Stevens HUDSONLab Seminar

10/09/2024

Presentation Order

- John Anticev (advisor: Prof. Zhuo Feng)
- Soumen Sikder Shuvo (advisor: Prof. Zhuo Feng)
- Hamed Sajadinia (advisor: Prof. Zhuo Feng)
- Wuxinlin Cheng (advisor: Prof. Zhuo Feng)

- Hanfei Yu (advisor: Prof. Hao Wang)
- Rui (Ricky) Wei (advisor: Prof. Hao Wang)
- Qingyang Yu (advisor: Prof. Hao Wang)
- Zixun Xiong (advisor: Prof. Hao Wang)

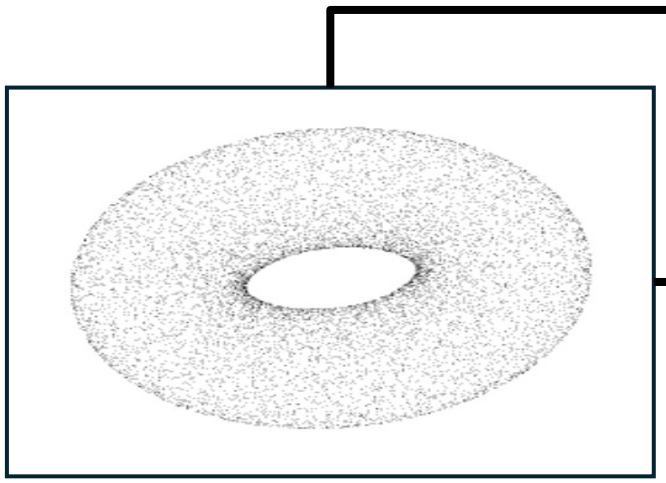
SGM-PINN: Sampling Graphical Models for Faster Training of Physics-Informed Neural Networks

John Anticev

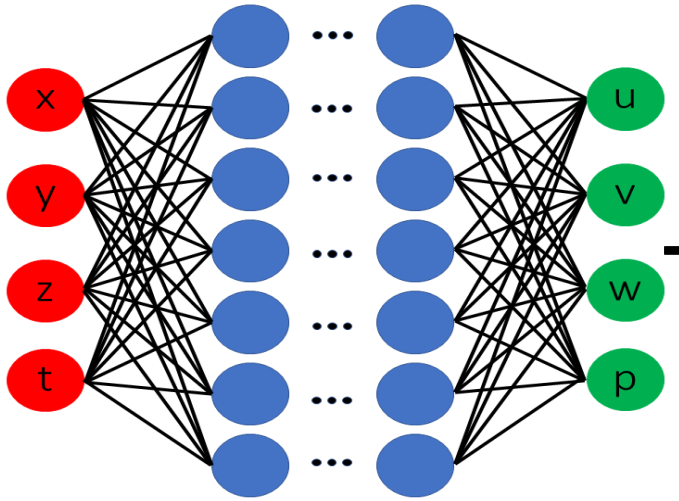
janticev@stevens.edu

Physically Informed Neural Networks (PINNs)

- PINNs [1] use a deep neural network as a universal function approximator
- Spatio-temporal data (x, y, z, t) is used to predict outputs (e.g. u, v, w, p in Navier-Stokes equations)



Training Data **x** (point cloud)
Boundary/Initial Conditions



Loss Function
(PDE+BC+IC Loss)

Derivatives

$$\frac{\partial u}{\partial x_1}, \ldots, \frac{\partial u}{\partial x_d}; \frac{\partial^n u}{\partial x_1}, \ldots, \frac{\partial^n u}{\partial x_d}$$

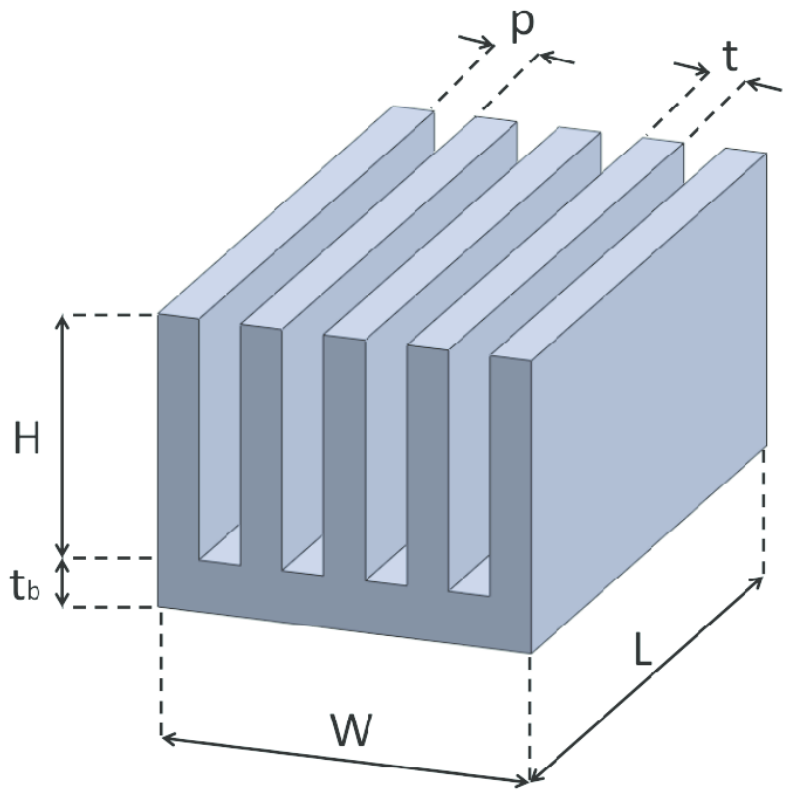
Residuals

[1] M. Raissi, P. Perdikaris, G.E. Karniadakis, "Physics-informed neural networks…", J. Comput. Phys. 378 (2019), 686-707

Parameterized PINN Models

With parametrized domain geometry [1][2]:

- PINNs can solve a family of designs at once
- A single training phase → fast inference of many designs
- Fast design space exploration



https://www.researchgate.net/figure/46_tbl1_305819333

[1] Luning Sun, Han Gao, Shaowu Pan, Jian-Xun Wang, *Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data*, Computer Methods in Applied Mechanics and Engineering 361 (2020)
[2] Christopher J. Arthurs, Andrew P. King, *Active training of physics-informed neural networks to aggregate and interpolate parametric solutions to the Navier-Stokes equations*, J. Comput. Phys. 438 (2021)

Importance Sampling for Training PINNs

Training for a loss function $\mathcal{L}(\boldsymbol{\theta})$ with model parameters $\boldsymbol{\theta}$, the goal is:

$$\theta^* = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(\boldsymbol{\theta}; \boldsymbol{x}_i). \quad (1)$$

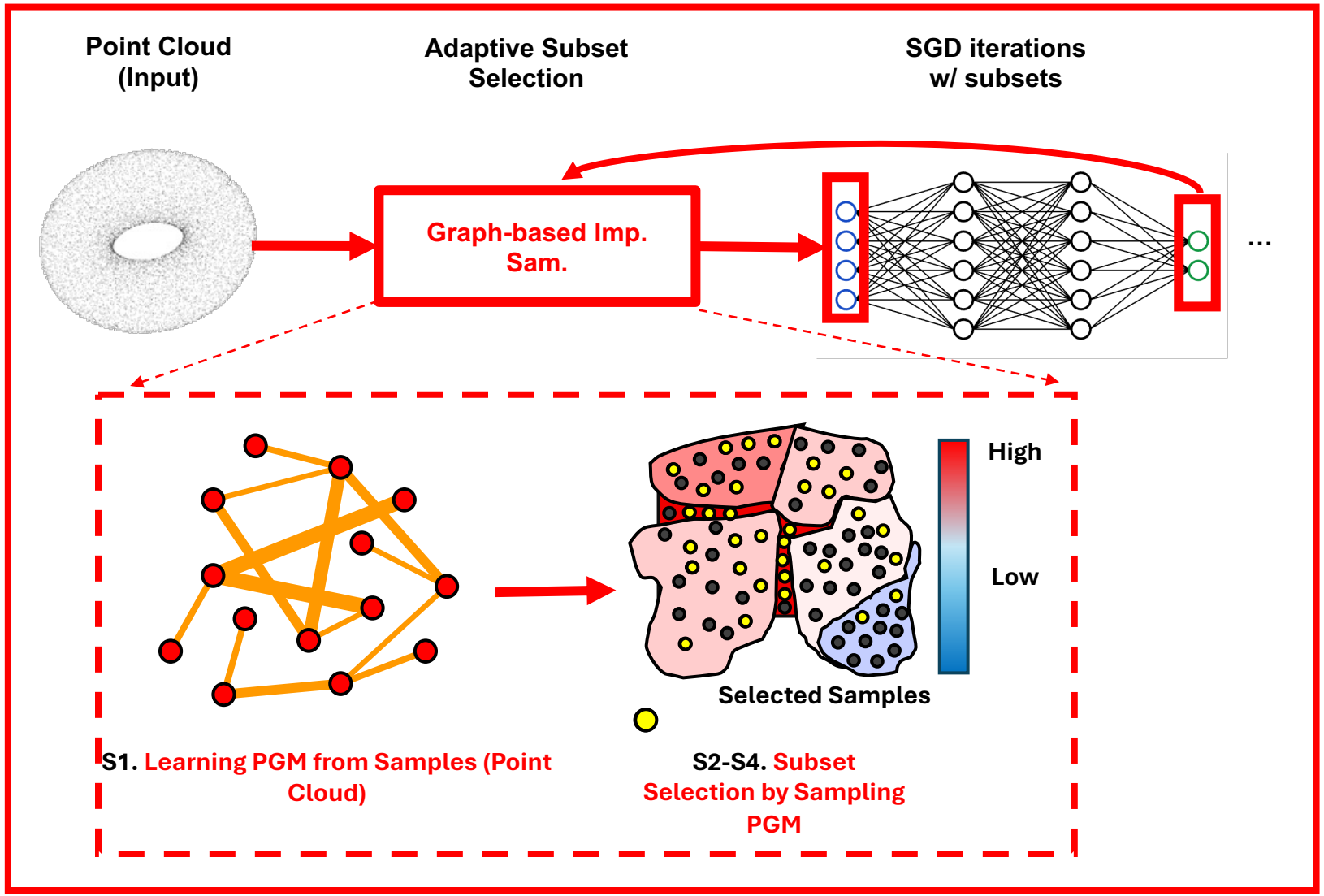
(1) is solved with methods like stochastic gradient descent (SGD)

$$\theta^{(t+1)} = \theta^{(t)} - \alpha^t \nabla_{\boldsymbol{\theta}} \mathcal{L}\left(\boldsymbol{\theta}^{(t)}\right). \quad (2)$$

[1][2] establish that sampling from a distribution $p^{(t)} \propto \mathcal{L}(\theta^t)$ can better estimate of $\nabla_\theta \mathcal{L}(\theta^t)$ during training, and accelerate convergence.

[1] Katharopoulos, A., & Fleuret, F. “Not All Samples Are Created Equal: Deep Learning with Importance Sampling.” ICML (2018)
[2] Nabian, Mohammad Amin et al. “Efficient training of physics-informed neural networks via importance sampling.” Computer-Aided Civil and Infrastructure Engineering 36 (2021): 962 - 977.

SGM-PINN Overview

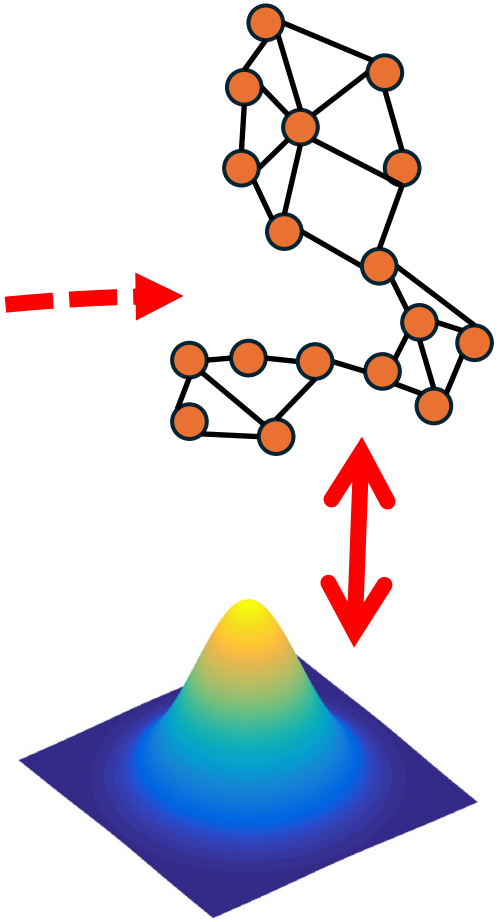


S1: Graph Construction via Probabilistic Graphical Model (PGM)

- The PGM is defined as a precision matrix that maximizes the probability density function (pdf) of a multivariate gaussian

$$f(\boldsymbol{x}) = \frac{1}{(2\pi)^{p/2}\det(\boldsymbol{\Sigma})^{1/2}} \exp\left\{-\frac{1}{2}\boldsymbol{x}^{\top}\boldsymbol{\Sigma}^{-1}\boldsymbol{x}\right\}$$

Precision Matrix

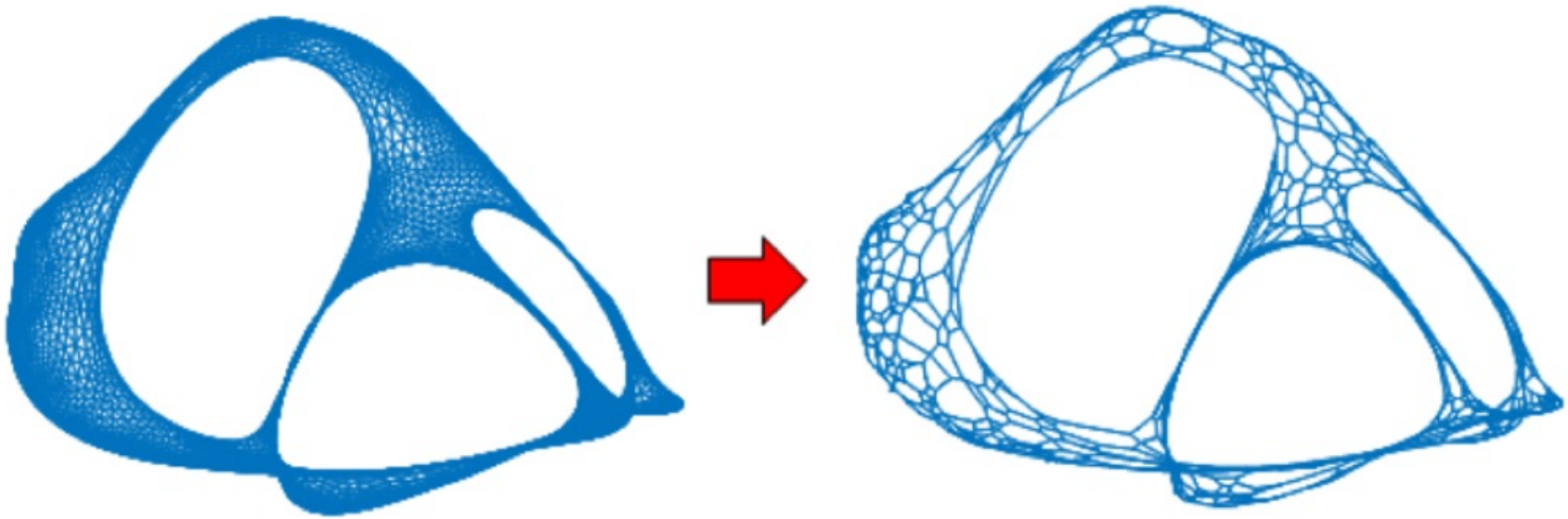


- The precision matrix denotes an undirected graph where each edge corresponds to the conditional dependence between two samples

[1] J. Friedman, et al. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* (2008)

S2: Graph Clustering via Spectral Coarsening

- Structure-preserving coarsening: Fewer nodes, fewer edges, similar spectral properties
- Highly efficient, near-linear time algorithm



[1] Ali Aghdaei and Zhuo Feng. 2022. “HyperEF: Spectral Hypergraph Coarsening by Effective-Resistance Clustering.” ICCAD (2022)

S3: Stability Scoring

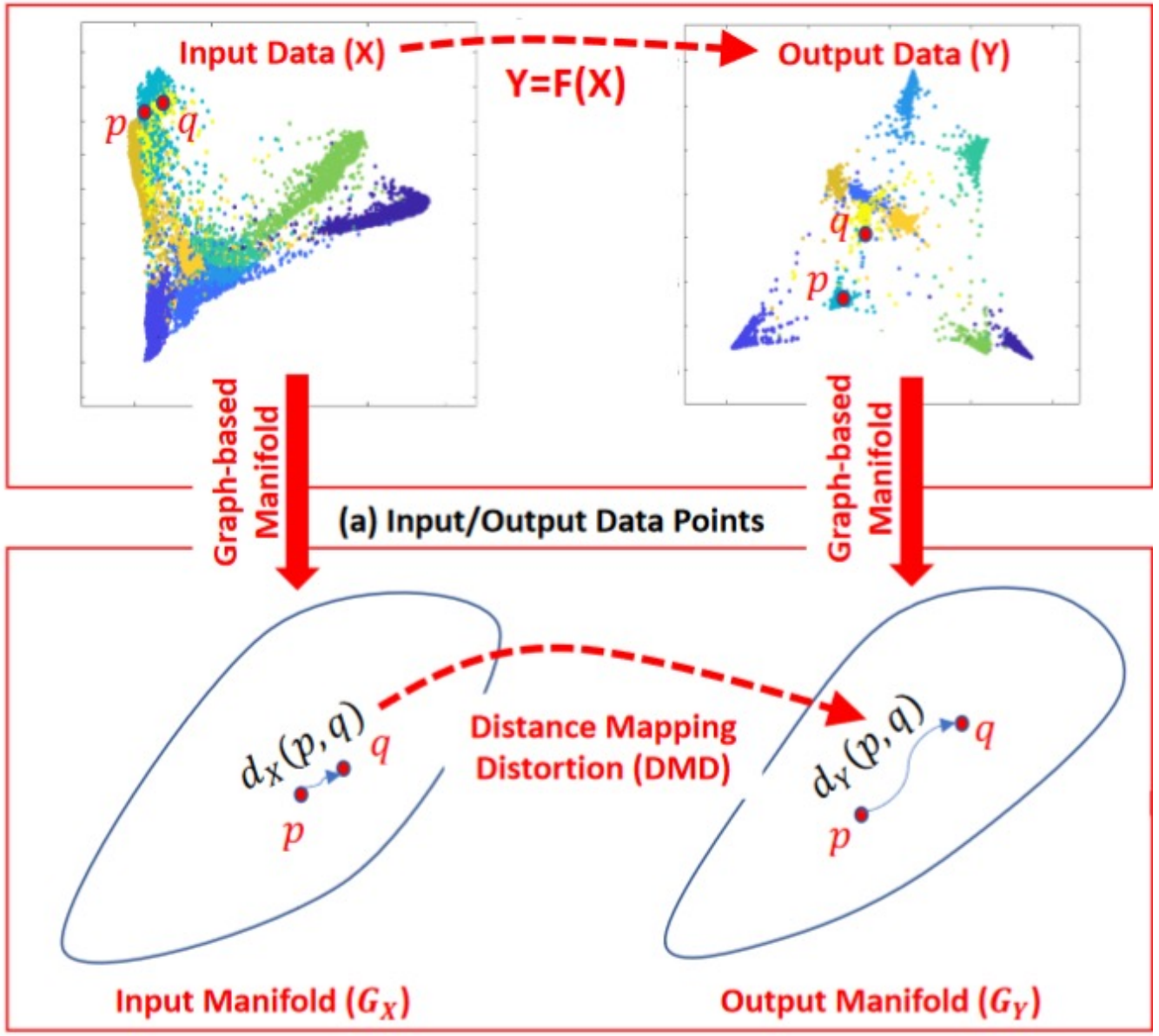
[1] introduces the Distance Mapping Distortion (DMD) metric: $\gamma_F(p,q)$

$$\gamma^F(p,q) \triangleq \frac{d_Y(p,q)}{d_X(p,q)} \qquad (1)$$

$$\gamma^F(p,q) \leq K^* \qquad (2)$$

K^* is the best Lipschitz constant

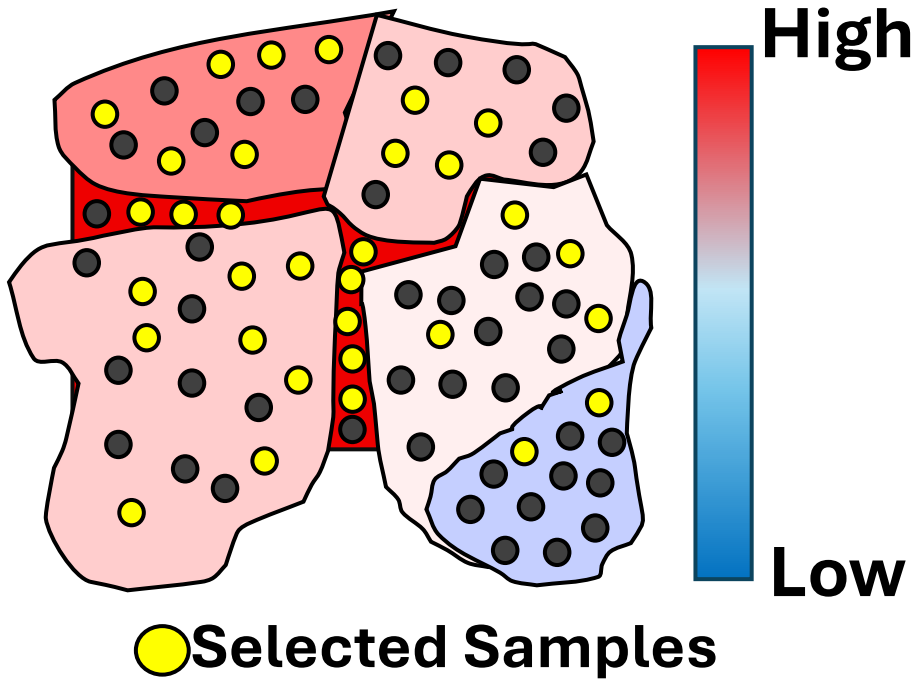
DMD (local) ↑ $\|\nabla_{x_i}\mathcal{L}(\theta)\|_2$ ↑ Stability ↓



[1] Wuxinlin Cheng, Chenhui Deng, Zhiqiang Zhao, Yaohui Cai, Zhiru Zhang, and Zhuo Feng. "SPADE: A Spectral Method for Black-Box Adversarial Robustness Evaluation." ICML (2021)

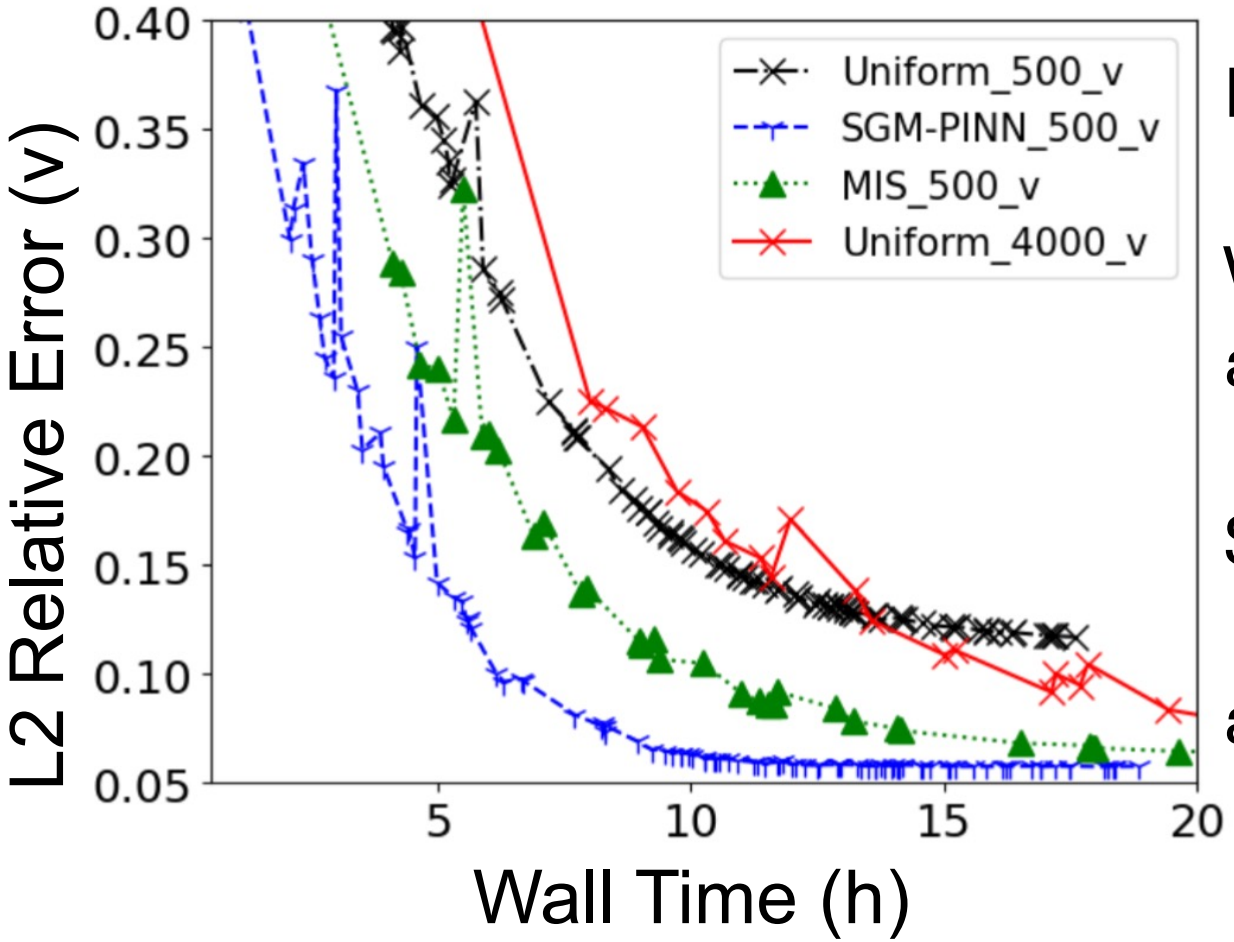
S4: Score Estimation and Importance Sampling

- Small fraction of sample scores is updated every few epochs
- Cluster importance is ranked based on sample scores
- Next ‘mini-epoch’ is taken from each cluster



S4. Subset Selection by Sampling PGM

Results: Non-Parameterized LDC



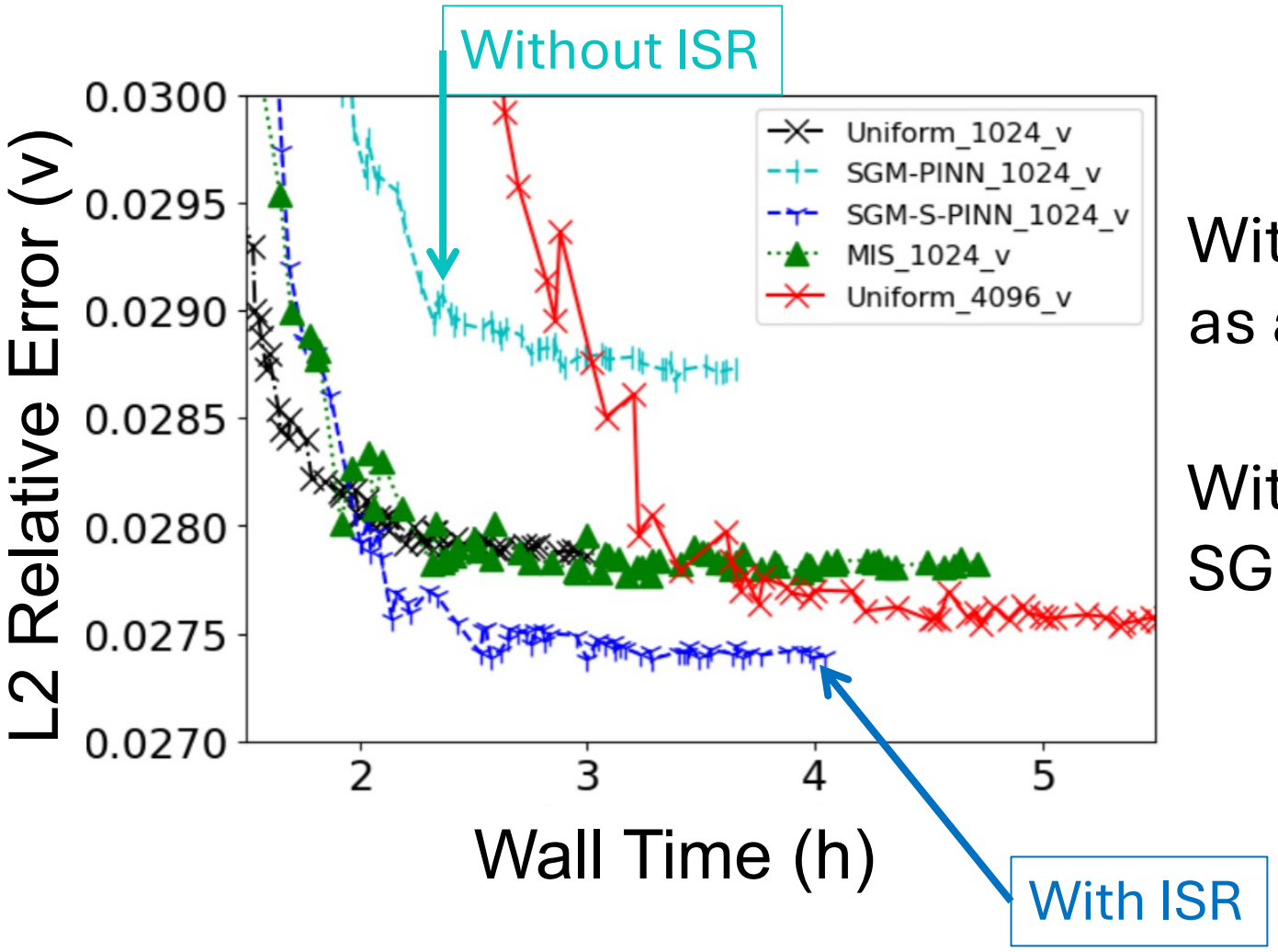
Batch Size can be reduced to 1/8th

With the Baseline's best result/time as a reference:

SGM-S is **2.95x** faster than the baseline

and **1.74x** faster than MIS

Results:Parameterized AR



With the Baseline's best result as a reference, SGM-S is 2x faster

With MIS's best result as a reference, SGM-S is 1.53x faster

Conclusion

- We introduce a graph-based importance sampling framework for accelerating the training of PINNs
- SGM-PINN allows the importance score of multiple samples to be estimated via selection of highly-correlated clusters
- The inclusion of a stability score (local Lipschitz Estimation) can improve parameterized training
- 2X-3X runtime improvement training select PINNs related to CFD problems

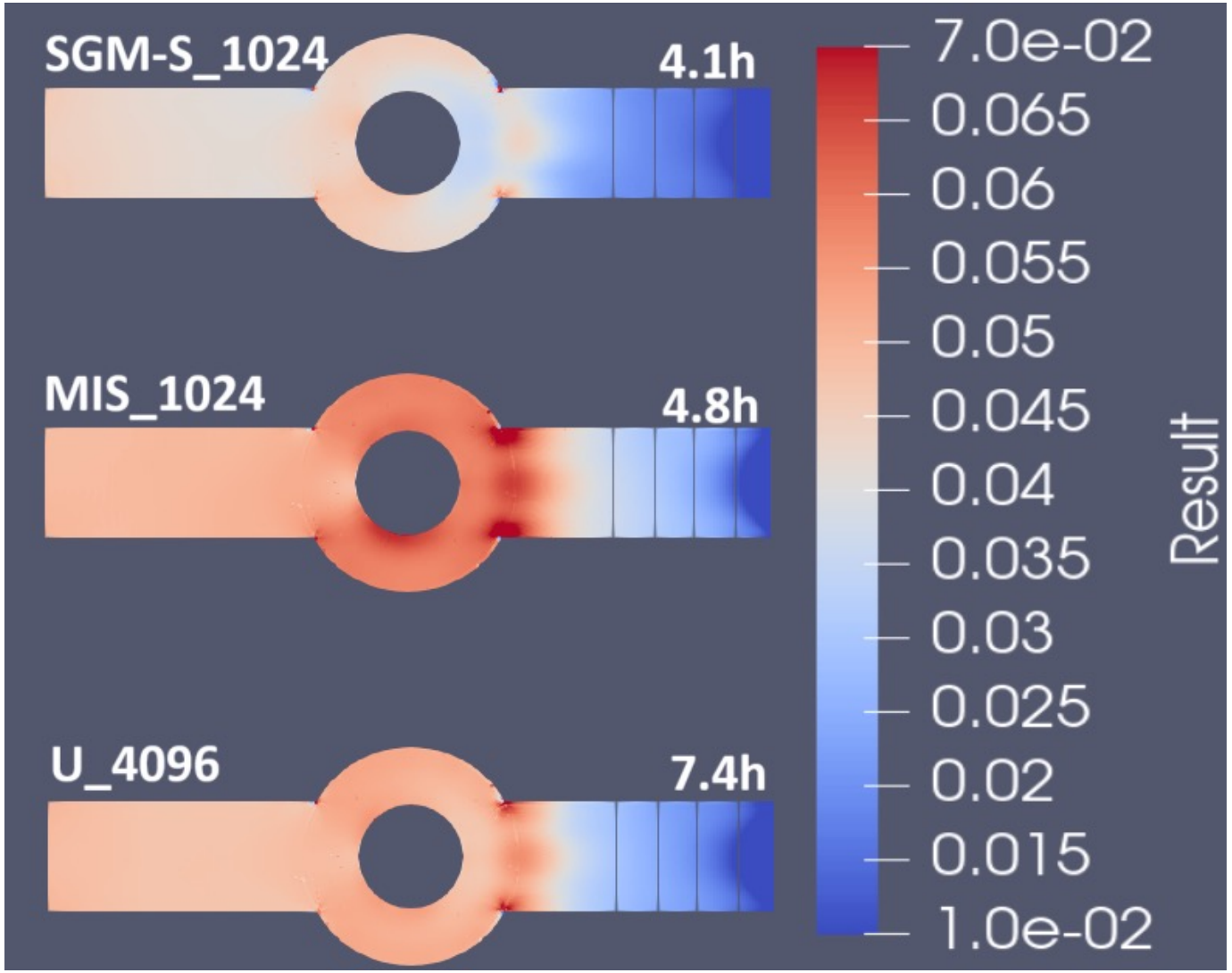
Backups

Ends here

Backup- Viz Parameterized AR

Visualization of error in predicting pressure, *p*

Solution is averaged from r_i =1.0, 0.88, and 0.75



Backup: Future Work

- Further reduce overhead
- Include importance when considering the Boundary Conditions
- More complex examples from more domains
- Automate hyperparameter selection (esp. coarsening level)

A Spectral Framework for Assessing the Geodesic Distance Between Graphs

Soumen Sikder Shuvo

sshuvo@stevens.edu

Advisor: Prof. Dr. Zhuo Feng

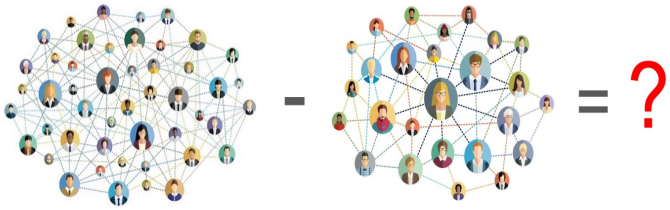
Introduction

- Graph Neural Networks (GNNs) have become crucial for many tasks involving graph-based data.
- A robust metric for GNN generalization and stability analysis is still not common.
- A distance metric between graphs is very important for these tasks.
- Graph Geodesic Distance (GGD): A Spectral distance metric between graphs [1].

Graph Distance Metrics

- Graph: Non-Euclidean Data
- Existing Distance Metrics:
  - Graph Edit Distance
    - Simple.
    - Not very accurate.
  - Kernel Based Distance: Wasserstein Weisfeiler-Leman (WWL), Tree Movers Distance (TMD) [2], etc.
    - State of the Art
    - Requires proper node features

10 - 7 = 3



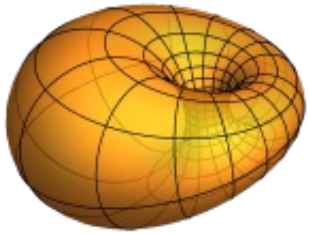
Methodology

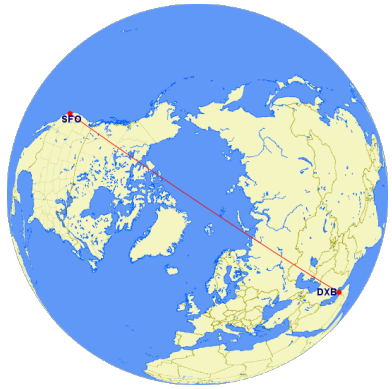
Phases:

- Graph matching task to get the approximate matching between nodes to secure the infimum in the manifold.
- Generating Symmetric Positive Definite (SPD) matrices from matched graph structures and Calculate GGD using Riemannian Metric.
- For Graphs of different sizes, we use coarsening method based on effective resistance and node features.

Geodesic in Riemannian Manifold

- Riemannian Manifold: A smooth geometric space where geometric notions such (distance, angles, curvature) are defined.
- Geodesic is the infimum (shortest distance) between two points on a manifold.
- Riemannian Metric:
  - Affine Invariant Riemannian Metric (AIRM)
  - Log-Euclidean Riemannian Metric (LERM)





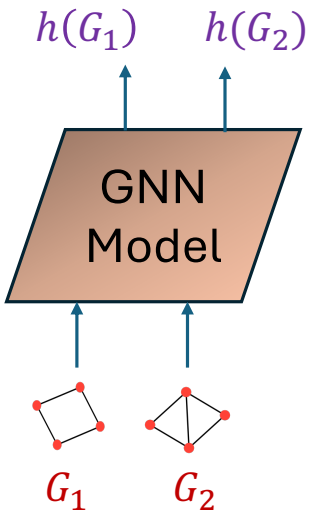
GGD as a Distance Metric

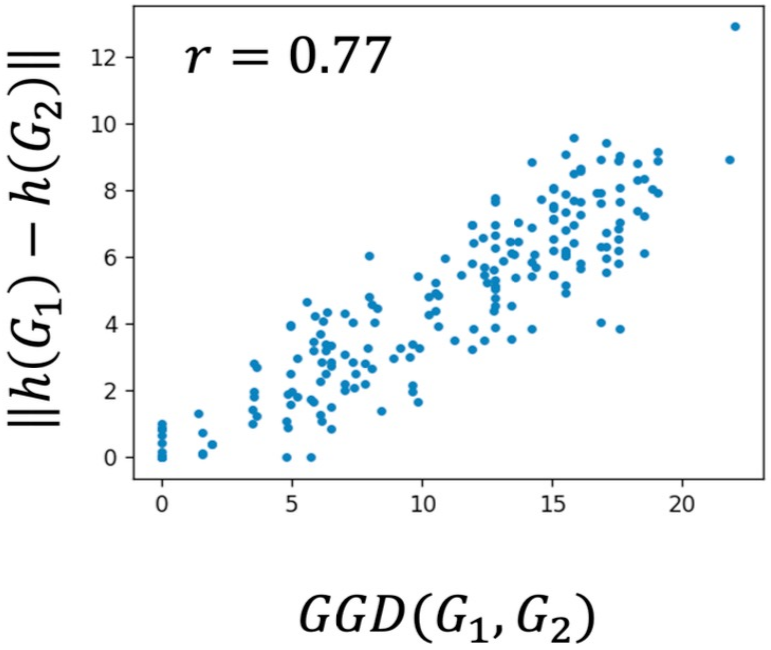
1. $GGD\ (G_1, G_1) = 0$
2. $GGD\ (G_1, G_2) \geq 0$
3. $GGD\ (G_1, G_2) = GGD\ (G_2, G_1)$
4. $GGD(G_1, G_2) + GGD\ (G_2, G_3) \geq GGD\ (G_1, G_3)$

Results: Graph Classification

Method	Accuracy in percentage			
	MUTAG	PC-3H	SW-620H	BZR
GGD	86±7.5	**78.34**	**77.6±3.5**	**83.23**
TMD, L = 3	77±5.2	71.24	70.2±2.3	73.43
TMD, L = 4	78.2±6	71.37	70.8±2.3	73.96
TMD, L = 5		71.89	71.2±1.88	75.13
Graph Convolutional Network	77±3.8	70.56	69.4	72.56
Graph Isomorphism Networs	82.6±4.6	75.34	73.4	77.09
Wasserstein Weisfeiler-Leman	72.4±2.6	65.46	68.34	73.59
Weisfeiler-Lehman Subtree	76±6.3	68.43	70.56	N/A
Fused Gromov-Wasserstein	**88.33±5.6**	61.77	59.3	53.66

Results: Stability Analysis





References

[1] Soumen Sikder Shuvo, Ali Aghdaei, and Zhuo Feng. *"Geodesic Distance Between Graphs: A Spectral Metric for Assessing the Stability of Graph Neural Networks."* arXiv preprint arXiv:2406.10500 (2024).

[2] Ching-Yao Chuang and Stefanie Jegelka. *"Tree mover's distance: Bridging graph metrics and stability of graph neural networks."* Advances in Neural Information Processing Systems (2022).

Thank You

Questions?

Hypergraph related problems

HAMED SAJADINIA / PROFESSOR FENG LAB

HSAJADIN@STEVENS.EDU

Background

Bachelor of Science in Electronic Engineering

Shahid Chamran University, Ahvaz, Iran

Research Areas: *Analog and Digital Circuit Design, Programming*

Master of Science in Telecommunication Engineering

Iran University of Science and Technology, Tehran, Iran

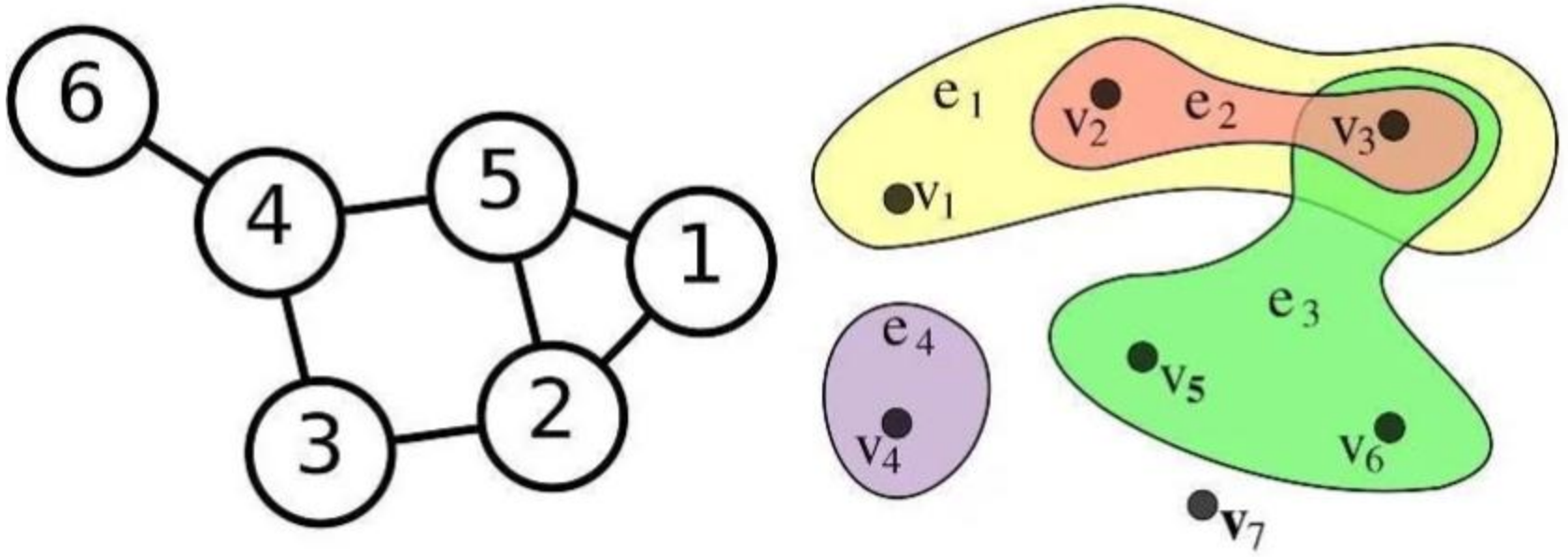
Research Areas: *High-frequency circuit Design, MMIC, RF/Microwave Circuits and Systems*

Research Experience:

Electronic Engineer/Researcher, ***IUST Research Lab***

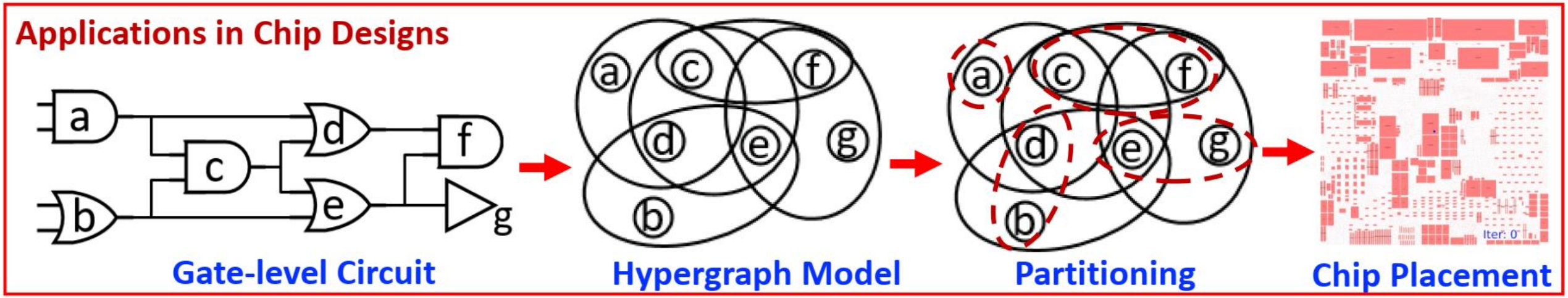
Analog Integrated Circuits Design, Programming Cryptography, RF/Microwave Circuits

Graph vs Hypergraph

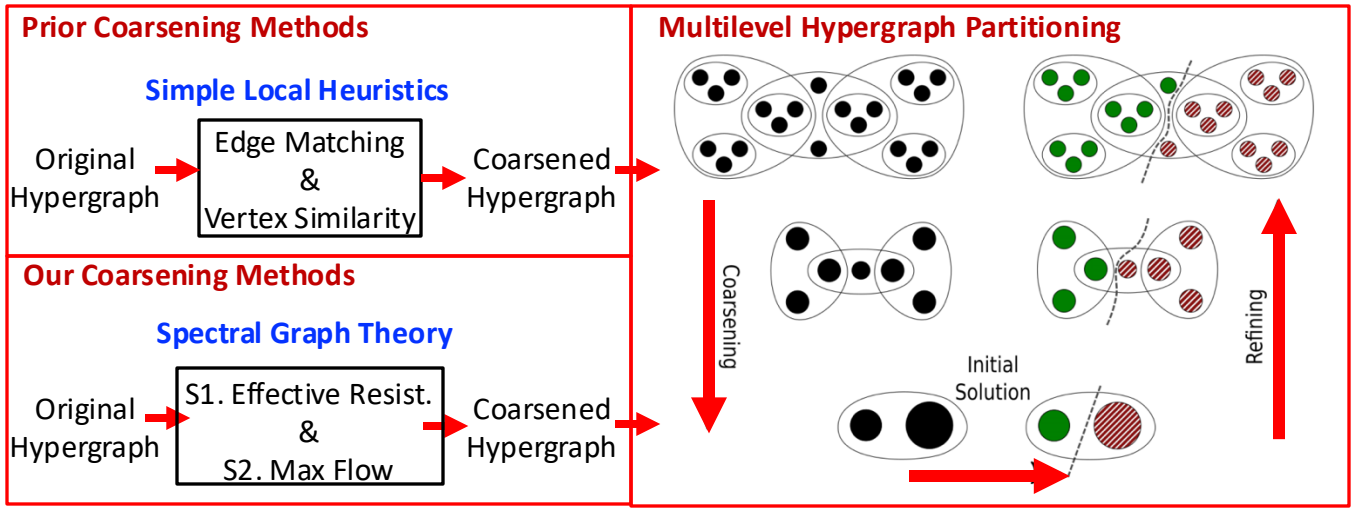


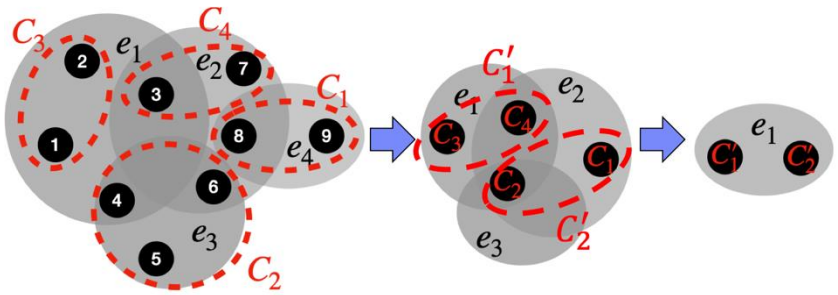
Hypergraph and it’s application

What is the Goal?



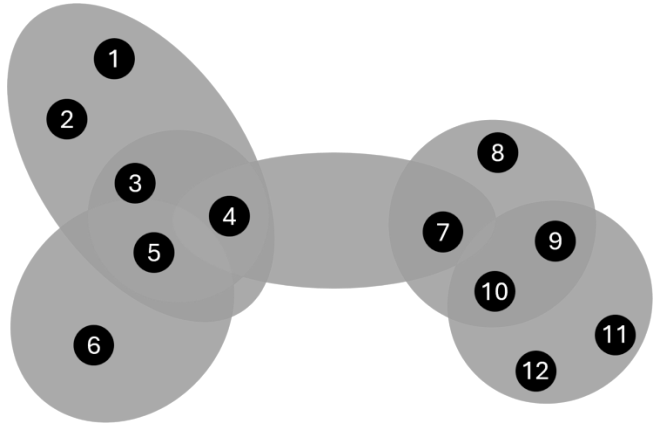
Hypergraph Partitioning

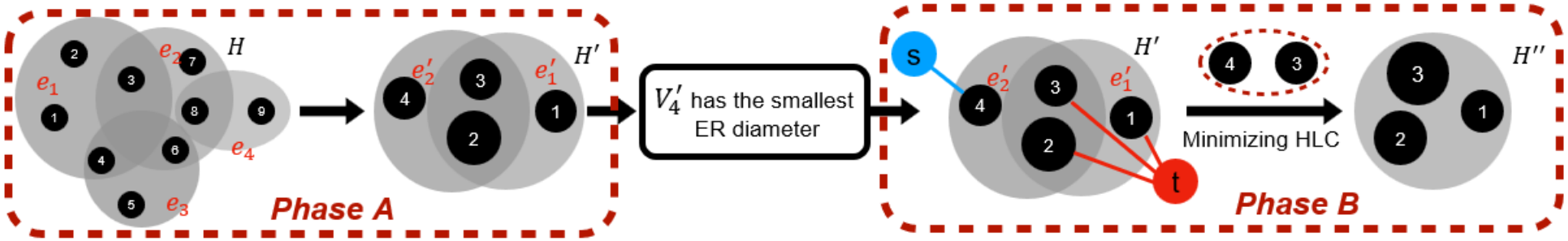




Spectral vs Local Hypergraph Partitioning

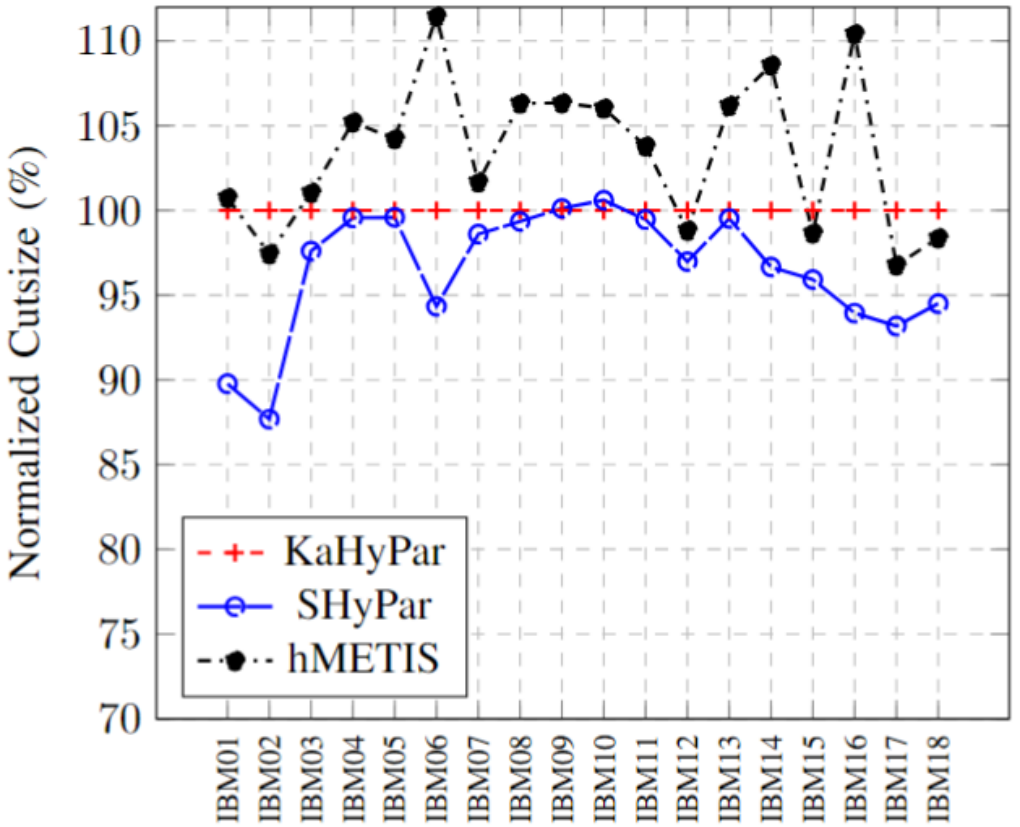
- Our contribution:
  - Resistance-base Clustering for coarsening
    - Previously using heuristic (edge size)
  - Flow-base clustering as community detection
    - Previously using Louvain algorithm





Hypergraph Partitioning Results

Benchmark	Statistics		$\epsilon = 2\%$				
	$\|V\|$	$\|E\|$	SpecPart	hMETIS	KaHyPar	MedPart	SHyPar
IBM01	12,752	14,111	202	213	202	202	201
IBM02	19,601	19,584	336	339	328	352	327
IBM03	23,136	27,401	959	972	958	955	952
IBM04	27,507	31,970	593	617	579	583	579
IBM05	29,347	28,446	1720	1744	1712	1748	1707
IBM06	32,498	34,826	963	1037	963	1000	969
IBM07	45,926	48,117	935	975	894	913	882
IBM08	51,309	50,513	1146	1146	1157	1158	1140
IBM09	53,395	60,902	620	637	620	625	620
IBM10	69,429	75,196	1318	1313	1318	1327	1254
IBM11	70,558	81,454	1062	1114	1062	1069	1051
IBM12	71,076	77,240	1920	1982	2163	1955	1986
IBM13	84,199	99,666	848	871	848	850	831
IBM14	147,605	152,772	1859	1967	1849	1876	1842
IBM15	161,570	186,608	2741	2886	2737	2896	2728
IBM16	183,484	190,048	1915	2095	1952	1972	1887
IBM17	185,495	189,581	2354	2520	2284	2336	2285
IBM18	210,613	201,920	1535	1587	1915	1955	1521



Thank You…!

Serverless Computing for AI Systems

Hanfei Yu

IntelliSys Lab

STEVENS
INSTITUTE OF TECHNOLOGY
1870

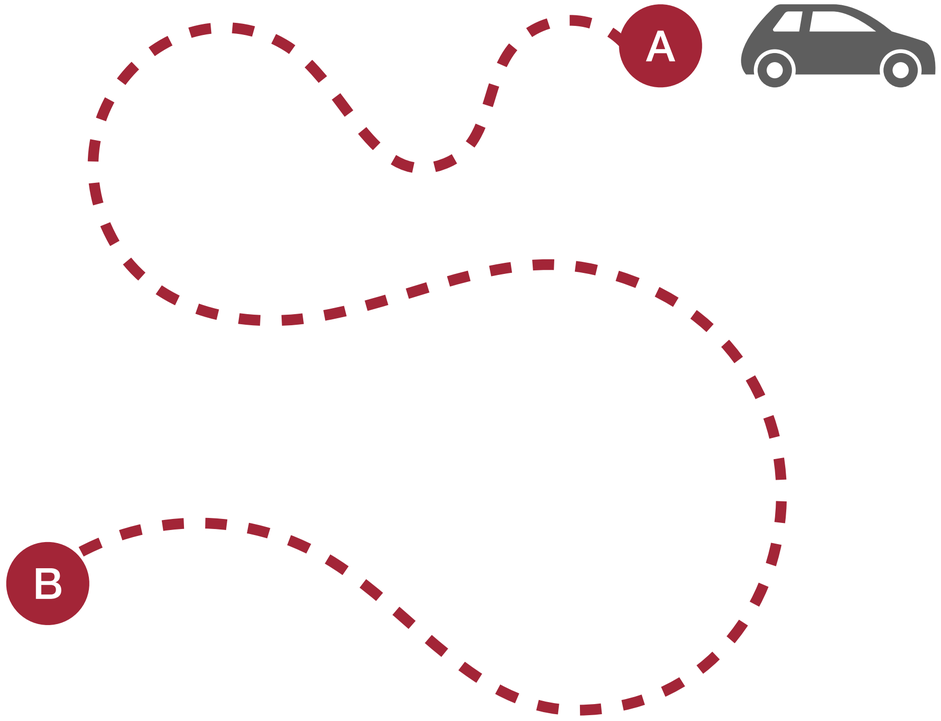
2019 July	Graduated from SJTU
2021 March	Graduated from UWT
2021 June	Joined IntelliSys Lab @ LSU
2024 Sep	Moved to IntelliSys Lab @ SIT

Research Interests	Large-Scale AI/ML Systems
	Serverless Computing
	Distributed DRL Systems
	LLM Serving Systems

Research Works

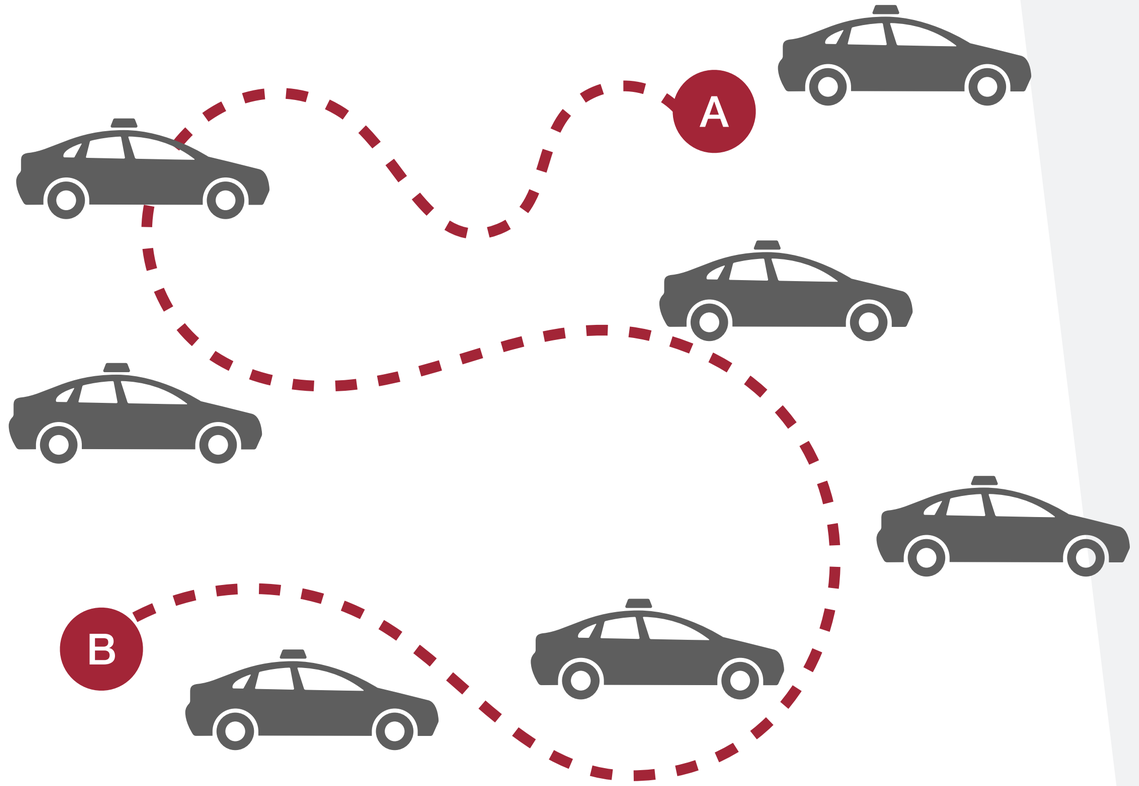
- Large-scale **distributed DRL systems** on serverless computing
  - *VLDB'25, SC'24 (Best Student Paper Finalist), AAAI'24*
- Accelerated **serverless inference** of large-scale Deep Learning systems
  - *SoCC'24*
- Resource management and function scheduling for **serverless computing**
  - *ASPLOS'24, TPDS'24, HPDC'23, WWW'22, ACSOS'21*
- Memory-efficient **LLM serving** of Mixture-of-Experts (MoE) architectures
  - *In-progress*

Cloud / HPC



Car rental

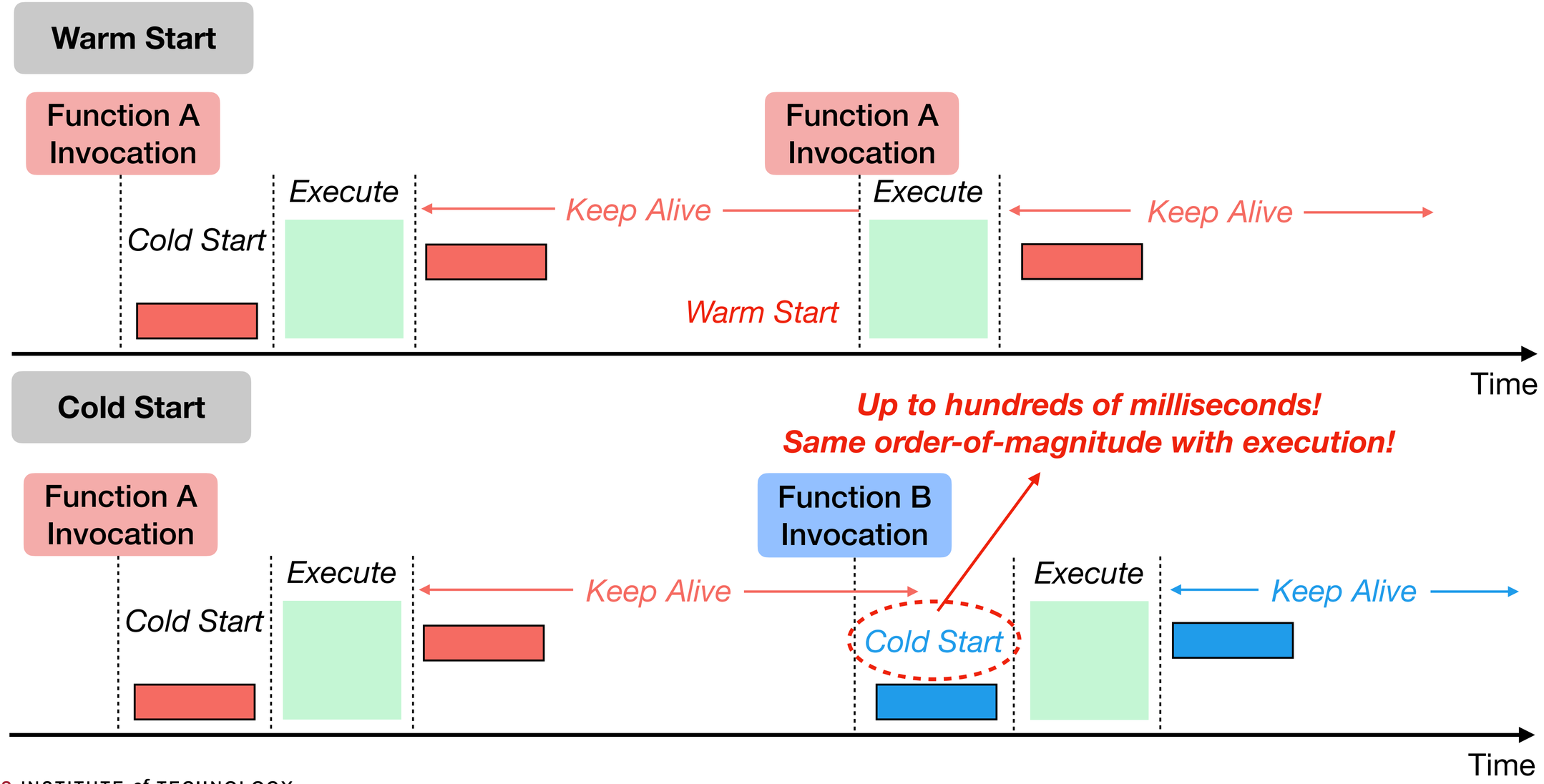
Serverless



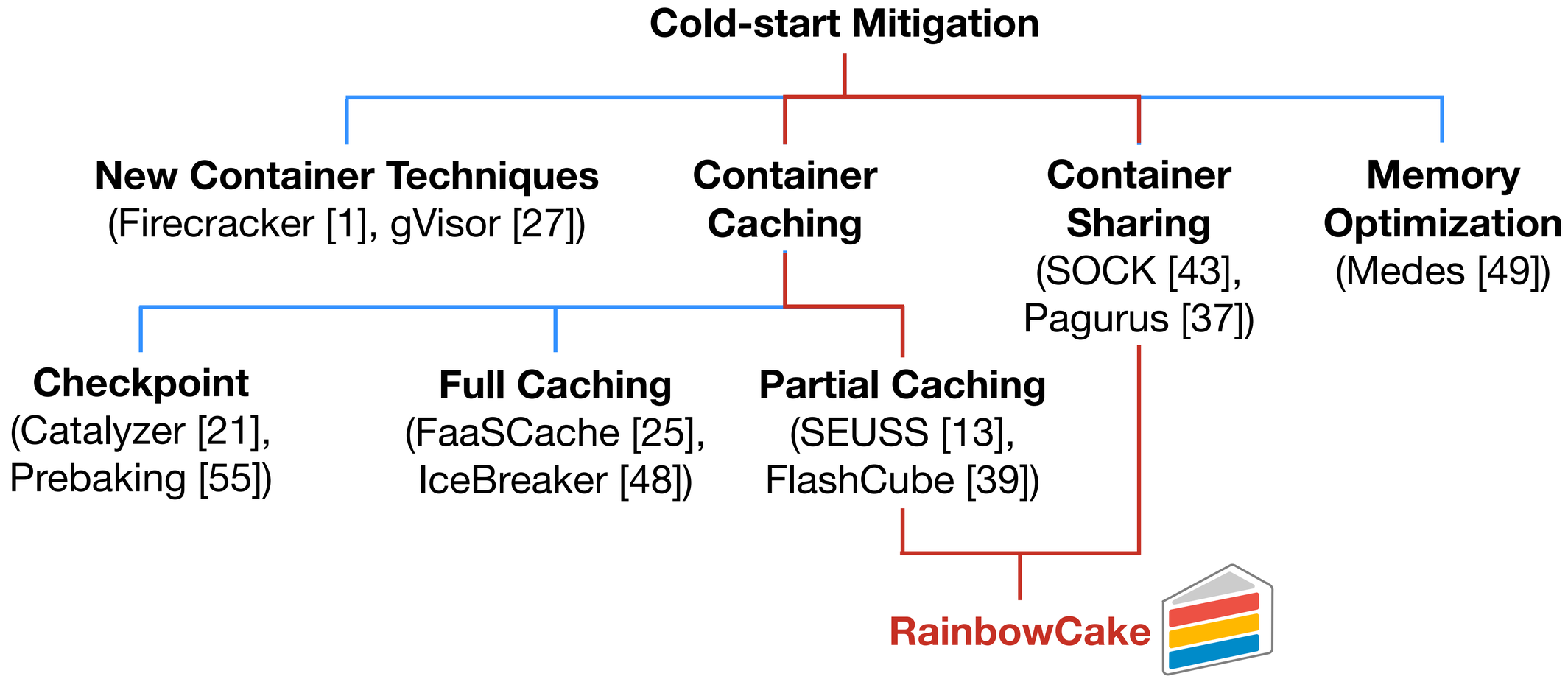
***Cruise* (Self-driving Taxi)**

From *A Serverless Vision for Cloud Computing* by Prof. Ana Klimovic

Cold-start in Serverless

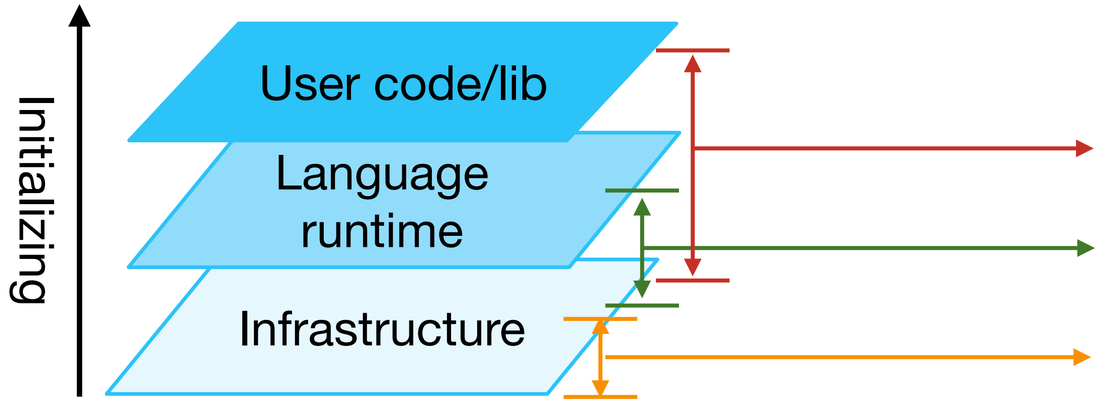


RainbowCake (ASPLOS'24)



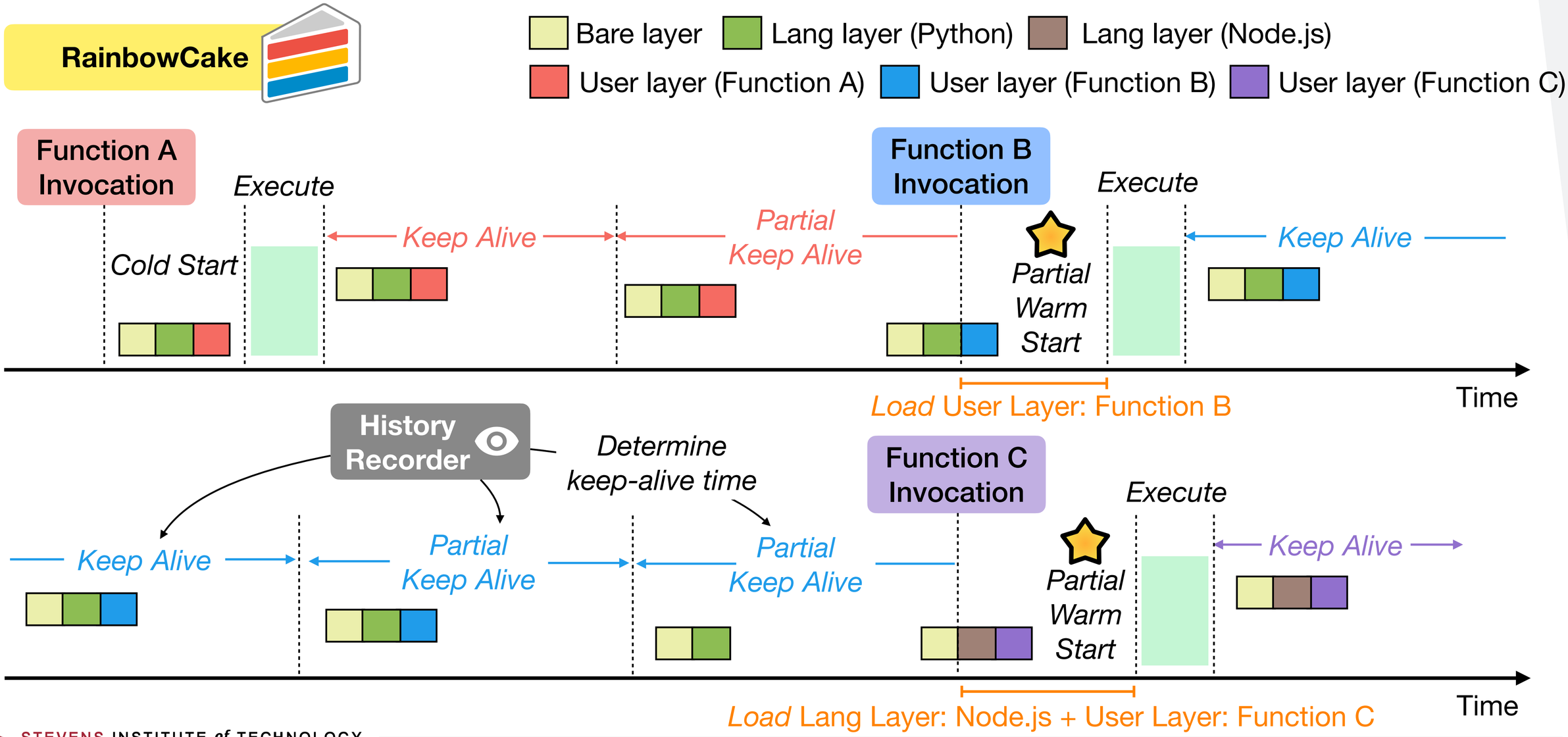
Layered Container Structure

- *Function startup goes through three layers*
  - **Bare layer**: infrastructure, environment, and utility preparation
  - **Lang layer**: language runtime creation
  - **User layer**: load user code and any necessary libraries



	Memory	Latency	Compatibility
User container	High	Low	Low
Lang container	Medium	Medium	Medium
Bare container	Low	High	High

RainbowCake Workflow



Combining container caching and sharing

Layer-wise pre-warming and keep-alive decisions

Mitigating cold-starts with minimal memory waste

RainbowCake

68%

Function startup latency reduction

77%

Memory waste reduction

STEVENS
INSTITUTE OF TECHNOLOGY
1870

THANK YOU

Stevens Institute of Technology
1 Castle Point Terrace, Hoboken, NJ 07030