

Generative and Multi-phase Learning for Computer Systems Optimization

Yi Ding Nikita Mishra Henry Hoffmann

Department of Computer Science, University of Chicago

Introduction

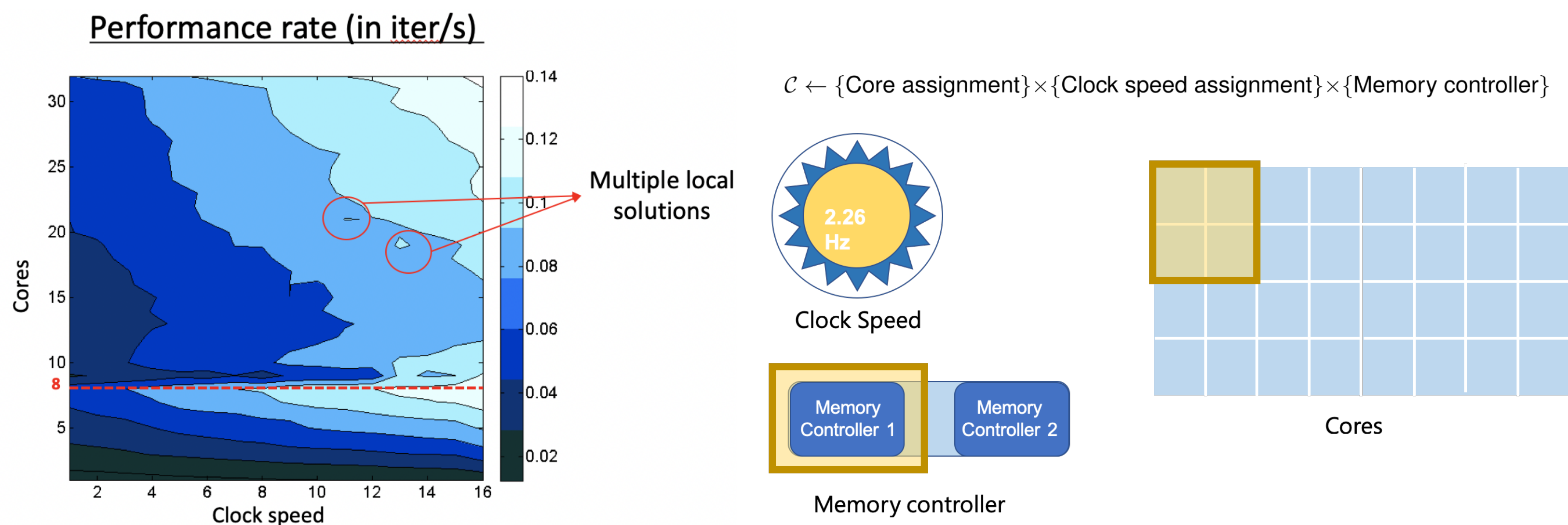
Optimizing modern computer systems requires tradeoff:

- Deliver reliable performance
- Minimize energy consumption

Resource management via system configuration:

- Resources have complex, non-linear effects on performance and energy
- Resource interactions create local optimal

How to find the optimal system configuration?



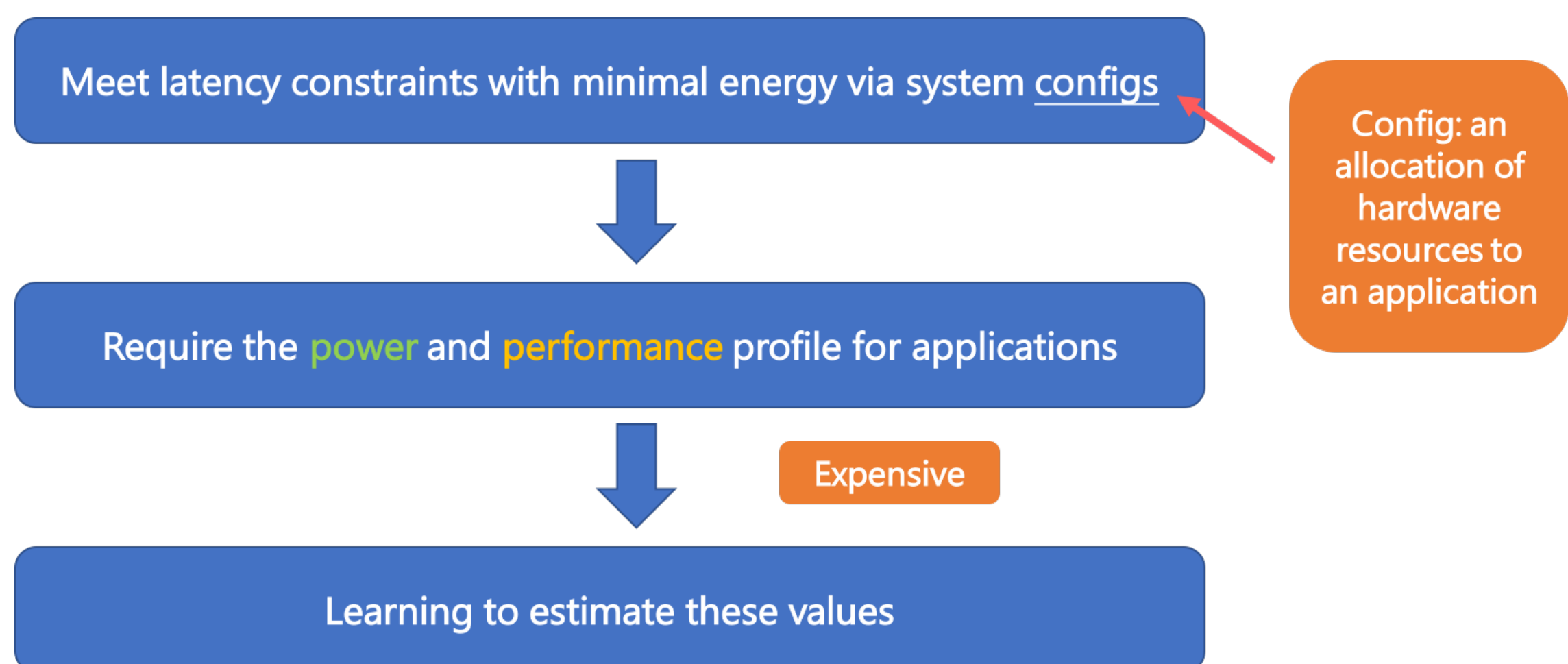
Left: A contour plot of performance rate for streamcluster benchmark at different configurations. Right: Example of a configuration space.

Machine Learning to the Rescue

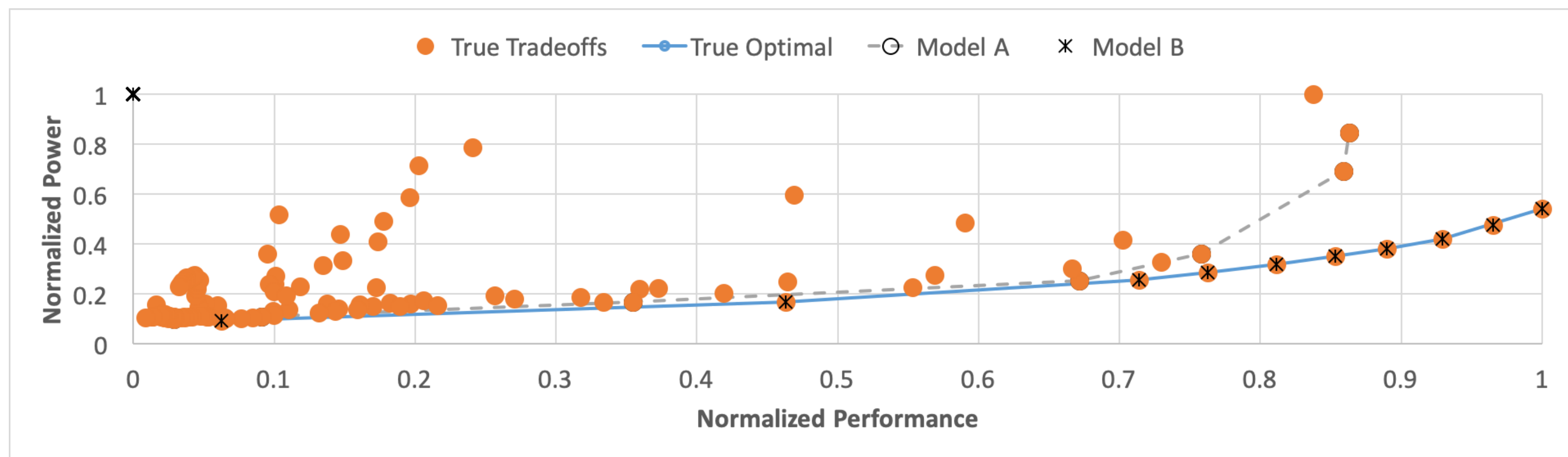
Machine learning replaces heuristics, but:

- Scarce data: expensive collection, limited range behavior → **Generative model**
- Asymmetric benefits: only configurations on optimal frontier useful → **Multi-phase sampling**

Problem Formulation



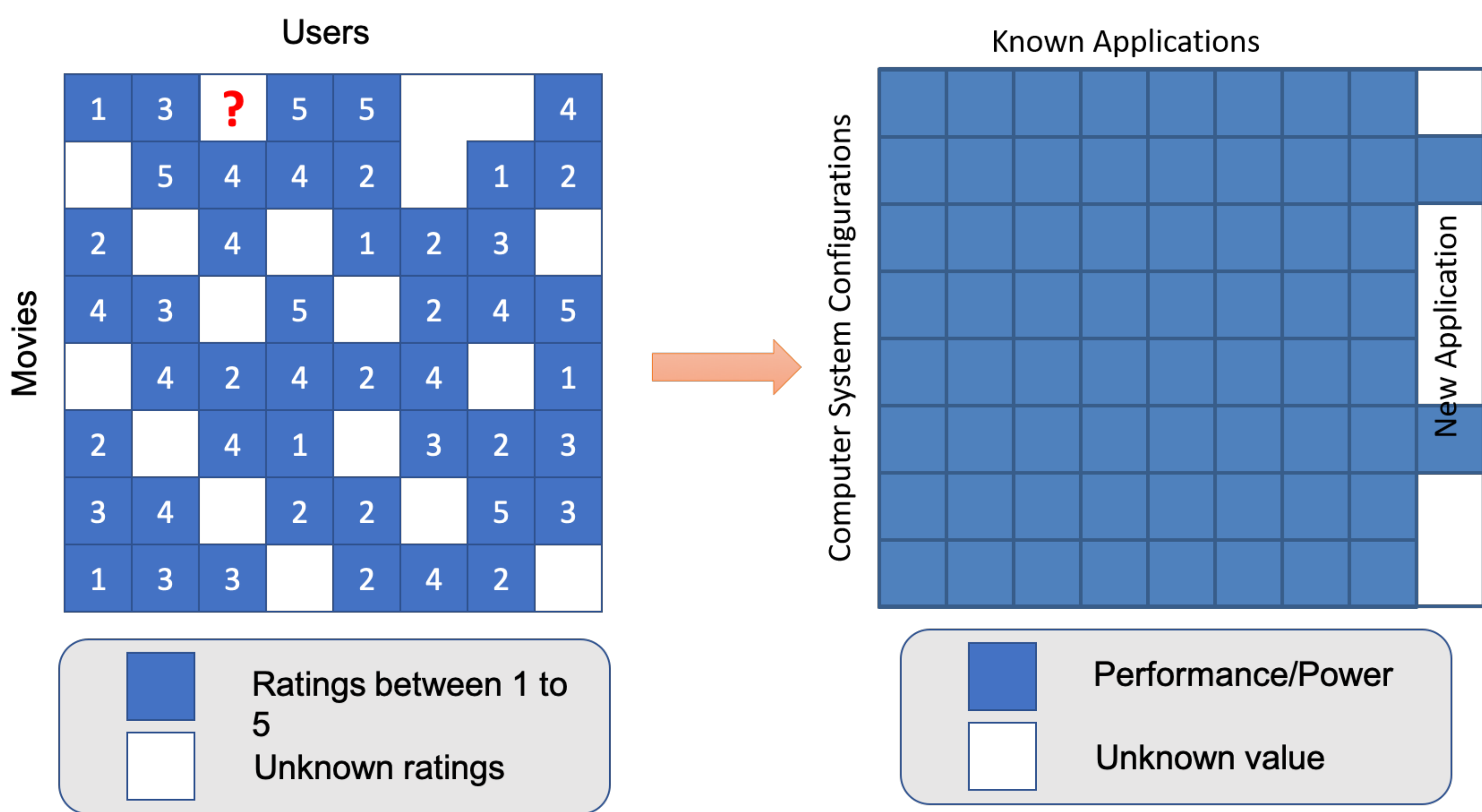
Motivational Example: SRAD on ARM big.LITTLE system



	Model A	Model B
Optimal points	Far enough	True data
Non-optimal points	True data	Far enough
Goodness of fit	99%	0
Energy over optimal	22% ❌	0 ✅

High prediction accuracy \neq good system results

Recommender Systems → Learning by Examples

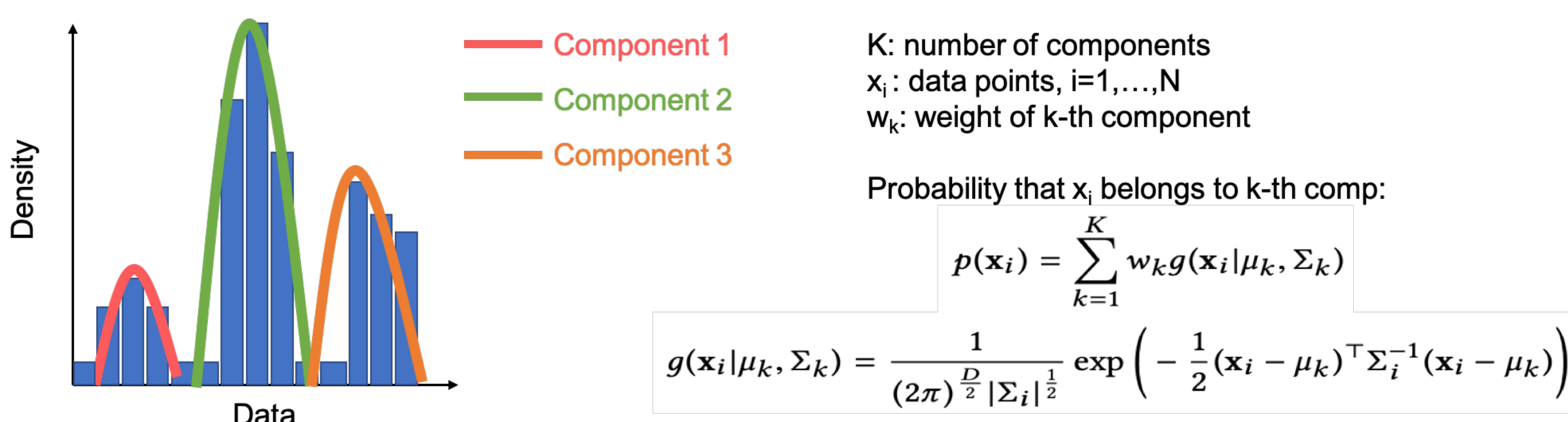


Generating Data for Accuracy

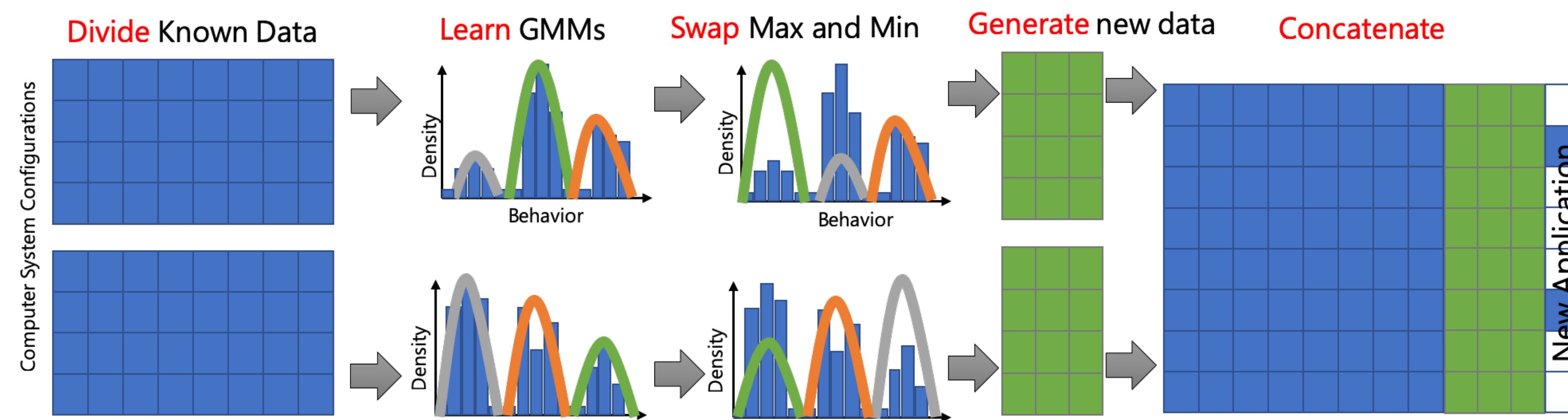
Goal: different enough but still realistic to be plausible

How:

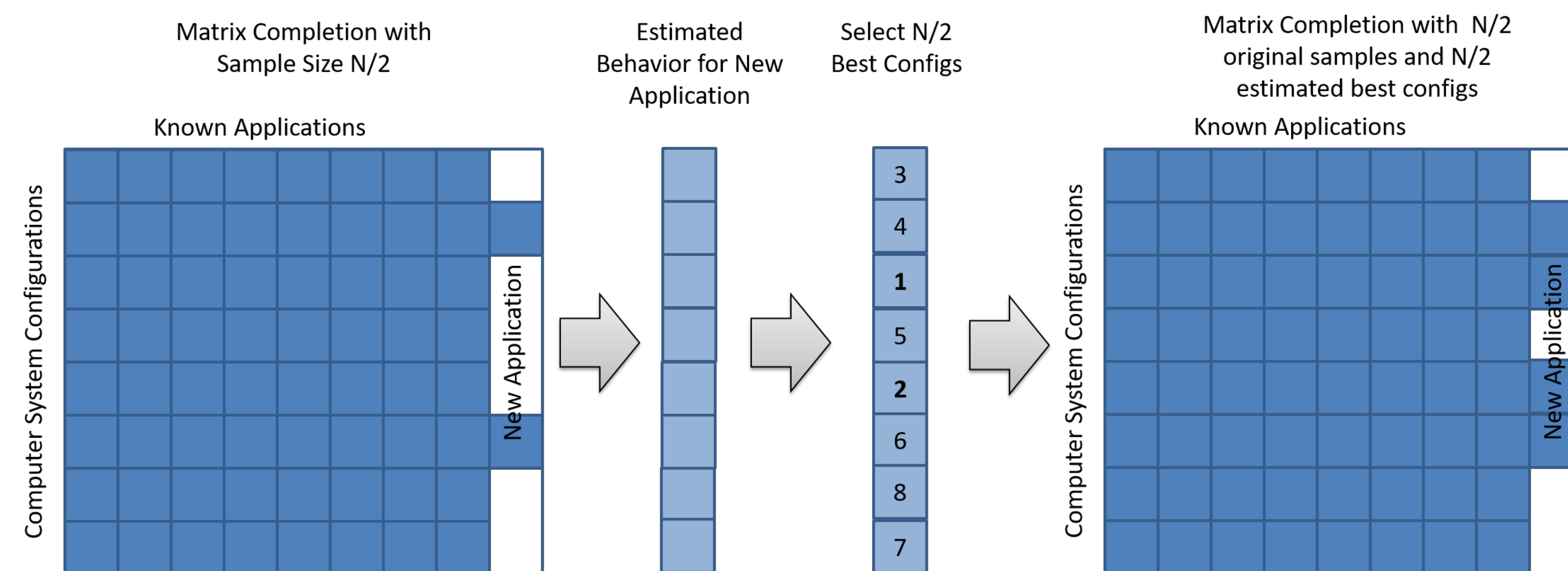
- Random number generator ← different but not plausible
- Gaussian Mixture Model (GMM) ← plausible but not different



Generating Data with a GMM



Multi-phase Sampling



Input: Known and unknown applications, sampling budget N .

while True do

Phase-1:

(1) Sample half of the budget $N/2$ configurations. (2) Run learner to get an initial estimation. (3) Rank configurations by estimated energy efficiency $\left[= \frac{\text{estimated performance}}{\text{estimated power}} \right]$.

Phase-2:

(1) Sample the $N/2$ most energy efficient configs. (2) Run learner again to obtain the final estimation.

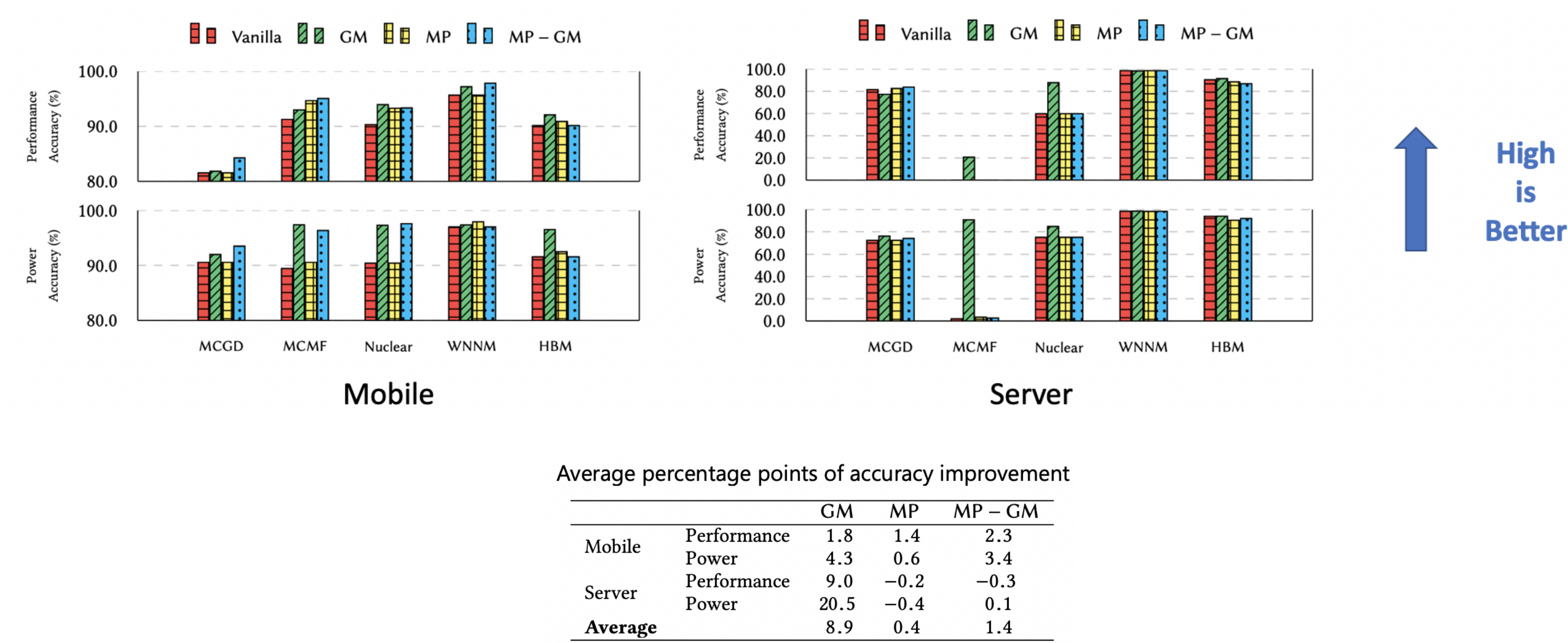
Output: Estimation of performance and power.

Experimental Setup

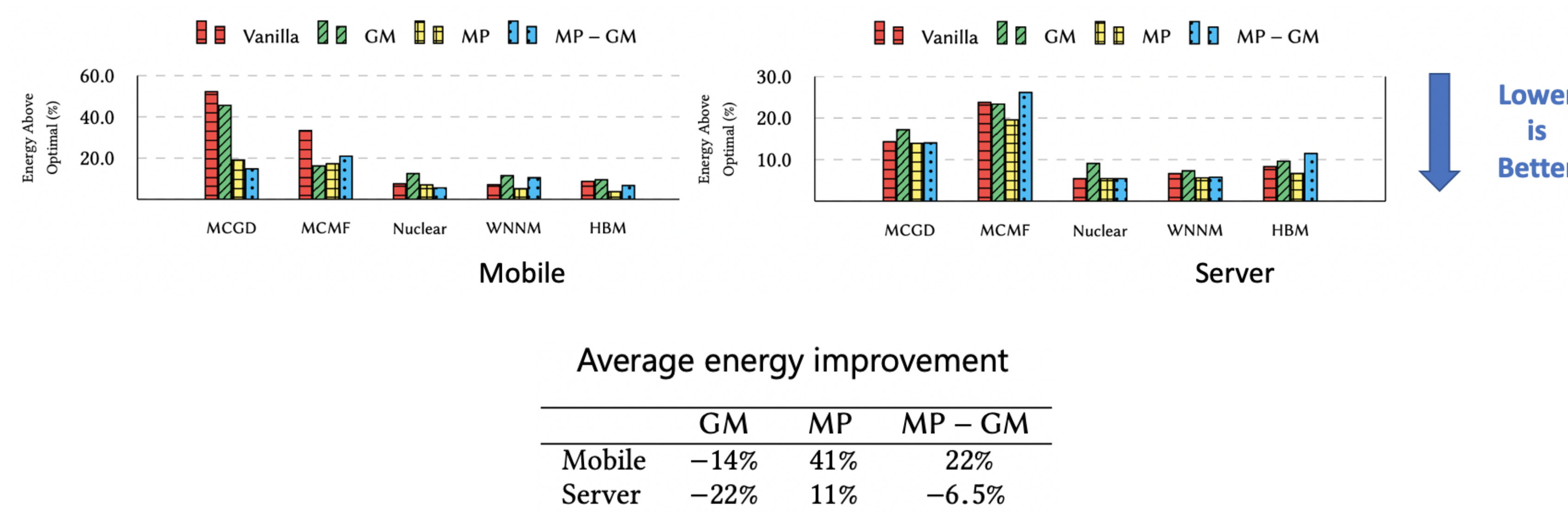
	Mobile	Server
System	Ubuntu 14.04	Linux 3.2.0 system
Architecture	ARM big.LITTLE	Intel Xeon E5-2690
# Applications	21	22
# Configurations	128	1024

Learning Models	Category	Frameworks	Meanings
MCGD	MC	Vanilla	Basic learners
MCMF	MC	GM	Generative model
Nuclear	MC	MP	Multi-phase sampling
WNNM	MC	MP-GM	Combine GM and MP
HBM	Bayesian		

Improve Prediction Accuracy w/ GM



Improve Energy Savings w/ MP



Conclusion

- Generative model improves prediction accuracy.
- Multi-phase method improves energy by biasing the learner to the configurations that are likely to be most energy efficient.
- Improving accuracy does not necessarily improve energy consumption.
- We advocate to de-emphasize accuracy but incorporate system structure into the learners.