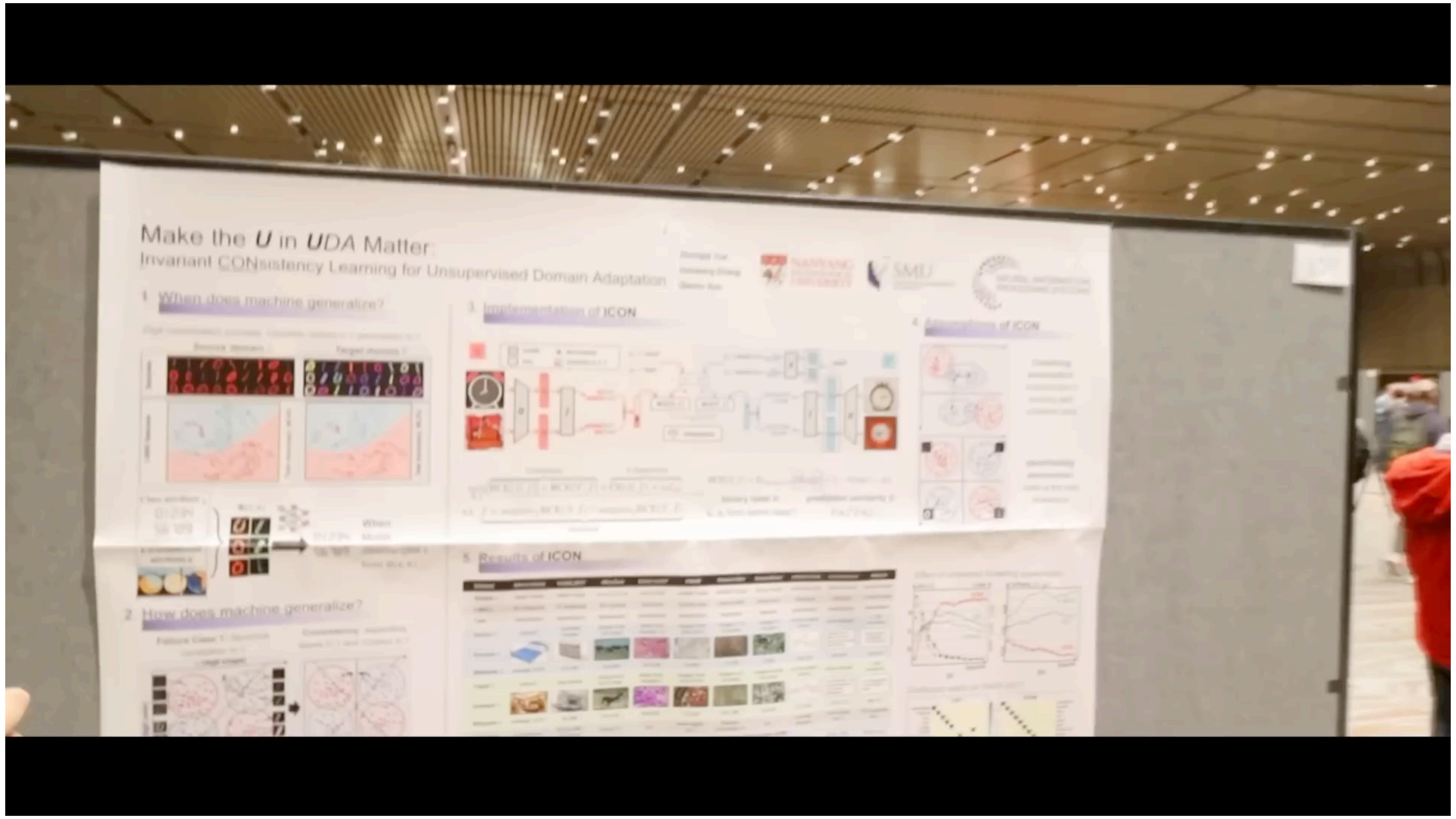
POSTER SESSION: GENERAL IDEA

Similar to a normal presentation

Key differences:

- Shorter talks
- Only 1 slide
- Direct interaction with speakers
- More flexible Q&A

In general more networking


Make the U in UDA Matter:
Invariant CONsistency Learning for Unsupervised Domain Adaptation
1 When does machine generalize?
3 Implementation of ICON
5 Results of ICON
2 How does machine generalize?

[NeurIPS 2023 Poster Session 1 (Tuesday Evening)](https://www.youtube.com/watch?v=oUqnvQm_k9M)
https://www.youtube.com/watch?v=oUqnvQm_k9M

POSTER SESSION: FORMALITIES PART 1

Date and time: 19.02.2025, 14:00 to 17:00

Location: OMZ U011

Poster size: A0

Plotting:

- We can plot your posters at our institute
- Please submit your posters as PDFs for printing to us no later than 17.02, 10 o’clock

All members of the group have to attend for the poster session

- If in-person is not possible, a virtual presentation can be arranged

POSTER SESSION: FORMALITIES PART 2

Presentations should last about 15 minutes

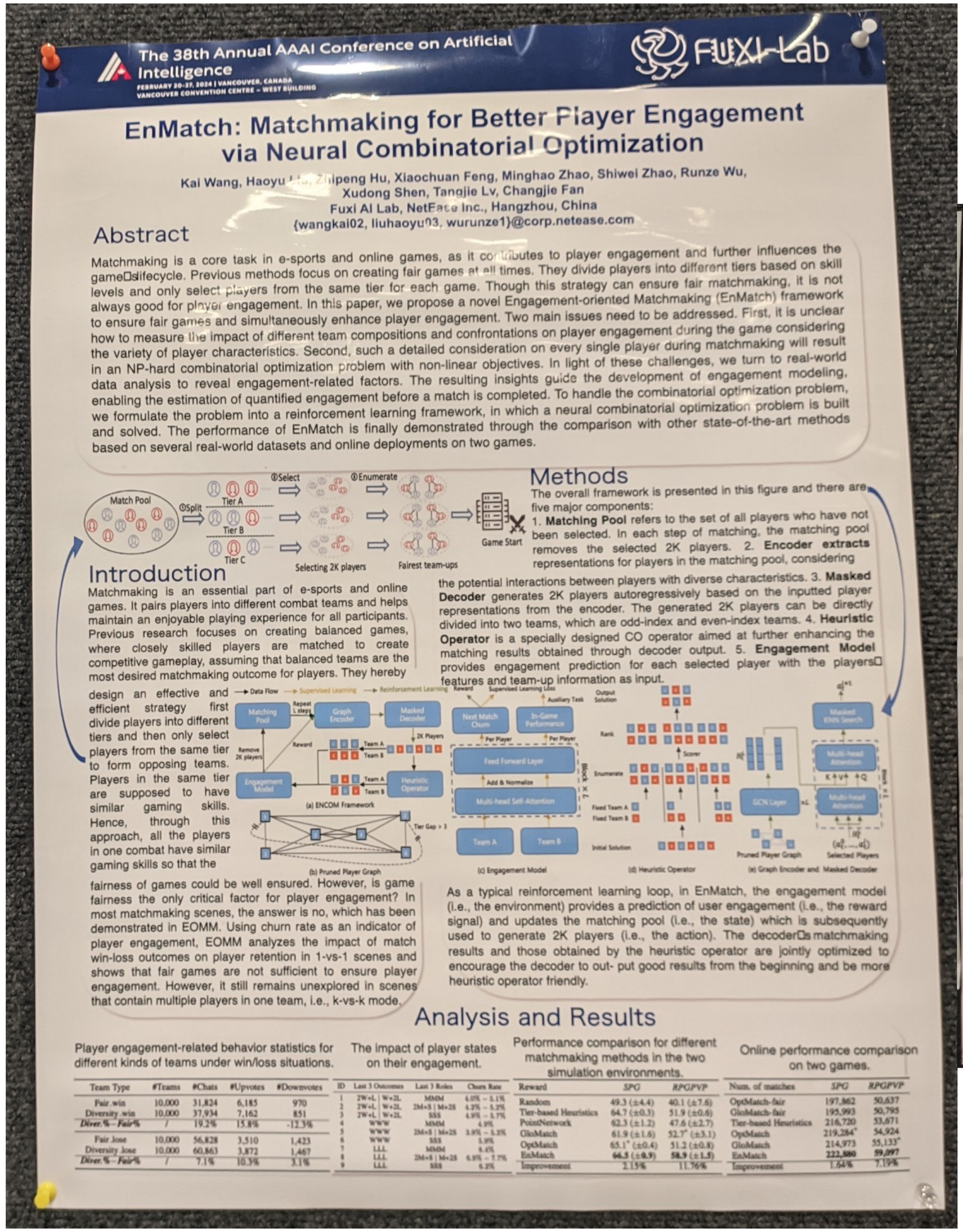
Afterwards 5 to 10 minutes of questions

Preliminary schedule

Time	
14:00	Setting up posters
14:15	First poster session: Groups 1, 3 and 4
15:30	Break
15:45	Second poster session: Groups 5, 6 and 7
17:00	Estimated end

WHAT MAKES A GOOD POSTER?

NOT SO GOOD POSTERS


The 38th Annual AAAI Conference on Artificial Intelligence
FEBRUARY 20-27, 2024 | VANCOUVER, CANADA
VANCOUVER CONVENTION CENTRE – WEST BUILDING
FUXI-Lab
Match Pool
①Split
Tier A
Tier B
Tier C
②Select
③Enumerate
Selecting 2K players
Fairest team-ups
Game Start
Data Flow
Supervised Learning
Reinforcement Learning
Matching Pool
Repeat L steps
Graph Encoder
Masked Decoder
2K Players
Reward
Remove 2K players
Team A
Team B
Engagement Model
Heuristic Operator
(a) ENCOM Framework
Tier Gap > 3
(b) Pruned Player Graph
Supervised Learning Loss
Auxiliary Task
Next Match Churn
In-Game Performance
Per Player
Feed Forward Layer
Add & Normalize
Multi-head Self-Attention
Block × L
(c) Engagement Model
Output Solution
Rank
Scorer
Enumerate
Fixed Team A
Fixed Team B
Initial Solution
(d) Heuristic Operator
GCN Layer
Masked KNN Search
Multi-head Attention
Pruned Player Graph
Selected Players
(e) Graph Encoder and Masked Decoder

EnMatch: Matchmaking for Better Player Engagement via Neural Combinatorial Optimization

Kai Wang, Haoyu Liu, Zhipeng Hu, Xiaochuan Feng, Minghao Zhao, Shiwei Zhao, Runze Wu, Xudong Shen, Tangjie Lv, Changjie Fan
Fuxi AI Lab, NetEase Inc., Hangzhou, China
{wangkai02, liuhaoyu03, wurunze1}@corp.netease.com

Abstract

Matchmaking is a core task in e-sports and online games, as it contributes to player engagement and further influences the game□slifecycle. Previous methods focus on creating fair games at all times. They divide players into different tiers based on skill levels and only select players from the same tier for each game. Though this strategy can ensure fair matchmaking, it is not always good for player engagement. In this paper, we propose a novel Engagement-oriented Matchmaking (EnMatch) framework to ensure fair games and simultaneously enhance player engagement. Two main issues need to be addressed. First, it is unclear how to measure the impact of different team compositions and confrontations on player engagement during the game considering the variety of player characteristics. Second, such a detailed consideration on every single player during matchmaking will result in an NP-hard combinatorial optimization problem with non-linear objectives. In light of these challenges, we turn to real-world data analysis to reveal engagement-related factors. The resulting insights guide the development of engagement modeling, enabling the estimation of quantified engagement before a match is completed. To handle the combinatorial optimization problem, we formulate the problem into a reinforcement learning framework, in which a neural combinatorial optimization problem is built and solved. The performance of EnMatch is finally demonstrated through the comparison with other state-of-the-art methods based on several real-world datasets and online deployments on two games.

Introduction

Matchmaking is an essential part of e-sports and online games. It pairs players into different combat teams and helps maintain an enjoyable playing experience for all participants. Previous research focuses on creating balanced games, where closely skilled players are matched to create competitive gameplay, assuming that balanced teams are the most desired matchmaking outcome for players. They hereby design an effective and efficient strategy first divide players into different tiers and then only select players from the same tier to form opposing teams. Players in the same tier are supposed to have similar gaming skills. Hence, through this approach, all the players in one combat have similar gaming skills so that the fairness of games could be well ensured. However, is game fairness the only critical factor for player engagement? In most matchmaking scenes, the answer is no, which has been demonstrated in EOMM. Using churn rate as an indicator of player engagement, EOMM analyzes the impact of match win-loss outcomes on player retention in 1-vs-1 scenes and shows that fair games are not sufficient to ensure player engagement. However, it still remains unexplored in scenes that contain multiple players in one team, i.e., k-vs-k mode.

Methods

The overall framework is presented in this figure and there are five major components:
1. **Matching Pool** refers to the set of all players who have not been selected. In each step of matching, the matching pool removes the selected 2K players. 2. **Encoder extracts** representations for players in the matching pool, considering the potential interactions between players with diverse characteristics. 3. **Masked Decoder** generates 2K players autoregressively based on the inputted player representations from the encoder. The generated 2K players can be directly divided into two teams, which are odd-index and even-index teams. 4. **Heuristic Operator** is a specially designed CO operator aimed at further enhancing the matching results obtained through decoder output. 5. **Engagement Model** provides engagement prediction for each selected player with the players□ features and team-up information as input.

As a typical reinforcement learning loop, in EnMatch, the engagement model (i.e., the environment) provides a prediction of user engagement (i.e., the reward signal) and updates the matching pool (i.e., the state) which is subsequently used to generate 2K players (i.e., the action). The decoder□s matchmaking results and those obtained by the heuristic operator are jointly optimized to encourage the decoder to out- put good results from the beginning and be more heuristic operator friendly.

Analysis and Results

Player engagement-related behavior statistics for different kinds of teams under win/loss situations.

Team Type	#Teams	#Chats	#Upvotes	#Downvotes
Fair win	10,000	31,824	6,185	970
Diversity win	10,000	37,934	7,162	851
Diver.%–Fair%	/	19.2%	15.8%	-12.3%
Fair lose	10,000	56,828	3,510	1,423
Diversity lose	10,000	60,863	3,872	1,467
Diver.%–Fair%	/	7.1%	10.3%	3.1%

The impact of player states on their engagement.

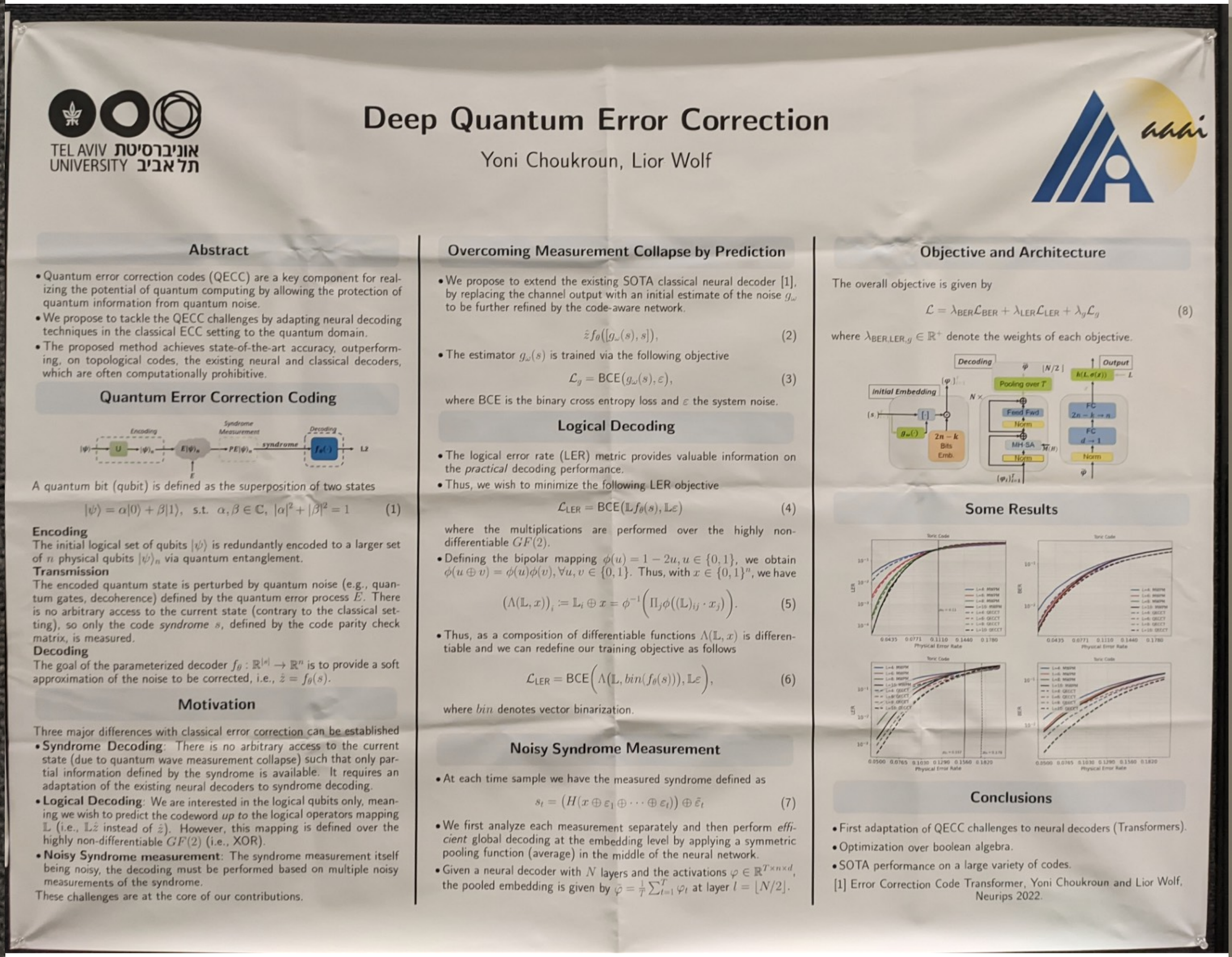
ID	Last 3 Outcomes	Last 3 Roles	Churn Rate
1	2W+L | W+2L	MMM	4.0% – 5.1%
2	2W+L | W+2L	2M+S | M+2S	4.2% – 5.2%
3	2W+L | W+2L	SSS	4.9% – 5.7%
4	WWW	MMM	4.9%
5	WWW	2M+S | M+2S	3.9% – 5.3%
6	WWW	SSS	5.9%
7	LLL	MMM	6.4%
8	LLL	2M+S | M+2S	6.9% – 7.7%
9	LLL	SSS	6.2%

Performance comparison for different matchmaking methods in the two simulation environments.

Reward	*SPG*	*RPGPVP*
Random	49.3 (±4.4)	40.1 (±7.6)
Tier-based Heuristics	64.7 (±0.3)	51.9 (±0.6)
PointNetwork	62.3 (±1.2)	47.6 (±2.7)
GloMatch	61.9 (±1.6)	52.7* (±3.1)
OptMatch	65.1* (±0.4)	51.2 (±0.8)
EnMatch	**66.5** (±**0.9**)	**58.9** (±**1.5**)
Improvement	2.15%	11.76%

Online performance comparison on two games.

Num. of matches	*SPG*	*RPGPVP*
OptMatch-fair	197,862	50,637
GloMatch-fair	195,993	50,795
Tier-based Heuristics	216,720	53,671
OptMatch	219,284*	54,924
GloMatch	214,973	55,133*
EnMatch	**222,880**	**59,097**
Improvement	1.64%	7.19%


TEL AVIV UNIVERSITY אוניברסיטת תל אביב
aaai
Encoding
Syndrome Measurement
Decoding
$|\psi\rangle$
U
$|\psi\rangle_n$
$E|\psi\rangle_n$
$PE|\psi\rangle_n$
syndrome
$f_\theta(\cdot)$
L2
E
Decoding
Initial Embedding
(s_t)
[·]
$g_\omega(\cdot)$
2n − k Bits Emb.
N ×
Feed Fwd
Norm
MH-SA
Pooling over T
$\lfloor N/2 \rfloor$
Output
$h(L, \sigma(x))$
FC 2n − k → n
FC d → 1
Norm
Toric Code
LER
BER
Physical Error Rate
L=4 MWPM
L=6 MWPM
L=8 MWPM
L=10 MWPM
L=4 QECCT
L=6 QECCT
L=8 QECCT
L=10 QECCT

Deep Quantum Error Correction

Yoni Choukroun, Lior Wolf

Abstract

- Quantum error correction codes (QECC) are a key component for realizing the potential of quantum computing by allowing the protection of quantum information from quantum noise.
- We propose to tackle the QECC challenges by adapting neural decoding techniques in the classical ECC setting to the quantum domain.
- The proposed method achieves state-of-the-art accuracy, outperforming, on topological codes, the existing neural and classical decoders, which are often computationally prohibitive.

Quantum Error Correction Coding

A quantum bit (qubit) is defined as the superposition of two states

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \text{ s.t. } \alpha, \beta \in \mathbb{C}, |\alpha|^2 + |\beta|^2 = 1 \quad (1)$$

Encoding
The initial logical set of qubits $|\psi\rangle$ is redundantly encoded to a larger set of n physical qubits $|\psi\rangle_n$ via quantum entanglement.
Transmission
The encoded quantum state is perturbed by quantum noise (e.g., quantum gates, decoherence) defined by the quantum error process E. There is no arbitrary access to the current state (contrary to the classical setting), so only the code *syndrome* s, defined by the code parity check matrix, is measured.
Decoding
The goal of the parameterized decoder $f_\theta : \mathbb{R}^{|s|} \to \mathbb{R}^n$ is to provide a soft approximation of the noise to be corrected, i.e., $\hat{z} = f_\theta(s)$.

Motivation

Three major differences with classical error correction can be established
- **Syndrome Decoding**: There is no arbitrary access to the current state (due to quantum wave measurement collapse) such that only partial information defined by the syndrome is available. It requires an adaptation of the existing neural decoders to syndrome decoding.
- **Logical Decoding**: We are interested in the logical qubits only, meaning we wish to predict the codeword *up to* the logical operators mapping $\mathbb{L}$ (i.e., $\mathbb{L}\hat{z}$ instead of $\hat{z}$). However, this mapping is defined over the highly non-differentiable $GF(2)$ (i.e., XOR).
- **Noisy Syndrome measurement**: The syndrome measurement itself being noisy, the decoding must be performed based on multiple noisy measurements of the syndrome.

These challenges are at the core of our contributions.

Overcoming Measurement Collapse by Prediction

- We propose to extend the existing SOTA classical neural decoder [1], by replacing the channel output with an initial estimate of the noise g_ω to be further refined by the code-aware network.

$$\hat{z} f_\theta([g_\omega(s), s]), \quad (2)$$

- The estimator $g_\omega(s)$ is trained via the following objective

$$\mathcal{L}_g = \text{BCE}(g_\omega(s), \varepsilon), \quad (3)$$

where BCE is the binary cross entropy loss and ε the system noise.

Logical Decoding

- The logical error rate (LER) metric provides valuable information on the *practical* decoding performance.
- Thus, we wish to minimize the following LER objective

$$\mathcal{L}_{\text{LER}} = \text{BCE}(\mathbb{L} f_\theta(s), \mathbb{L}\varepsilon) \quad (4)$$

where the multiplications are performed over the highly non-differentiable $GF(2)$.
- Defining the bipolar mapping $\phi(u) = 1 - 2u, u \in \{0,1\}$, we obtain $\phi(u \oplus v) = \phi(u)\phi(v), \forall u, v \in \{0,1\}$. Thus, with $x \in \{0,1\}^n$, we have

$$\left(\Lambda(\mathbb{L}, x)\right)_i := \mathbb{L}_i \oplus x = \phi^{-1}\Big(\Pi_j \phi((\mathbb{L})_{ij} \cdot x_j)\Big). \quad (5)$$

- Thus, as a composition of differentiable functions $\Lambda(\mathbb{L}, x)$ is differentiable and we can redefine our training objective as follows

$$\mathcal{L}_{\text{LER}} = \text{BCE}\Big(\Lambda(\mathbb{L}, bin(f_\theta(s))), \mathbb{L}\varepsilon\Big), \quad (6)$$

where bin denotes vector binarization.

Noisy Syndrome Measurement

- At each time sample we have the measured syndrome defined as

$$s_t = (H(x \oplus \varepsilon_1 \oplus \cdots \oplus \varepsilon_t)) \oplus \tilde{\varepsilon}_t \quad (7)$$

- We first analyze each measurement separately and then perform *efficient* global decoding at the embedding level by applying a symmetric pooling function (average) in the middle of the neural network.
- Given a neural decoder with N layers and the activations $\varphi \in \mathbb{R}^{T \times n \times d}$, the pooled embedding is given by $\bar{\varphi} = \frac{1}{T}\sum_{t=1}^{T} \varphi_t$ at layer $l = \lfloor N/2 \rfloor$.

Objective and Architecture

The overall objective is given by

$$\mathcal{L} = \lambda_{\text{BER}}\mathcal{L}_{\text{BER}} + \lambda_{\text{LER}}\mathcal{L}_{\text{LER}} + \lambda_g\mathcal{L}_g \quad (8)$$

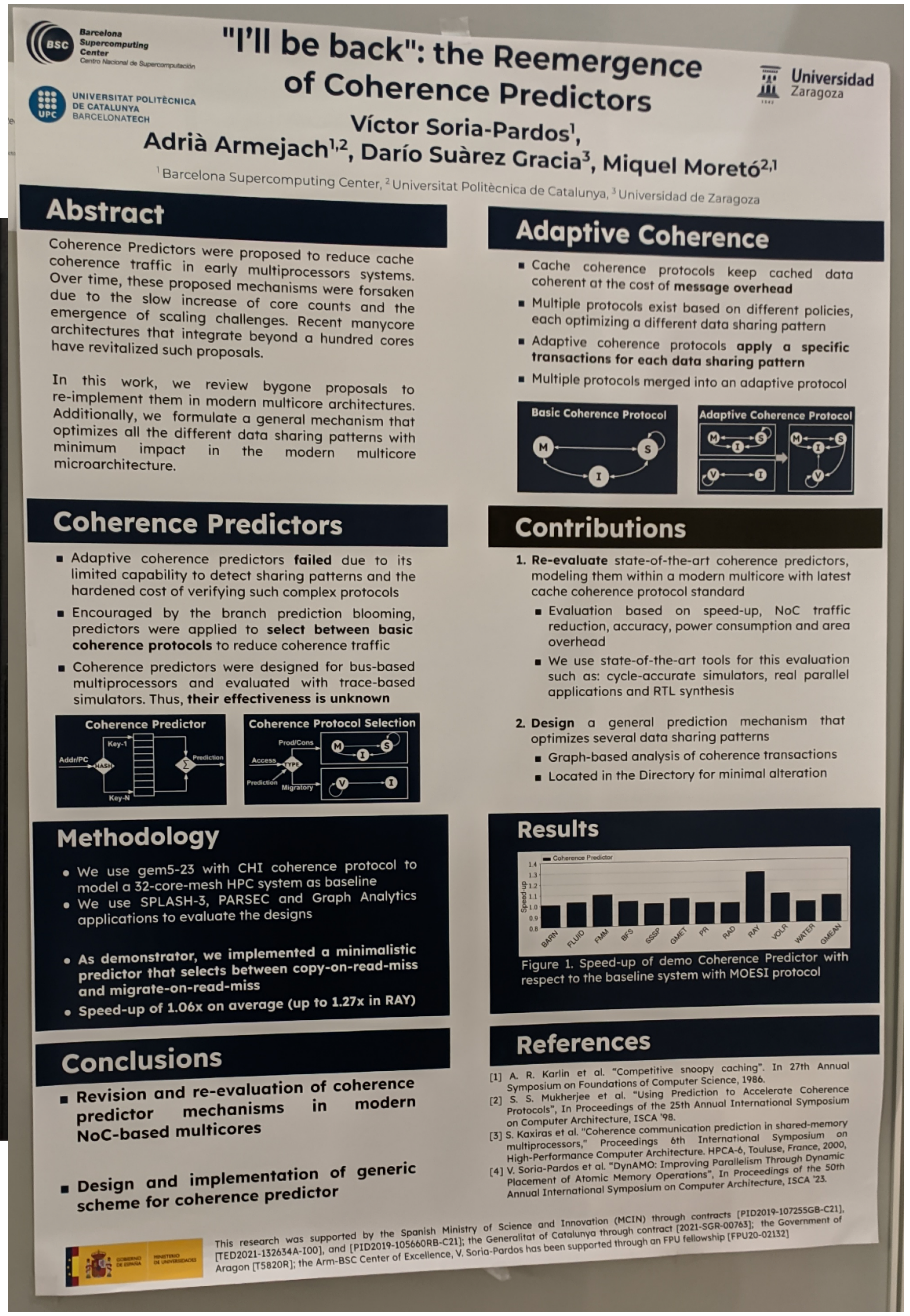
where $\lambda_{\text{BER,LER},g} \in \mathbb{R}^+$ denote the weights of each objective.

Some Results

Conclusions

- First adaptation of QECC challenges to neural decoders (Transformers).
- Optimization over boolean algebra.
- SOTA performance on a large variety of codes.

[1] Error Correction Code Transformer, Yoni Choukroun and Lior Wolf, Neurips 2022.


BSC Barcelona Supercomputing Center Centro Nacional de Supercomputación
UPC UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH
Universidad Zaragoza
Basic Coherence Protocol
Adaptive Coherence Protocol
M
S
I
V
Coherence Predictor
Addr/PC
HASH
Key-1
Key-N
Σ
Prediction
Coherence Protocol Selection
Access
TYPE
Prod/Cons
Migratory
Prediction
Coherence Predictor
Speed-up
1.4
1.3
1.2
1.1
1.0
0.9
0.8
BARN
FLUID
FMM
BFS
SSSP
GMET
PR
RAD
RAY
VOLR
WATER
GMEAN
GOBIERNO DE ESPAÑA
MINISTERIO DE UNIVERSIDADES

"I'll be back": the Reemergence of Coherence Predictors

Víctor Soria-Pardos[1], Adrià Armejach[1,2], Darío Suàrez Gracia[3], Miquel Moretó[2,1]
[1] Barcelona Supercomputing Center, [2] Universitat Politècnica de Catalunya, [3] Universidad de Zaragoza

Abstract

Coherence Predictors were proposed to reduce cache coherence traffic in early multiprocessors systems. Over time, these proposed mechanisms were forsaken due to the slow increase of core counts and the emergence of scaling challenges. Recent manycore architectures that integrate beyond a hundred cores have revitalized such proposals.

In this work, we review bygone proposals to re-implement them in modern multicore architectures. Additionally, we formulate a general mechanism that optimizes all the different data sharing patterns with minimum impact in the modern multicore microarchitecture.

Coherence Predictors

- Adaptive coherence predictors **failed** due to its limited capability to detect sharing patterns and the hardened cost of verifying such complex protocols
- Encouraged by the branch prediction blooming, predictors were applied to **select between basic coherence protocols** to reduce coherence traffic
- Coherence predictors were designed for bus-based multiprocessors and evaluated with trace-based simulators. Thus, **their effectiveness is unknown**

Methodology

- We use gem5-23 with CHI coherence protocol to model a 32-core-mesh HPC system as baseline
- We use SPLASH-3, PARSEC and Graph Analytics applications to evaluate the designs
- **As demonstrator, we implemented a minimalistic predictor that selects between copy-on-read-miss and migrate-on-read-miss**
- **Speed-up of 1.06x on average (up to 1.27x in RAY)**

Conclusions

- **Revision and re-evaluation of coherence predictor mechanisms in modern NoC-based multicores**
- **Design and implementation of generic scheme for coherence predictor**

Adaptive Coherence

- Cache coherence protocols keep cached data coherent at the cost of **message overhead**
- Multiple protocols exist based on different policies, each optimizing a different data sharing pattern
- Adaptive coherence protocols **apply a specific transactions for each data sharing pattern**
- Multiple protocols merged into an adaptive protocol

Contributions

1. **Re-evaluate** state-of-the-art coherence predictors, modeling them within a modern multicore with latest cache coherence protocol standard
   - Evaluation based on speed-up, NoC traffic reduction, accuracy, power consumption and area overhead
   - We use state-of-the-art tools for this evaluation such as: cycle-accurate simulators, real parallel applications and RTL synthesis
2. **Design** a general prediction mechanism that optimizes several data sharing patterns
   - Graph-based analysis of coherence transactions
   - Located in the Directory for minimal alteration

Results

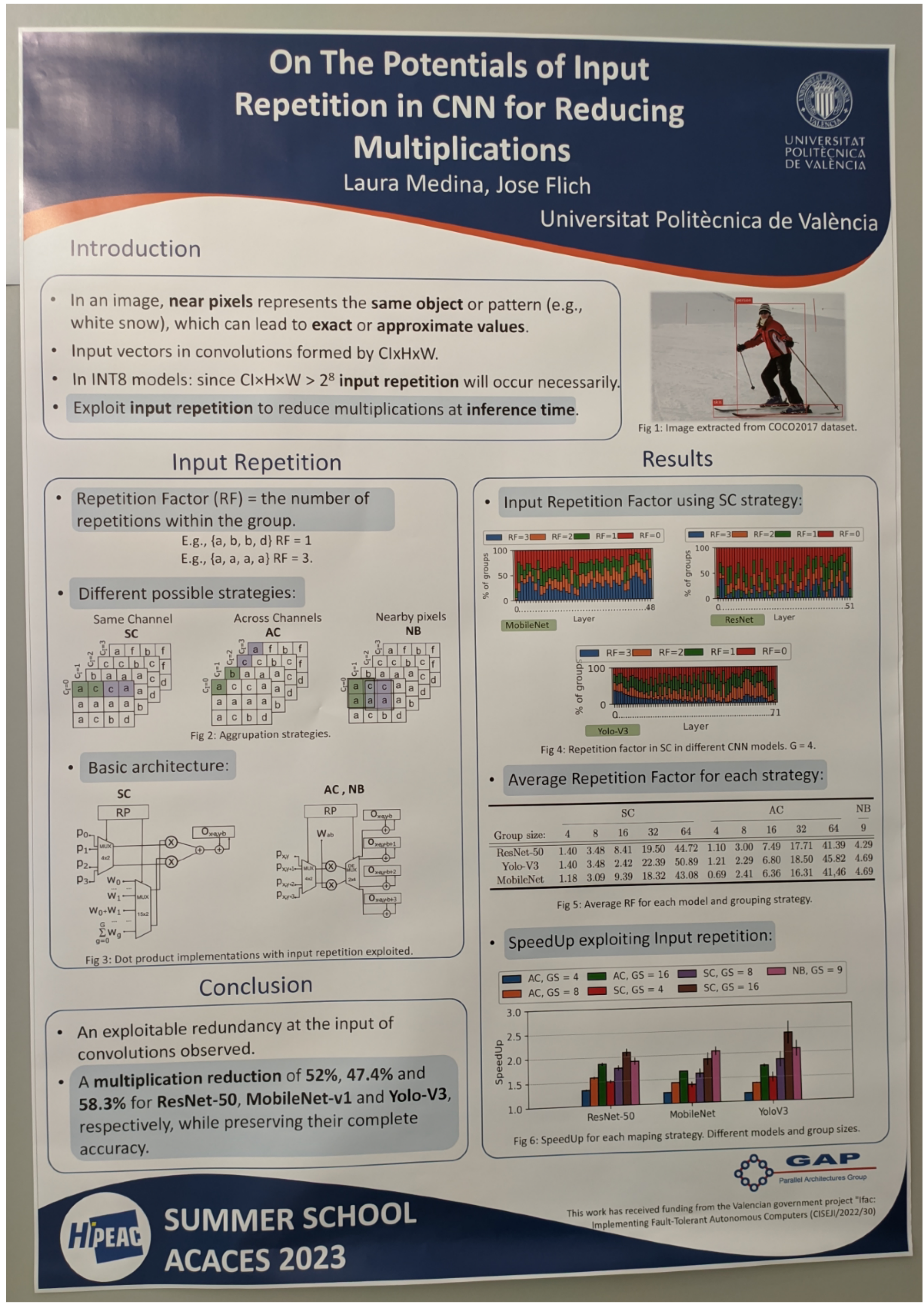
Figure 1. Speed-up of demo Coherence Predictor with respect to the baseline system with MOESI protocol

References

[1] A. R. Karlin et al. "Competitive snoopy caching". In 27th Annual Symposium on Foundations of Computer Science, 1986.
[2] S. S. Mukherjee et al. "Using Prediction to Accelerate Coherence Protocols", In Proceedings of the 25th Annual International Symposium on Computer Architecture, ISCA '98.
[3] S. Kaxiras et al. "Coherence communication prediction in shared-memory multiprocessors," Proceedings 6th International Symposium on High-Performance Computer Architecture. HPCA-6, Touluse, France, 2000.
[4] V. Soria-Pardos et al. "DynAMO: Improving Parallelism Through Dynamic Placement of Atomic Memory Operations", In Proceedings of the 50th Annual International Symposium on Computer Architecture, ISCA '23.

This research was supported by the Spanish Ministry of Science and Innovation (MCIN) through contracts [PID2019-107255GB-C21], [TED2021-132634A-I00], and [PID2019-105660RB-C21]; the Generalitat of Catalunya through contract [2021-SGR-00763]; the Government of Aragon [T5820R]; the Arm-BSC Center of Excellence, V. Soria-Pardos has been supported through an FPU fellowship [FPU20-02132]

GOOD POSTERS



On The Potentials of Input Repetition in CNN for Reducing Multiplications

Laura Medina, Jose Flich

Universitat Politècnica de València

Introduction

- In an image, **near pixels** represents the **same object** or pattern (e.g., white snow), which can lead to **exact** or **approximate values**.
- Input vectors in convolutions formed by ClxHxW.
- In INT8 models: since Cl×H×W > 2^8 **input repetition** will occur necessarily.
- Exploit **input repetition** to reduce multiplications at **inference time**.

Fig 1: Image extracted from COCO2017 dataset.

Input Repetition

- Repetition Factor (RF) = the number of repetitions within the group.

  E.g., {a, b, b, d} RF = 1

  E.g., {a, a, a, a} RF = 3.

- Different possible strategies:

Fig 2: Aggrupation strategies.

- Basic architecture:

Fig 3: Dot product implementations with input repetition exploited.

Results

- Input Repetition Factor using SC strategy:

Fig 4: Repetition factor in SC in different CNN models. G = 4.

- Average Repetition Factor for each strategy:

	SC					AC					NB
Group size:	4	8	16	32	64	4	8	16	32	64	9
ResNet-50	1.40	3.48	8.41	19.50	44.72	1.10	3.00	7.49	17.71	41.39	4.29
Yolo-V3	1.40	3.48	2.42	22.39	50.89	1.21	2.29	6.80	18.50	45.82	4.69
MobileNet	1.18	3.09	9.39	18.32	43.08	0.69	2.41	6.36	16.31	41,46	4.69

Fig 5: Average RF for each model and grouping strategy.

- SpeedUp exploiting Input repetition:

Fig 6: SpeedUp for each maping strategy. Different models and group sizes.

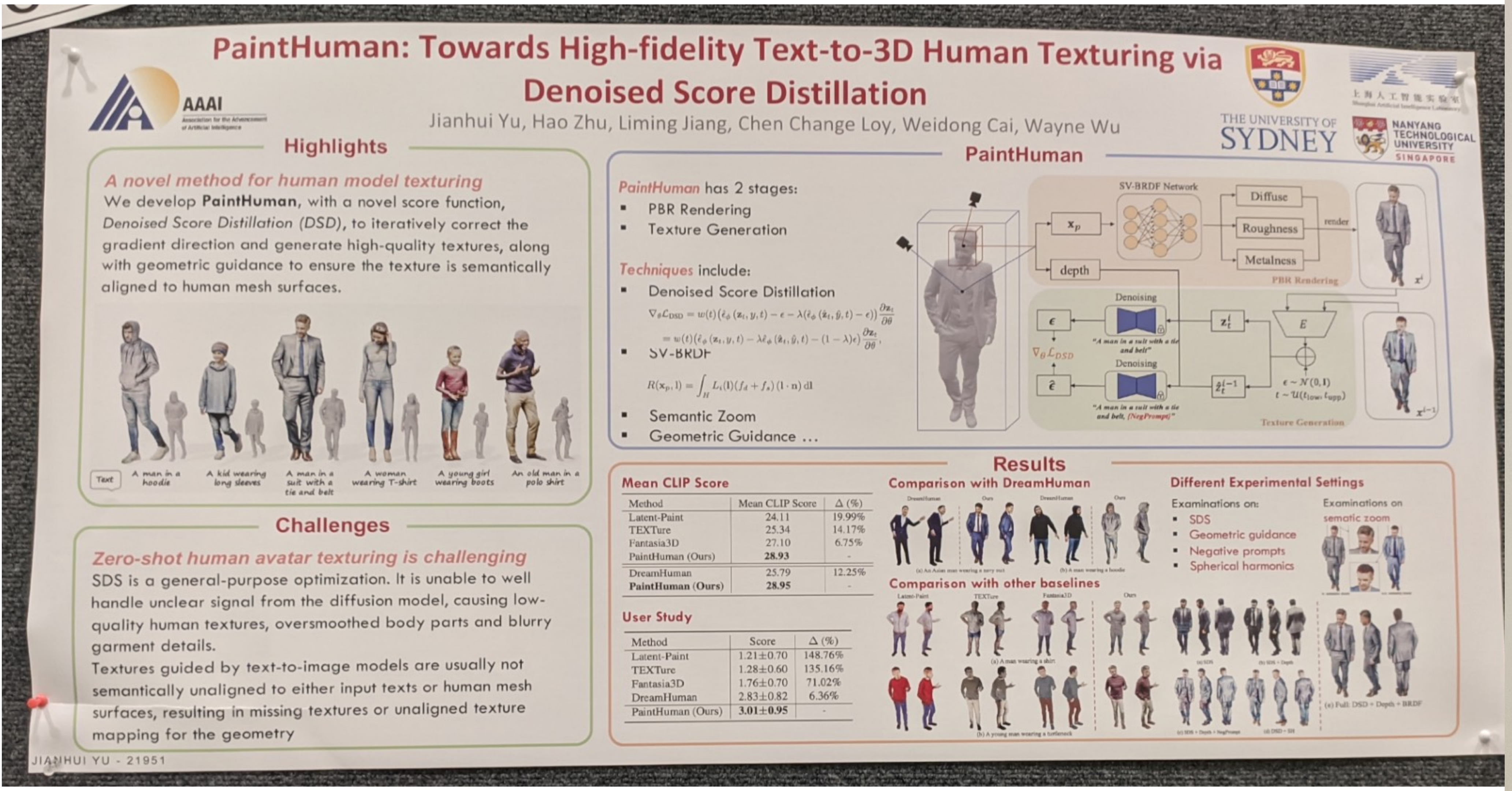
Conclusion

- An exploitable redundancy at the input of convolutions observed.
- A **multiplication reduction** of **52%**, **47.4%** and **58.3%** for **ResNet-50**, **MobileNet-v1** and **Yolo-V3**, respectively, while preserving their complete accuracy.

This work has received funding from the Valencian government project "Ifac: Implementing Fault-Tolerant Autonomous Computers (CISEJI/2022/30)

GAP Parallel Architectures Group

SUMMER SCHOOL ACACES 2023



PaintHuman: Towards High-fidelity Text-to-3D Human Texturing via Denoised Score Distillation

Jianhui Yu, Hao Zhu, Liming Jiang, Chen Change Loy, Weidong Cai, Wayne Wu

Highlights

A novel method for human model texturing

We develop **PaintHuman**, with a novel score function, *Denoised Score Distillation (DSD)*, to iteratively correct the gradient direction and generate high-quality textures, along with geometric guidance to ensure the texture is semantically aligned to human mesh surfaces.

Text
A man in a hoodie
A kid wearing long sleeves
A man in a suit with a tie and belt
A woman wearing T-shirt
A young girl wearing boots
An old man in a polo shirt

Challenges

Zero-shot human avatar texturing is challenging

SDS is a general-purpose optimization. It is unable to well handle unclear signal from the diffusion model, causing low-quality human textures, oversmoothed body parts and blurry garment details.

Textures guided by text-to-image models are usually not semantically unaligned to either input texts or human mesh surfaces, resulting in missing textures or unaligned texture mapping for the geometry

PaintHuman

PaintHuman has 2 stages:

- PBR Rendering
- Texture Generation

Techniques include:

- Denoised Score Distillation

$$\nabla_\theta \mathcal{L}_{DSD} = w(t)(\hat{\epsilon}_\phi(\mathbf{z}_t, y, t) - \epsilon - \lambda(\hat{\epsilon}_\phi(\hat{\mathbf{z}}_t, \hat{y}, t) - \epsilon))\frac{\partial \mathbf{z}_t}{\partial \theta}$$
$$= w(t)(\hat{\epsilon}_\phi(\mathbf{z}_t, y, t) - \lambda\hat{\epsilon}_\phi(\hat{\mathbf{z}}_t, \hat{y}, t) - (1-\lambda)\epsilon)\frac{\partial \mathbf{z}_t}{\partial \theta},$$

- SV-BRDF

$$R(\mathbf{x}_p, \mathbf{l}) = \int_H L_i(\mathbf{l})(f_d + f_s)(\mathbf{l} \cdot \mathbf{n})\, d\mathbf{l}$$

- Semantic Zoom
- Geometric Guidance …

SV-BRDF Network
x_p
depth
Diffuse
Roughness
Metalness
render
PBR Rendering
Denoising
"A man in a suit with a tie and belt"
"A man in a suit with a tie and belt, [NegPrompt]"
$\nabla_\theta \mathcal{L}_{DSD}$
E
$\epsilon \sim \mathcal{N}(0, \mathbf{I})$
$t \sim \mathcal{U}(t_{low}, t_{upp})$
Texture Generation

Results

Mean CLIP Score

Method	Mean CLIP Score	Δ (%)
Latent-Paint	24.11	19.99%
TEXTure	25.34	14.17%
Fantasia3D	27.10	6.75%
PaintHuman (Ours)	**28.93**	-
DreamHuman	25.79	12.25%
PaintHuman (Ours)	**28.95**	-

User Study

Method	Score	Δ (%)
Latent-Paint	1.21±0.70	148.76%
TEXTure	1.28±0.60	135.16%
Fantasia3D	1.76±0.70	71.02%
DreamHuman	2.83±0.82	6.36%
PaintHuman (Ours)	**3.01±0.95**	-

Comparison with DreamHuman

Comparison with other baselines

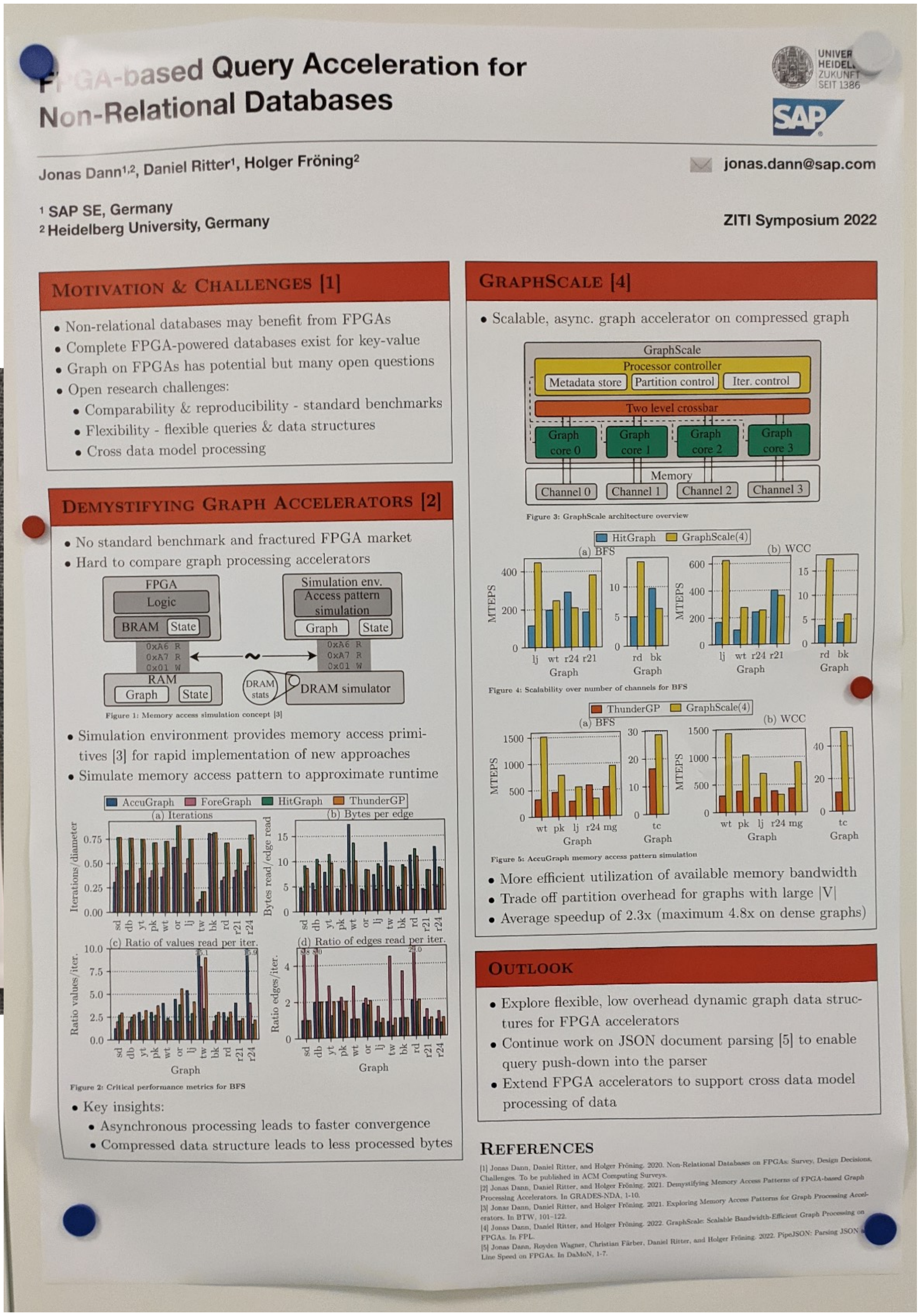
Different Experimental Settings

Examinations on:

- SDS
- Geometric guidance
- Negative prompts
- Spherical harmonics

Examinations on sematic zoom

JIANHUI YU - 21951



FPGA-based Query Acceleration for Non-Relational Databases

Jonas Dann[1,2], Daniel Ritter[1], Holger Fröning[2]

jonas.dann@sap.com

[1] SAP SE, Germany
[2] Heidelberg University, Germany

ZITI Symposium 2022

Motivation & Challenges [1]

- Non-relational databases may benefit from FPGAs
- Complete FPGA-powered databases exist for key-value
- Graph on FPGAs has potential but many open questions
- Open research challenges:
  - Comparability & reproducibility - standard benchmarks
  - Flexibility - flexible queries & data structures
  - Cross data model processing

Demystifying Graph Accelerators [2]

- No standard benchmark and fractured FPGA market
- Hard to compare graph processing accelerators

Figure 1: Memory access simulation concept [3]

- Simulation environment provides memory access primitives [3] for rapid implementation of new approaches
- Simulate memory access pattern to approximate runtime

Figure 2: Critical performance metrics for BFS

- Key insights:
  - Asynchronous processing leads to faster convergence
  - Compressed data structure leads to less processed bytes

GraphScale [4]

- Scalable, async. graph accelerator on compressed graph

Figure 3: GraphScale architecture overview

Figure 4: Scalability over number of channels for BFS

Figure 5: AccuGraph memory access pattern simulation

- More efficient utilization of available memory bandwidth
- Trade off partition overhead for graphs with large |V|
- Average speedup of 2.3x (maximum 4.8x on dense graphs)

Outlook

- Explore flexible, low overhead dynamic graph data structures for FPGA accelerators
- Continue work on JSON document parsing [5] to enable query push-down into the parser
- Extend FPGA accelerators to support cross data model processing of data

References

[1] Jonas Dann, Daniel Ritter, and Holger Fröning. 2020. Non-Relational Databases on FPGAs: Survey, Design Decisions, Challenges. To be published in ACM Computing Surveys.

[2] Jonas Dann, Daniel Ritter, and Holger Fröning. 2021. Demystifying Memory Access Patterns of FPGA-based Graph Processing Accelerators. In GRADES-NDA, 1-10.

[3] Jonas Dann, Daniel Ritter, and Holger Fröning. 2021. Exploring Memory Access Patterns for Graph Processing Accelerators. In BTW, 101-122.

[4] Jonas Dann, Daniel Ritter, and Holger Fröning. 2022. GraphScale: Scalable Bandwidth-Efficient Graph Processing on FPGAs. In FPL.

[5] Jonas Dann, Royden Wagner, Christian Färber, Daniel Ritter, and Holger Fröning. 2022. PipeJSON: Parsing JSON at Line Speed on FPGAs. In DaMoN, 1-7.

WHAT MAKES A GOOD POSTER?

- Large pictures
  - Maybe even many
- Little text
  - But in large font
- Concentrate on ONE main story