance tests
  - Effect size measures
  - Statistics library

— Comparing related work in autotuning and hyperparameter optimization.

# Outline

# Search algorithms: direct search

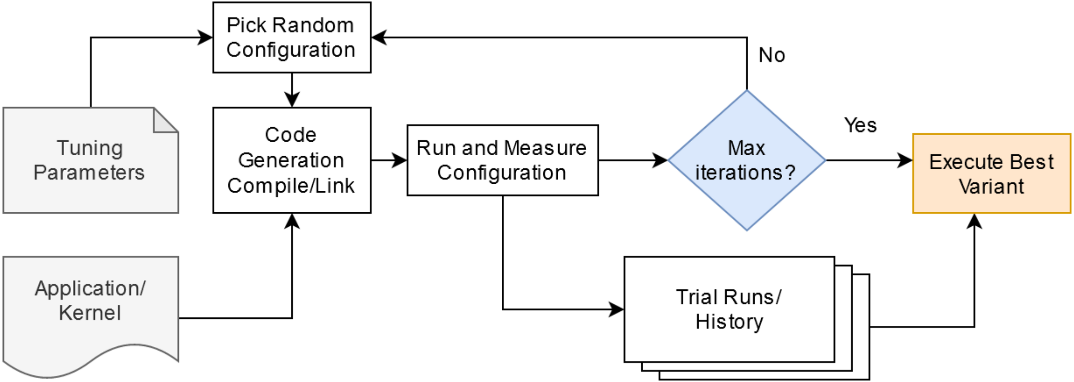



Figure: Pipeline for random search

# Search algorithms: model-based search

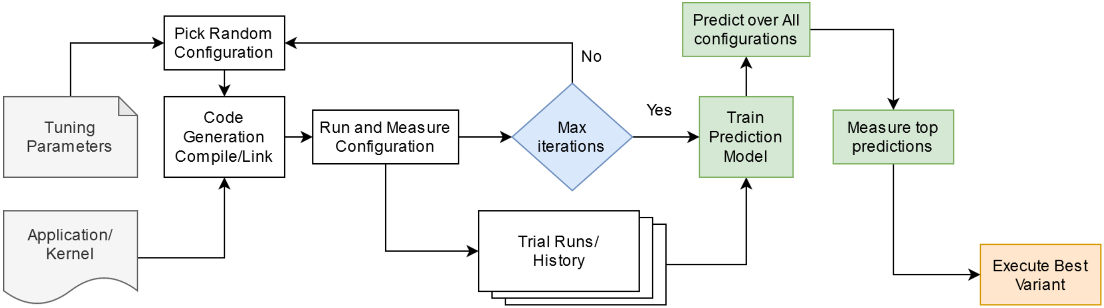



Figure: Pipeline for model-based search

# Search algorithms: Sequential Model-based Optimization

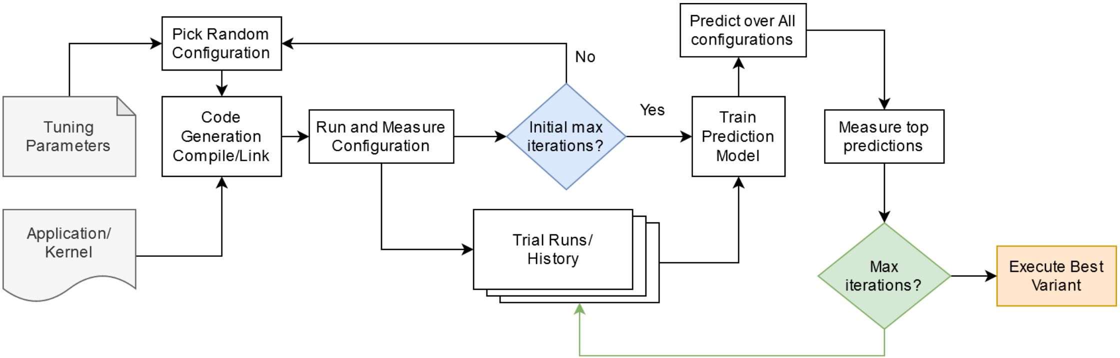



Figure: Pipeline for Sequential Model-Based Optimization

# Algorithms used in study

— Direct Search
  - Random Search (RS)

— Model-based search
  - Random Forests (RF)

— Sequential Model-Based Optimization (SMBO)
  - Bayesian Optimization based on Gaussian Processes (BO-GP)
  - Bayesian Optimization based on Tree-Parzen estimators (BO-TPE)
  - Genetic Algorithms (GA)

— Some of the best performing techniques from Autotuning literature[1] and Hyperparameter Optimization literature[2]

[1] Ben van Werkhoven. Kernel Tuner: A search-optimizing GPU code auto-tuner. en. In: Future Generation Computer Systems 90 (Jan. 2019), pp. 347358. ISSN: 0167-739X. DOI: 10.1016/j.future.2018.08.004. URL: http://www.sciencedirect.com/science/article/pii/S0167739X18313359

[2] James S. Bergstra et al. Algorithms for Hyper-Parameter Optimization. In: Advances in Neural Information Processing Systems 24. Ed. by J. Shawe-Taylor et al. Curran Associates, Inc., 2011, pp. 25462554. URL: http://papers.nips.cc/paper/4443-algorithms-for-hyper-parameter-optimization.pdf

# Benchmarks

- ImageCL[3]: Compiles to OpenCL
- Add benchmark: C = A + B
- Harris benchmark: Corner detection algorithm
- Mandelbrot benchmark: Generating a visualization of the Mandelbrot set.
- Thread dimensions: $\{X, Y, Z\}_t = [1..16]$
- Work group size: $\{X, Y, Z\}_w = [1..8]$
- $dim(S) = 6$, $|S| = 2\,097\,152$ configurations.
- Same benchmarks as previous ImageCL-based autotuning studies.

[3] Thomas L. Falch and Anne C. Elster. ImageCL: An image processing language for performance portability on heterogeneous systems. In: 2016 International Conference on High Performance Computing Simulation (HPCS). July 2016, pp. 562569. DOI: 10.1109/HPCSim.2016.7568385.

# Hardware

- Nvidia GTX 980
- Nvidia Titan V
- Nvidia RTX Titan
- Hardware from older to newer generations of hardware to investigate generational difference.

# Comparability

- Using significance tests to assess our results
- Most significance tests assume some parameterized distribution of the samples
- E.g. a gaussian/normal-distribution.
- Can we use these techniques for our autotuning studies?

# Distribution of samples: Mandelbrot benchmark

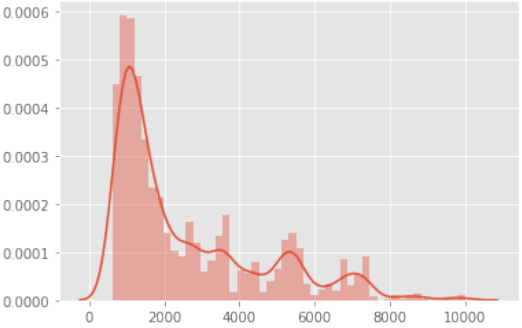



Figure: GTX980 Probability distribution of all samples

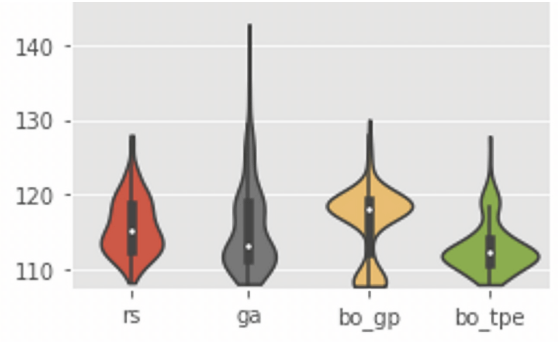



Figure: Titan V Probability distribution of results from algorithms

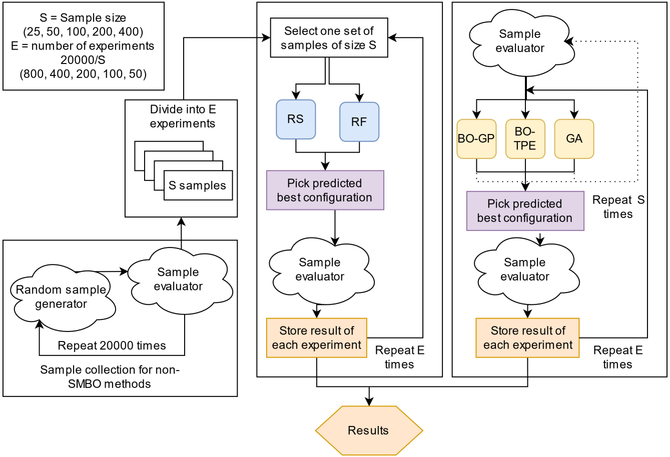
# Non-parametric significance tests

- Population are obviously non-gaussian.
- Cannot be modeled accurately with any distribution from the SciPy statistics package.
- Cannot make any assumptions about the underlying distribution, so we need a non-parametric significance test.
- Bootstrapping would drastically increase the experiment time.
- We propose to use the Mann-Whitney U (MWU)[4]
- Using the Common Language Effect Size: The likelihood of one algorithm outperforming another
- Using the Pingouin library[5]

[4] Andrea Arcuri and Lionel Briand. A practical guide for using statistical tests to assess randomized algorithms in software engineering. en. In: Proceeding of the 33rd international conference on Software engineering - ICSE 11. Waikiki, Honolulu, HI, USA: ACM Press, 2011, p. 1. ISBN: 978-1-4503-0445-0. DOI: 10.1145/1985793.1985795. URL: http://portal.acm.org/citation.cfm?doid=1985793.1985795

[5] Raphael Vallat. Pingouin: statistics in Python. In: Journal of Open Source Software 3.31 (Nov. 2018), p. 1026. ISSN: 2475-9066. DOI: 10.21105/joss.01026. URL: http://joss.theoj.org/papers/10.21105/joss.01026

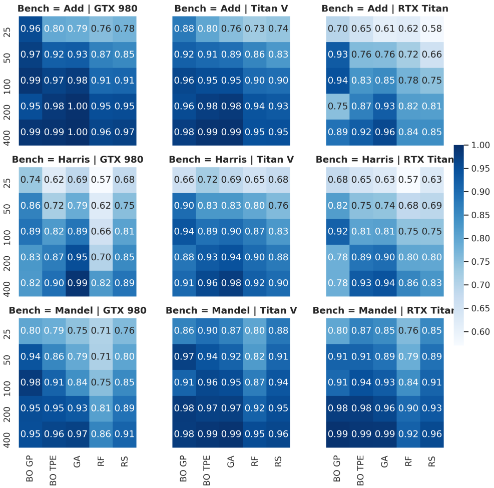
# Autotuning Experiment Structure

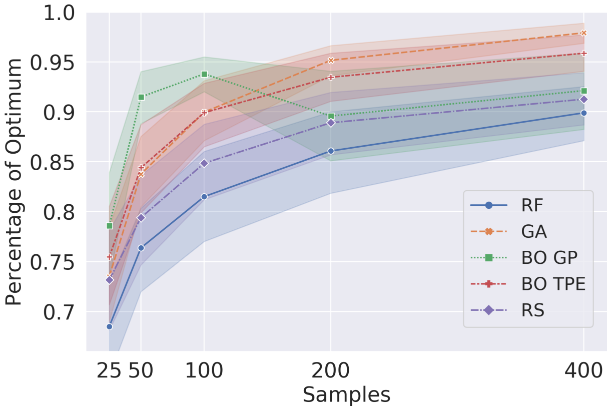


# Overview of related work

| Author | Samples/Experiments/Evaluations [a] | Significance test | Research field | Algorithms |
|---|---|---|---|---|
| Hutter et al. [10] | 30-300 Min / 25 / 1000 | Mann-Whitney U | AlgConf | SMAC, ROAR, TB-SPO, GGA(GA) |
| Eggensperger et al. [21] | Varies[b] (50 to 200) / 10 / n/a | Unpaired t-test | AlgConf | BO TPE, SMAC, Spearmint |
| Falkner et al. [22] | Varies[b] / Varies / Varies | n/a | AlgConf | RS, BO TPE, BO GP, HB, HB-LCNet and BOHB |
| Snoek et al. [7] | Varies[b] (1-50,1-100) / 100 / n/a | n/a | HypOpt | BO GP, Grid search |
| Bergstra et al. [8] | 230 / 20 / n/a | n/a | HypOpt | RS, BO TPE, BO GP, Manual |
| Bergstra et al. [23] | 1-128 / 256-2 / n/a | n/a | HypOpt | RS, Grid Search(GS) |
| Bergstra et al. [6] | 10-200 / n/a / n/a | n/a | HypOpt | Boosted Regression Trees, GS, Hill Climbing |
| Falch and Elster [5] | 100-6000 / 20 / n/a | n/a | Autotuning | NN, SVR, Regression Tree |
| van Werkhoven [12] | Varies[b] / 32 / 7 | n/a | Autotuning | Many Metaheuristic Methods |
| Willemsen et al.[24] | 20-220 / 35 / n/a | n/a | Autotuning | BO, RS, SA, MLS and GA |
| Ansel et al. [25] | Varies[b] / 30 / n/a | n/a | Autotuning | Multi-armed bandit, Manual |
| Nugteren et al. [11] | Varies[b] (107 or 117)/ 128 / n/a | n/a | Autotuning | RS, SA, PSO |
| Akiba et al. [26] | Varies[c] / 30 / n/a | "Paired MWU" | Autotuning | RS, HyperOpt, SMAC3, GPyOpt, TPE+CMA-ES |
| Grebhahn et al. [27] | 50, 125 / Unclear [d] / n/a | "Wilcox test" | SBSE | RF, SVR, kNN, CART, KRR, MR |
| Tørring | 25-400 / 800-50 / 10 | Mann-Whitney U | Autotuning | RS, BO TPE, BO GP, RF, GA |

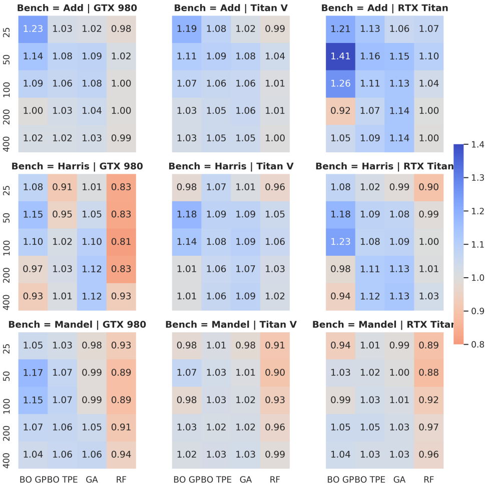
# Results: Convergence to optimum performance

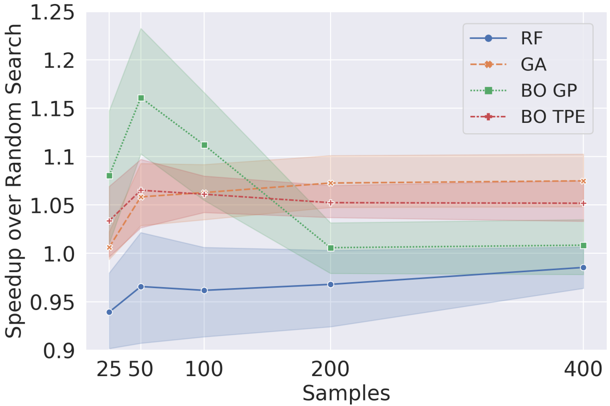



All results with a margin of more than 1% are statistically significant under the MWU test with $\alpha = 0.01$.

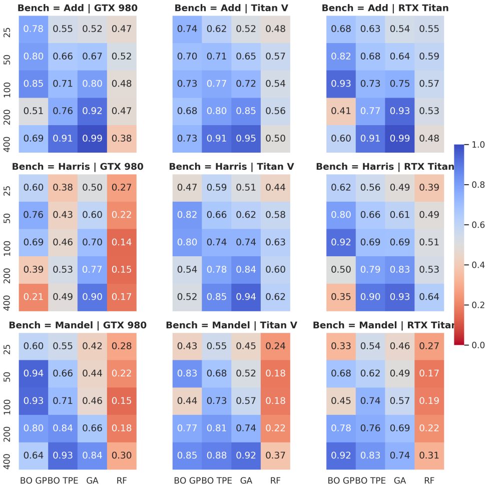
# Results: Median performance

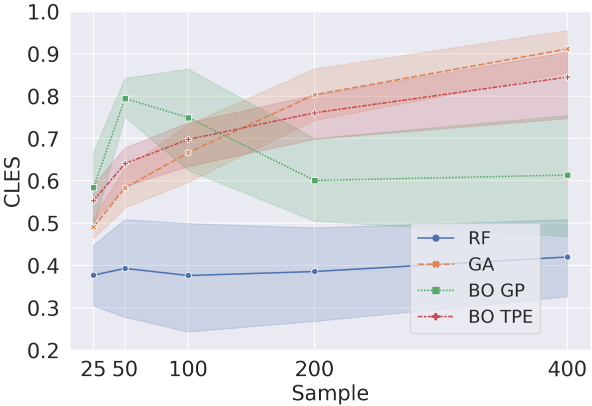



All results with a margin of more than 1% are statistically significant under the MWU test with $\alpha = 0.01$.

# Results: CLES over Random Search




All results with a margin of more than 1% are statistically significant under the MWU test with $\alpha = 0.01$.

## Discussion

- Generally BO GP performs the best for lower sample sizes
- Generally GA performs best for higher sample sizes
- Our use of BO GP seems to overfit, indicating that a better implementation of BO might perform better
- Results vary between benchmarks and hardware architectures, but there is a consistent trend
- Our benchmarks all have identical search spaces.
- Limited domain and only Nvidia GPUs

# Conclusion and Contributions

- Study on autotuning algorithms for Image-based GPU kernels.
  - Bayesian Optimization based on Gaussian Processes
  - Bayesian Optimization based on Tree-Parzen Estimators
  - Genetic Algorithms
- Presented Non-parametric significance tests and experiment setups which provides statistically significant results.
- Compare related work in autotuning and hyperparameter optimization.

# Future work

- Need for
  - more thorough benchmarking guidelines in autotuning.
  - comprehensive and representative benchmarking suites for autotuning[6].
- Performing new comparative studies with more sophisticated tools and a wider and more representative benchmark suite on a range of hardware configurations.

[6] Ingunn Sund, Knut A. Kirkhorn, Jacob O. Tørring and Anne C. Elster. BAT: A Benchmark suite for AutoTuners. In: Norwegian ICT-conference for research and education. 1. 2021, pp. 4457

*Thank you for listening!*

**Contact information**

Jacob O. Tørring: jacob.torring@ntnu.no

Anne C. Elster: elster@ntnu.no