Using Small-Scale History Data to Predict Large-Scale Performance of HPC Application

Wenju Zhou

University of Science and Technology of China

2020.05.12

USTC

Outline

- Background
- Two-level model
- Experiment results and analysis
- Conclusions

01 Background

High Performance Computing (HPC)

- architecture:
  computing nodes,
  interconnect, ...
- usage:
  physical simulations,
  molecular modeling, ...
- effect:
  provide computing power,
  reduce experimental risk, ...

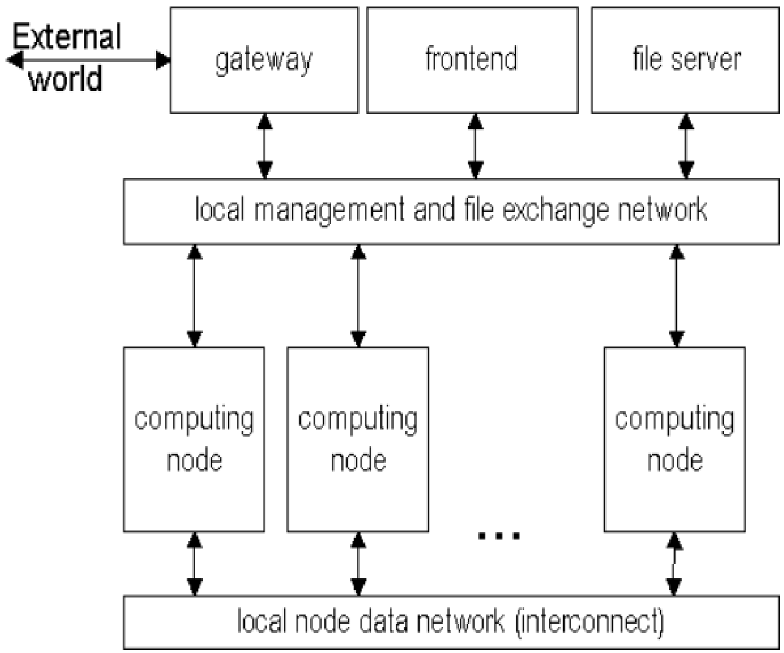


fig 1. architecture of HPC

Performance Modeling

features

· system settings
· application input
· execution options
· compiler flags
…

Model

performance

· execution time
· memory
· throughput
· energy
…

example

usage

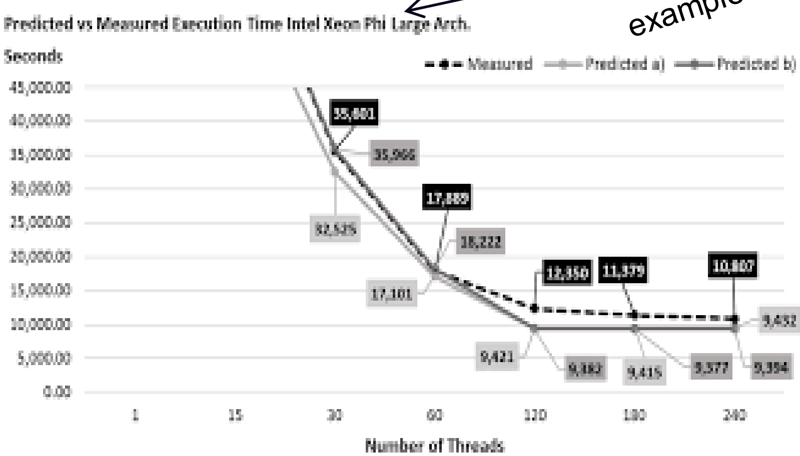


fig 2. architecture of HPC

· tuning
· scheduling
· energy saving
…

Performance Modeling

categories	desciption	advantaes	limitations
simulative methods	simulating execution by simulators	concept machine	overhead
analytical methods	analysis of code and system by experts	accuracy & Interpretability	overhead
empirical methods	statistical analysis of historical data	automation	cold start

· system complexity
· application complexity
· interaction between system and application

→

attention on empirical methods

Machine Learning in Performance Modeling

Machine Learning	
A technology to learn knowledge and experience from historical data.	A powerful approach for empirical modeling methods.

☐ D. N. Hieu, T. T. Minh, T. Van Quang, B. X. Giang, and T. Van Hoai, “A machine learning-based approach for predicting the execution time of cfd applications on cloud computing environment,” in International Conference on Future Data and Security Engineering, pp. 40–52, Springer, 2016

☐ P. Malakar, P. Balaprakash, V. Vishwanath, V. Morozov, and K. Kumaran, “Benchmarking machine learning methods for performance modeling of scientific applications,” in 2018 IEEE/ACM Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems (PMBS), pp. 33–44, IEEE, 2018.

Extrapolation Issue of Machine Learning

- Extrapolation: testset feature subspace outside trainset feature subspace.
- Issue high-accuracy interpolations but low-accuracy extrapolations —— machine learning can achieve high-accuracy predictions in trainset subspace, but prediction accuracy reduces drastically outside trainset subspace.

02 | Two-level Model

Problem Description

Features

$X : (x_1, x_2, x_3, \ldots, x_n, p)$

- $x_1, \ldots, x_n$: input parameters, Identically and Independent Distributed in X_{train} and X_{test}.
- p: number of processors, $p \in [a,b]$ in X_{train}, $p \in [c,d]$ in X_{test}, $b < c$.

Label

y

- y: execution time, y_{train} is known from historical logs, y_{test} is to be predicted.

Machine Learning Mechanism

Target model: $$f^* = \arg\min_f L(y_{test}, f(X_{test}))$$

Independent and Identically Distributed (IID) hypothesis

Approximate model: $$f^* \approx \arg\min_f L(y_{train}, f(X_{train}))$$

Motivation of Two-Level Model

Issues of one-level model	Overview of two-level model
- Simple algorithms: · own extrapolation ability in some way · cannot learn complex relations between input parameters and performance	- Interpolation level: learn accurate interpolation model and predict small-scale performance under input parameters in X_{test}
- Sophisticated algorithms: · learn accurate relations between input parameters and performance · overfitting on small-scale data	- Extrapolation level: construct model owning extrapolation ability and predict y_{test} with corresponding small-scale performance predictions

Workflow of Two-Level Model

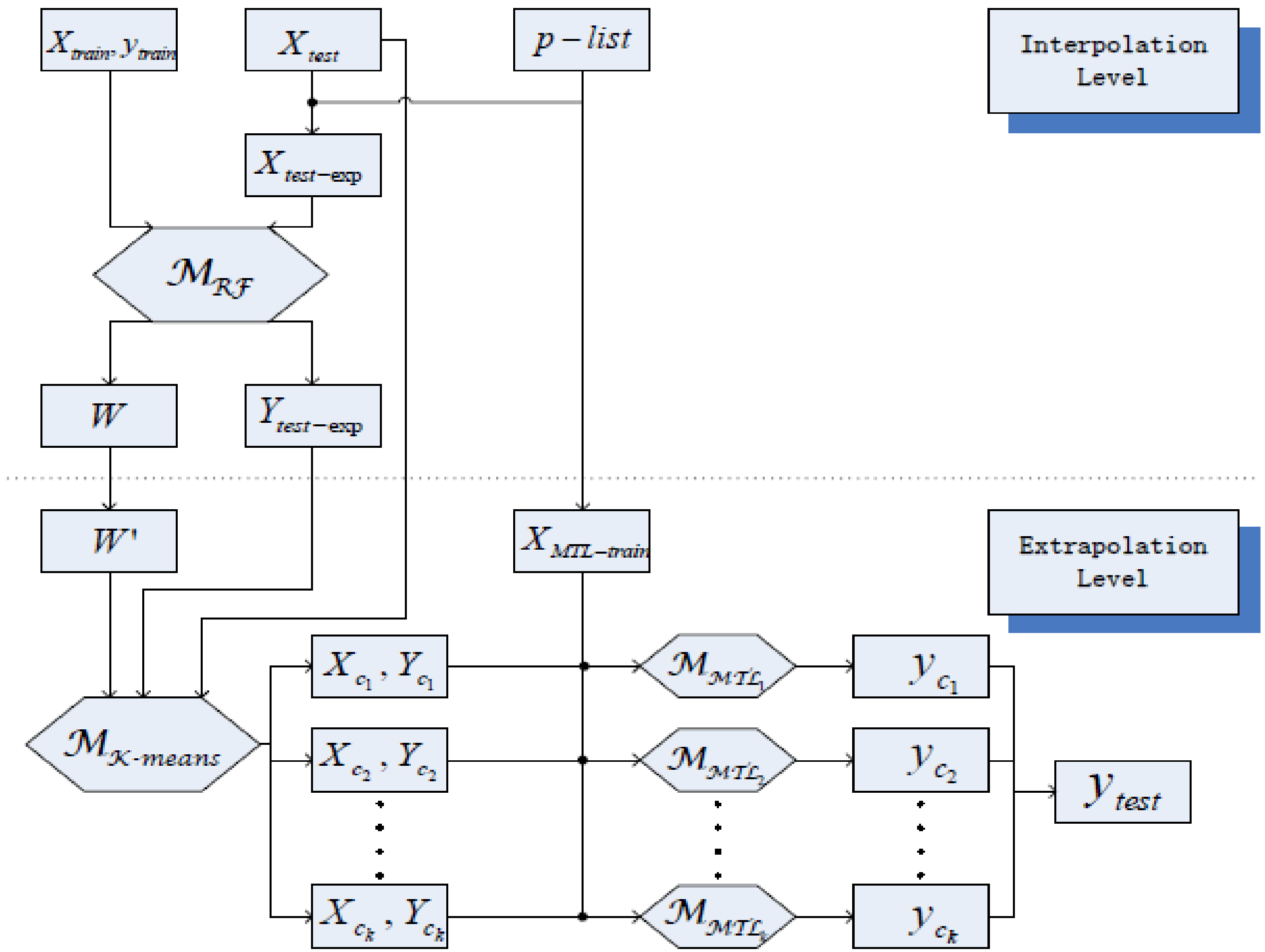
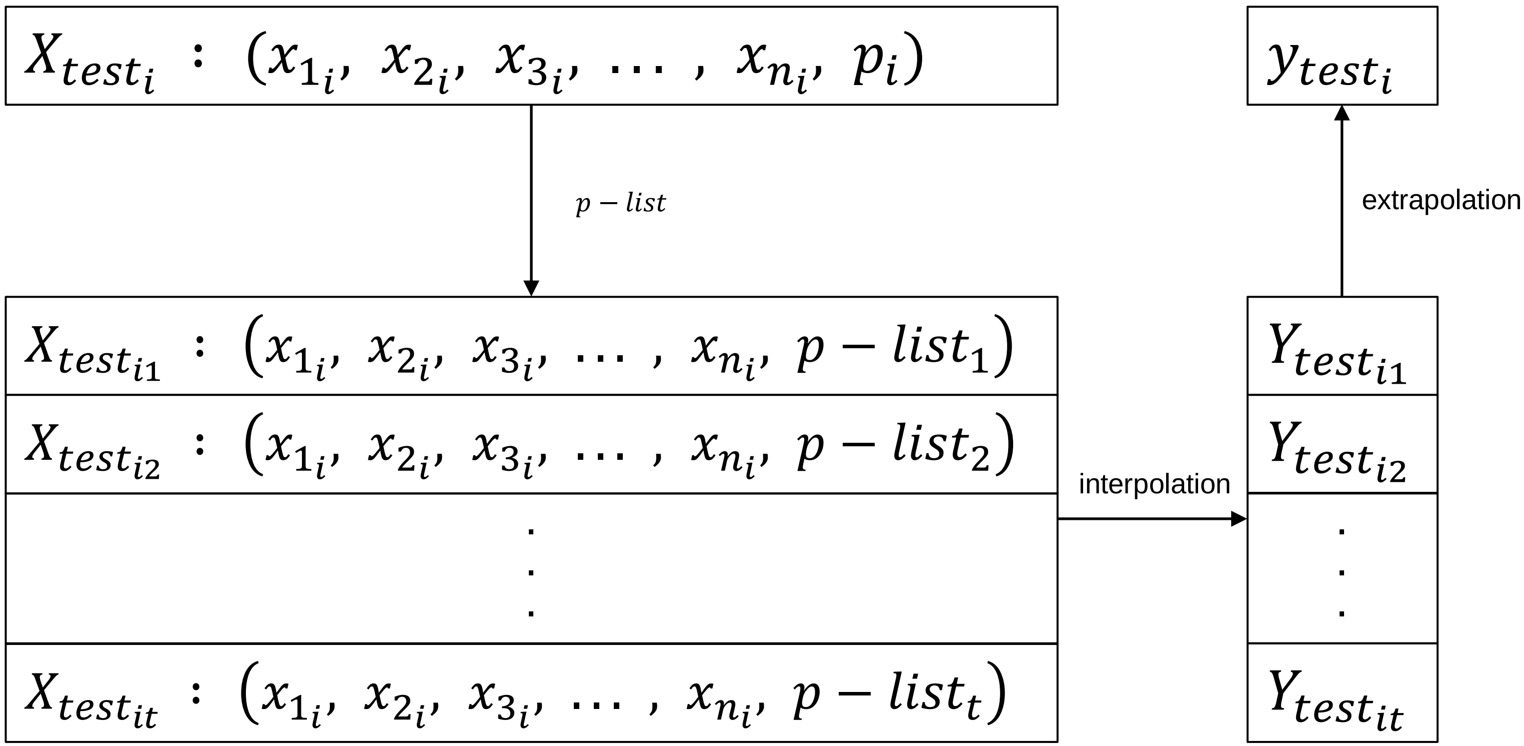


fig 3. workflow of two-level model

Prediction Specification



Analysis of Interpolation Level

- Task:
 small-scale performance predictions as extrapolation level training data
- Requirement:
· high accuracy
· random error distribution

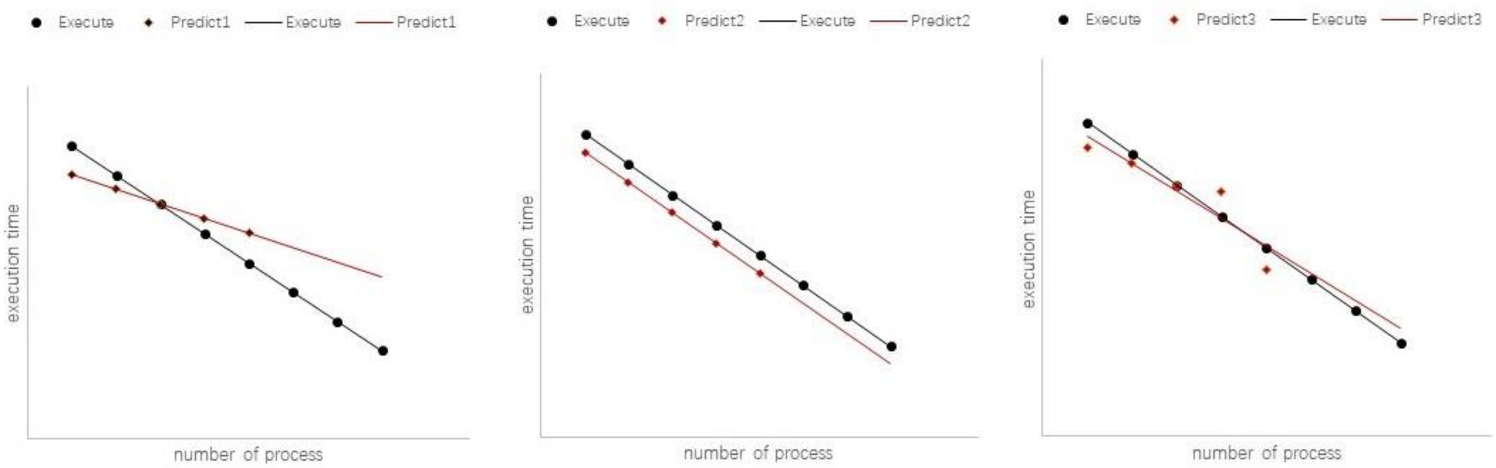


fig 4. influence of error distribution

Interpoaltion Level Model

- feature randomness
- sample randomness

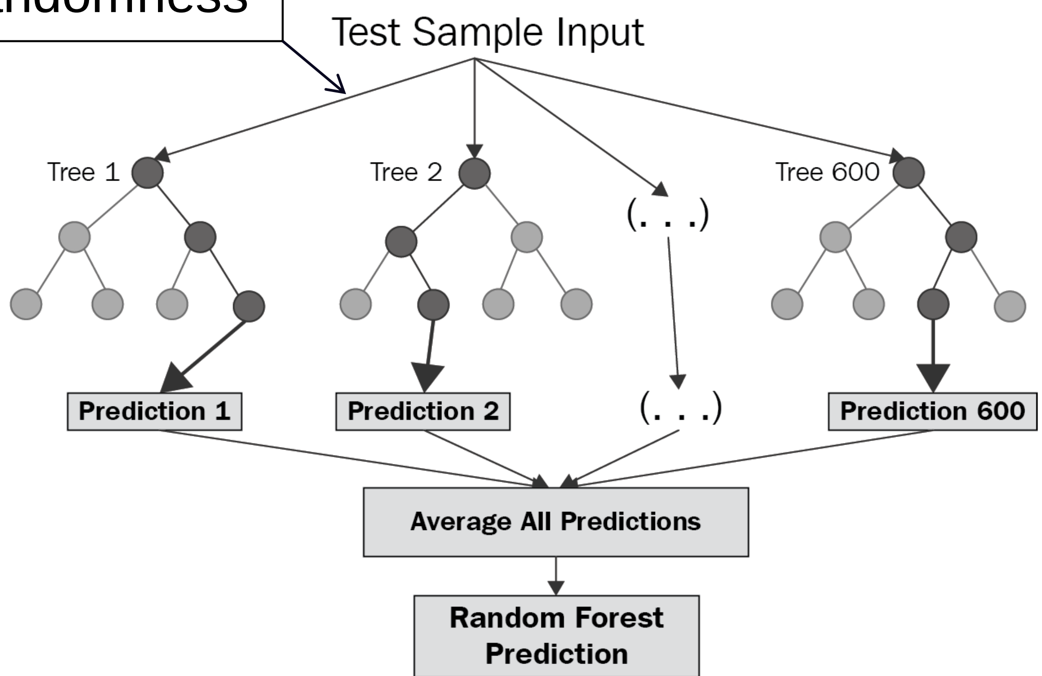


fig 5. random forest

- high accuracy
- random error distribution

Analysis of Extrapolation Level

- Task:
· predict large-scale performance with small-scale predictions

- Chanllenge:
· extrapolation (scalability) —— scalability may change with input parameters
· interpolation error —— only several data points for every input parameter combinations, easy to overfit causes error amplification

Extrapolation Level Model

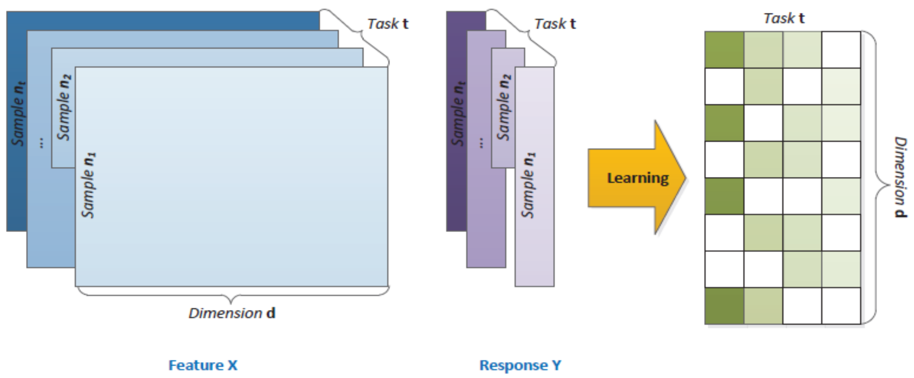
Scalability

Performance Modeling Normal Form(PMNF):

$$f(p) = \sum_{k=1}^{n} c_k \cdot p^{i_k} \cdot log_2{}^{j_k}(p)$$

Interpolation Error

Multi-Task Lasso:



03 | Experiment Results & Analysis

Experiment

Platform Node

Specification	configuration
CPU type	E5-2680 v4
CPU frequency	2.4GHz
#core	28
memory	128GB
network	100Gbps OPA

Monte Carlo Benchmark (MCB) Features

Name	Type	Values
nZonesX	integr	[100, 1000]
nZonesY	integer	[100, 1000]
xDim	float	[1.0, 10.0]
yDim	float	[1.0, 10.0]
xSource	float	[1.0, 10.0]
ysource	float	[1.0, 10.0]
numParticles	integer	$[1 \times 10^7, 2 \times 10^7]$
#process	integer	[16, 32, 48, ..., 512]

Kripke Features

Name	Type	Values
layout	enumeration	DGZ, DZG, GDZ, GZD, ZDG, ZGD
gset	integer	1, 2, 4, 8, 16, 32, 64, 128
dset	integer	8, 16, 32
pmethod	enumeration	sweep, bj
#process	integer	1, 2, 4, 8, 16, 32, 64, 128

Evaluation

Baseline

- Random Forest
- Multi-Layer Perceptron
- EPMNF

$$f(P) = \sum_{i=1}^{|P|} \sum_{k=1}^{n} c_i \cdot p_i^{j_{ik}} log_2^{l_{ik}}(p_i)$$

- Log Rgression

$$log(T) = \beta_1 log(x_1) + \beta_2 log(x_2) + ... \beta_n log(x_n) + \beta_p log(p) + error$$

Metrics

- MAE

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |\hat{y}_i - y_i|$$

- MAPE

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{\hat{y}_i - y_i}{y_i} \right|$$

- RMSE

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2}$$

Result: Comparison of Different Methods

App	Method	MAPE		MAE		RMSE	
		inter	extra	inter	extra	inter	extra
MCB	RF	0.1481	1.4097	14.62	42.12	21.55	48.20
	MLP	0.1729	0.5323	17.06	20.64	25.15	27.88
	EPMNF	0.2560	1.2191	21.21	33.95	27.73	41.57
	LR	0.1677	0.3640	18.99	9.87	31.28	12.42
	RFMTL	0.1481	**0.2577**	14.62	**8.37**	21.55	**11.45**
Kripke	RF	0.0610	0.8676	5.63	16.11	23.55	31.63
	MLP	0.3320	0.4758	27.24	12.71	52.91	26.98
	EPMNF	0.7324	3.4028	29.77	24.79	55.87	34.07
	LR	0.3088	0.6729	25.58	17.85	55.81	41.07
	RFMTL	0.0610	**0.2524**	5.63	**7.74**	23.55	**20.58**

· extrapolations are harder than interpolations
· the performance of the same method in diffrent applications varies greatly
· two-level model perform better than one-level model

Result: Comparison of Different Methods

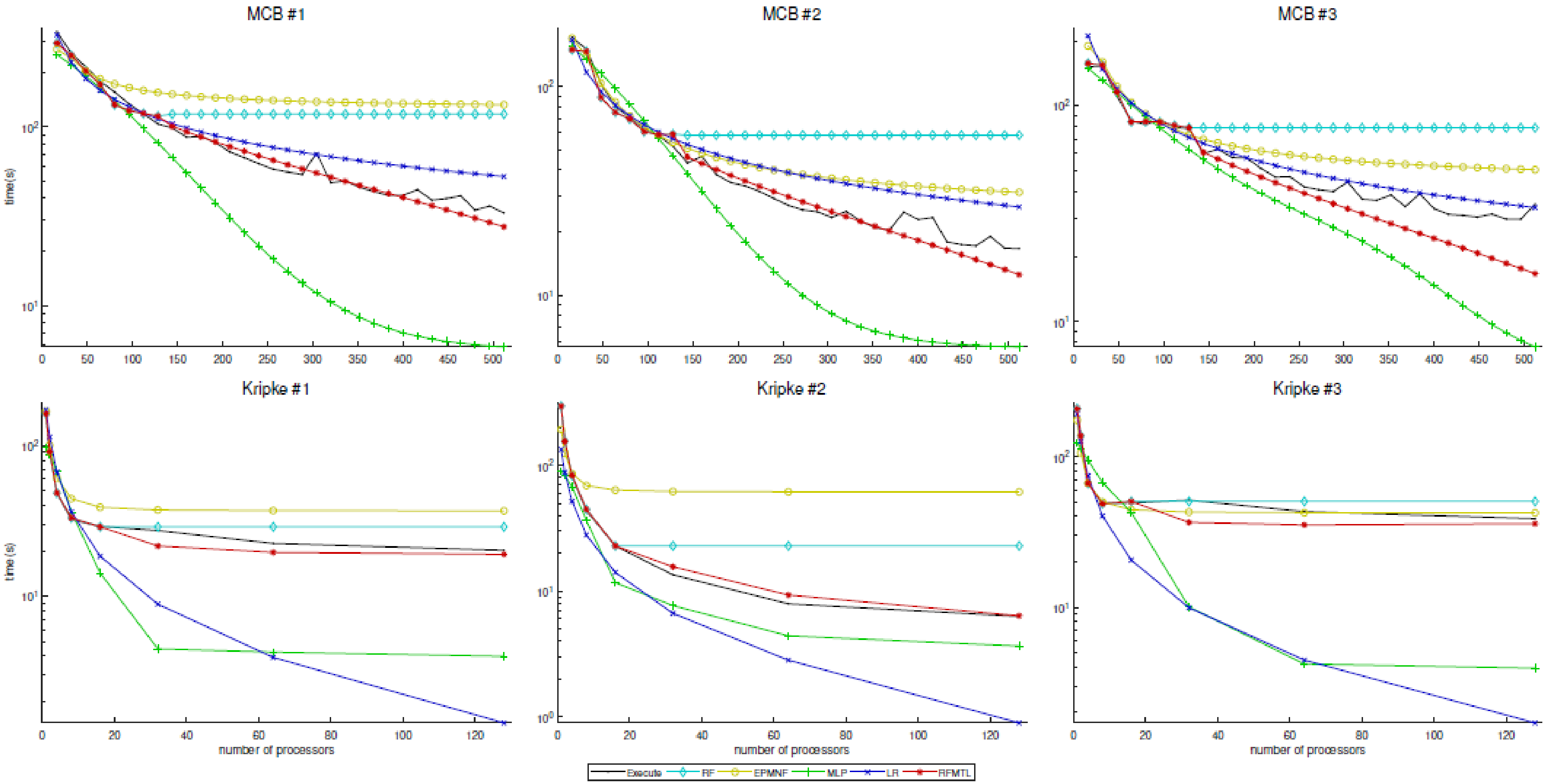


fig 6. case study of different methods

Result: Single-Task v.s. Multi-Task

App	Method	MAPE	MAE	RMSE
MCB	ST	0.6789	28.06	61.35
	MT	**0.2659**	**9.77**	**14.25**
Kripke	ST	0.6662	17.67	41.47
	MT	**0.2880**	**8.12**	**18.22**

Multi-Task Learning

- data amplification
- feature selection

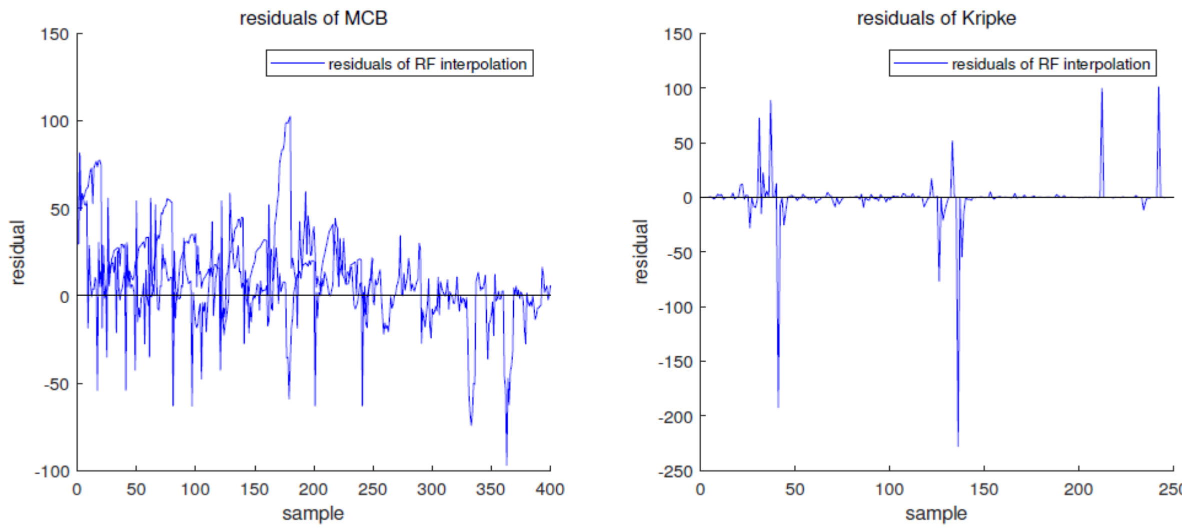


fig 7. residuals of random forest

Result: Single-Task v.s. Multi-Task

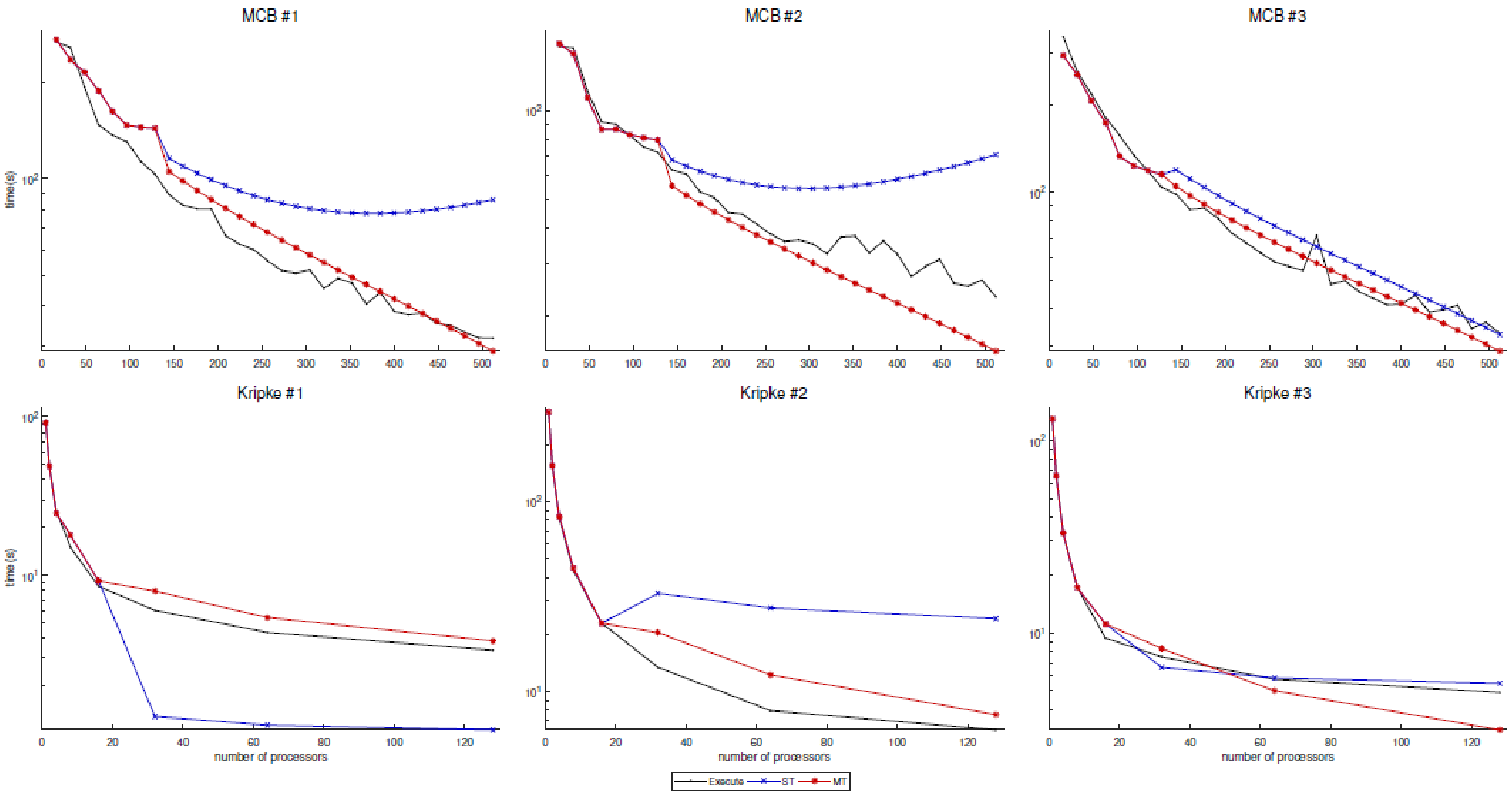


fig 8. case study of multi-task learning

Result: Clustering or Not

K-means Cluster

- distance:

$$Dist(X_i, X_j) = \sqrt{\sum_{k=1}^{n} W_k'(X_{ik} - X_{jk})^2}$$

- effect:

partition tasks into cluster by distance (relatedness) to learn high-related tasks jointly.

App	Method	MAPE	MAE	RMSE
MCB	NCL	0.2659	9.77	14.25
	CL	**0.2577**	**8.37**	**11.45**
Kripke	NCL	0.2880	8.12	**18.22**
	CL	**0.2524**	**7.74**	20.58

reasons for insignificance

· input sensitivity
· experimental feature values range
· clustering algorithm

Result: Clustering or Not

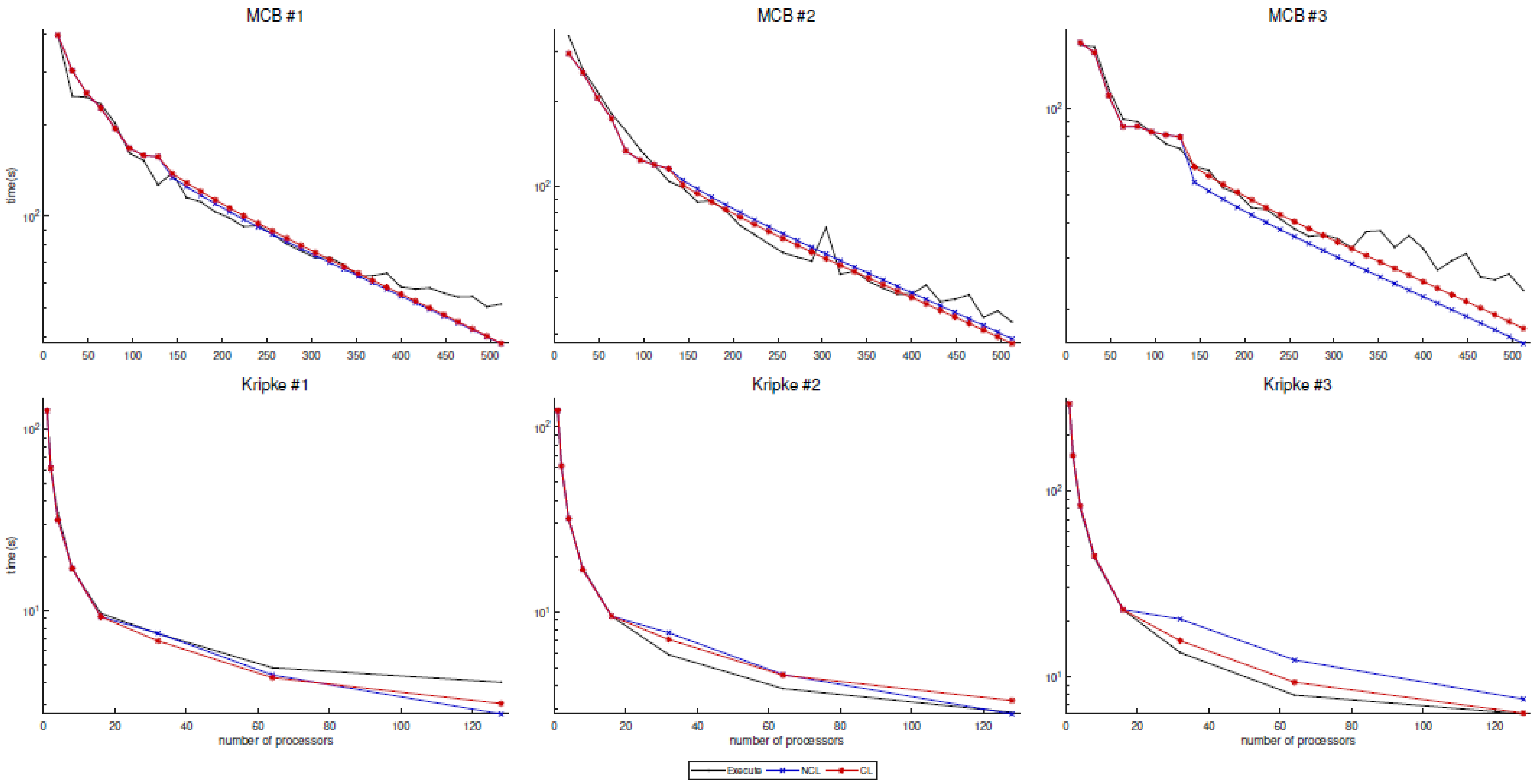


fig 9. case study of clustering

04 | Conclusions

Conclusions

- analyze the extrapolation problem and issues of one-level model

- propose a two-level model to predict large-scale performance with only small-scale historical data

- conduct experiment to validate the effectiveness of two-level model

Future Work

- improve two-level model by choosing more fitting clustering and multi-task learing algorithms

- improve scalability models with considering system information to model cross-platform performance

- research whether two-level model works for extrapoaltion problem caused by input parameter

Thanks.

Welcome further communication.

Wenju Zhou

zhouwenj@mail.ustc.edu.cn