CogSys: Efficient and Scalable Neuro-Symbolic Cognition System via Algorithm-Hardware Co-Design

Zishen Wan[1*], Hanchen Yang[1*], Ritik Raj[1*], Che-Kai Liu[1], Ananda Samajdar[2], Arijit Raychowdhury[1], Tushar Krishna[1]

[1]Georgia Institute of Technology, Atlanta, GA
[2]IBM Research, Yorktown Heights, NY
([*]Equal Contributions)

IEEE International Symposium on High-Performance Computer Architecture (HPCA), 2025

Executive Summary

- **Understand** neuro-symbolic workloads from architectural and system perspectives.

- Identify **optimization opportunities** for neuro-symbolic systems.

- Demonstrate scalability and efficiency improvement of neuro-symbolic workload via **co-designed** system.

Neural Networks in Our Daily Life

Image Recognition

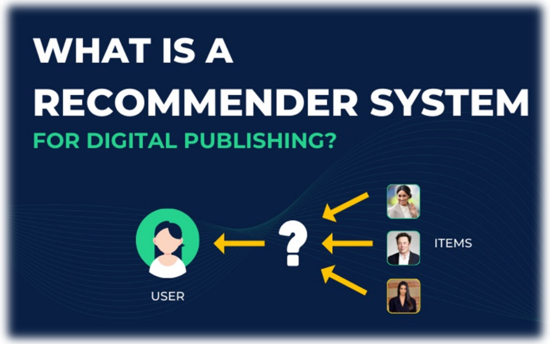
Speech Recognition

Language Translation

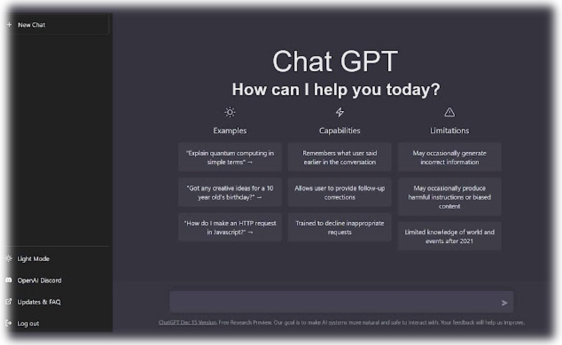
Autonomous Vehicle

Medical Diagnosis

Financial Services

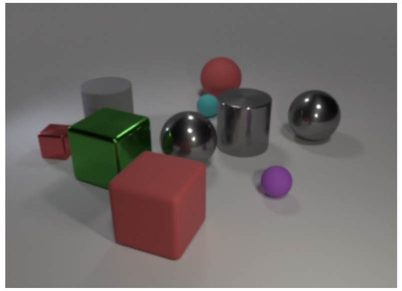


Recommendation Systems



ChatGPT

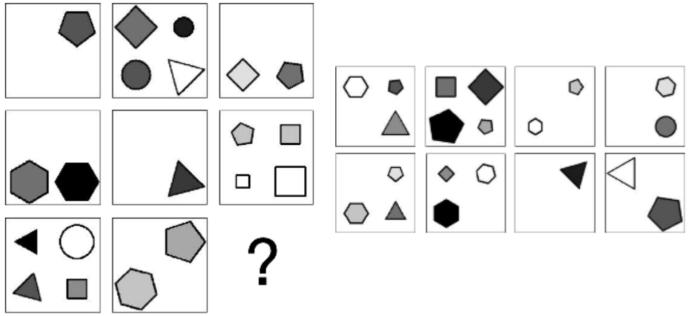
But… Is That Enough?



(i) *Remove all gray spheres. How many spheres are there?* (3), (ii) *Take away 3 cubes. How many objects are there?* (7), (iii) *How many blocks must be removed to get 1 block?* (2)

Complex Question Answering

NN accuracy: 50%

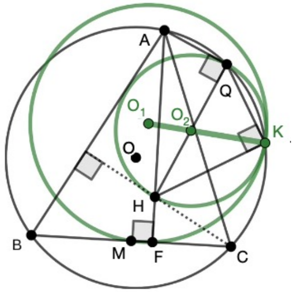


Abstract Reasoning

NN accuracy: 53%

IMO 2015 P3

"Let ABC be an acute triangle. Let (O) be its circumcircle, H its orthocenter, and F the foot of the altitude from A. Let M be the midpoint of BC. Let Q be the point on (O) such that QH ⊥ QA and let K be the point on (O) such that KH ⊥ KQ. Prove that the circumcircles (O_1) and (O_2) of triangles FKM and KQH are tangent to each other."



Automated Theorem Proving

NN accuracy: 20%

Interactive Learning

NN accuracy: 71%

Scenario
Imagine that a stranger will give Hank one thousand dollars to break all the windows in his neighbor's house without his neighbor's permission. Hank carries out the stranger's request.
Imagine that there are five people who are waiting in line to use a single-occupancy bathroom at a concert venue. Someone at the back of the line needs to throw up immediately. That person skips to the front of the line instead of waiting in the back.
At a summer camp, there is a pool. Right next to the pool is a tent where the kids at the camp have art class. The camp made a rule that there would be no cannonballing in the pool so that the art wouldn't get ruined by the splashing water. Today, there is a bee attacking this kid, and she needs to jump into the water quickly. This kid cannonballs into the pool.

Ethical Decision Making

NN accuracy: 65%

Farmer John has N cows ($2 \leq N \leq 10^5$). Each cow has a breed that is either Guernsey or Holstein. As is often the case, the cows are standing in a line, numbered $1 \cdots N$ in this order.

Over the course of the day, each cow writes down a list of cows. Specifically, cow i's list contains the range of cows starting with herself (cow i) up to and including cow E_i ($i \leq E_i \leq N$).

FJ has recently discovered that each breed of cow has exactly one distinct leader. FJ does not know who the leaders are, but he knows that each leader must have a list that includes all the cows of their breed, or the other breed's leader (or both).

Help FJ count the number of pairs of cows that could be leaders. It is guaranteed that there is at least one possible pair.

Problem

Competitive Programming

NN accuracy: 28.7%

But… Is That Enough?

(i) Remove all gray spheres. How many spheres are there? (3), (ii) Take away 3 cubes. How many objects are there? (7), (iii) How many blocks must be removed to get 1 block? (2)

Complex Question Answering

NN accuracy: 50%

Abstract Reasoning

NN accuracy: 53%

IMO 2015 P3

"Let ABC be an acute triangle. Let (O) be its circumcircle, H its orthocenter, and F the foot of the altitude from A. Let M be the midpoint of BC. Let Q be the point on (O) such that QH ⊥ QA and let K be the point on (O) such that KH ⊥ KQ. Prove that the circumcircles (O_1) and (O_2) of triangles FKM and KQH are tangent to each other."

Automated Theorem Proving

NN accuracy: 0%

Neuro-Symbolic AI

Interactive Learning

NN accuracy: 71%

Scenario

Imagine that a stranger will give Hank one thousand dollars to break all the windows in his neighbor's house without his neighbor's permission. Hank carries out the stranger's request.

Imagine that there are five people who are waiting in line to use a single-occupancy bathroom at a concert venue. Someone at the back of the line needs to throw up immediately. That person skips to the front of the line instead of waiting in the back.

At a summer camp, there is a pool. Right next to the pool is a tent where the kids at the camp have art class. The camp made a rule that there would be no cannonballing in the pool so that the art wouldn't get ruined by the splashing water. Today, there is a bee attacking this kid, and she needs to jump into the water quickly. This kid cannonballs into the pool.

Ethical Decision Making

NN accuracy: 65%

Farmer John has N cows ($2 \le N \le 10^5$). Each cow has a breed that is either Guernsey or Holstein. As is often the case, the cows are standing in a line, numbered $1 \cdots N$ in this order.

Over the course of the day, each cow writes down a list of cows. Specifically, cow i's list contains the range of cows starting with herself (cow i) up to and including cow E_i ($i \le E_i \le N$).

FJ has recently discovered that each breed of cow has exactly one distinct leader. FJ does not know who the leaders are, but he knows that each leader must have a list that includes all the cows of their breed, or the other breed's leader (or both).

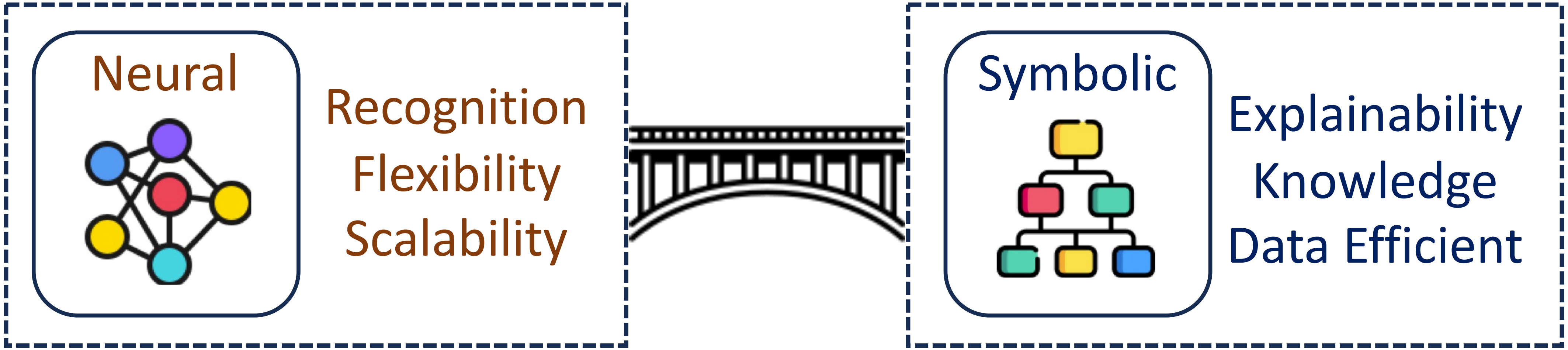
Help FJ count the number of pairs of cows that could be leaders. It is guaranteed that there is at least one possible pair.

Problem

Competitive Programming

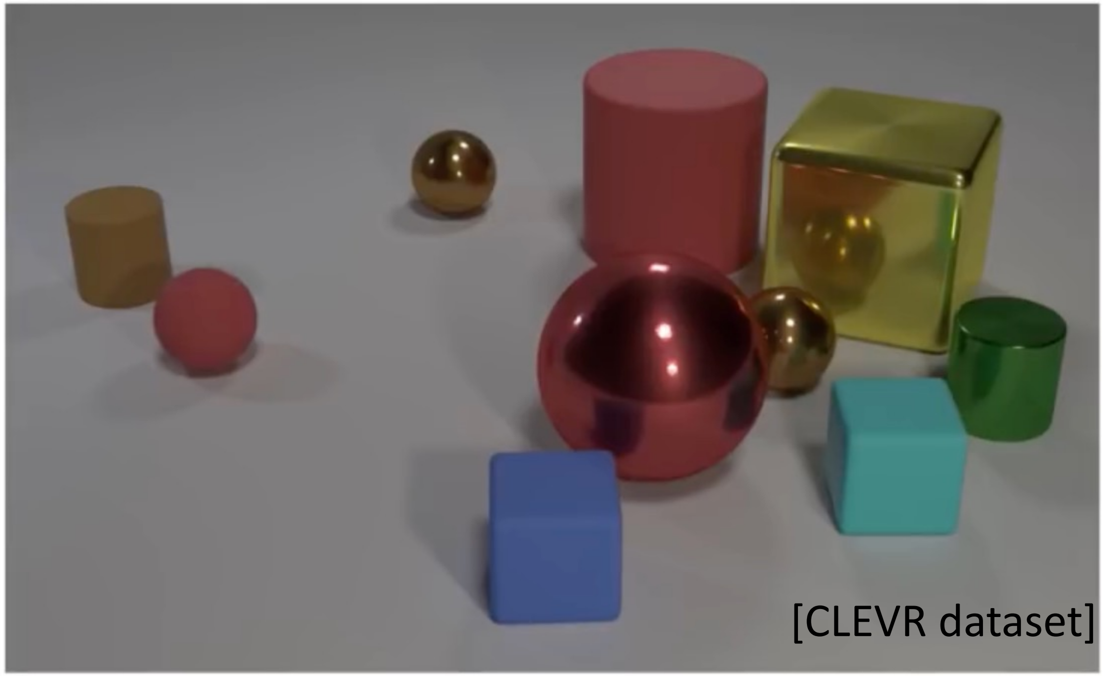
NN accuracy: 8.7%

What is Neuro-Symbolic AI?



Towards Cognitive and Trustworthy AI Systems

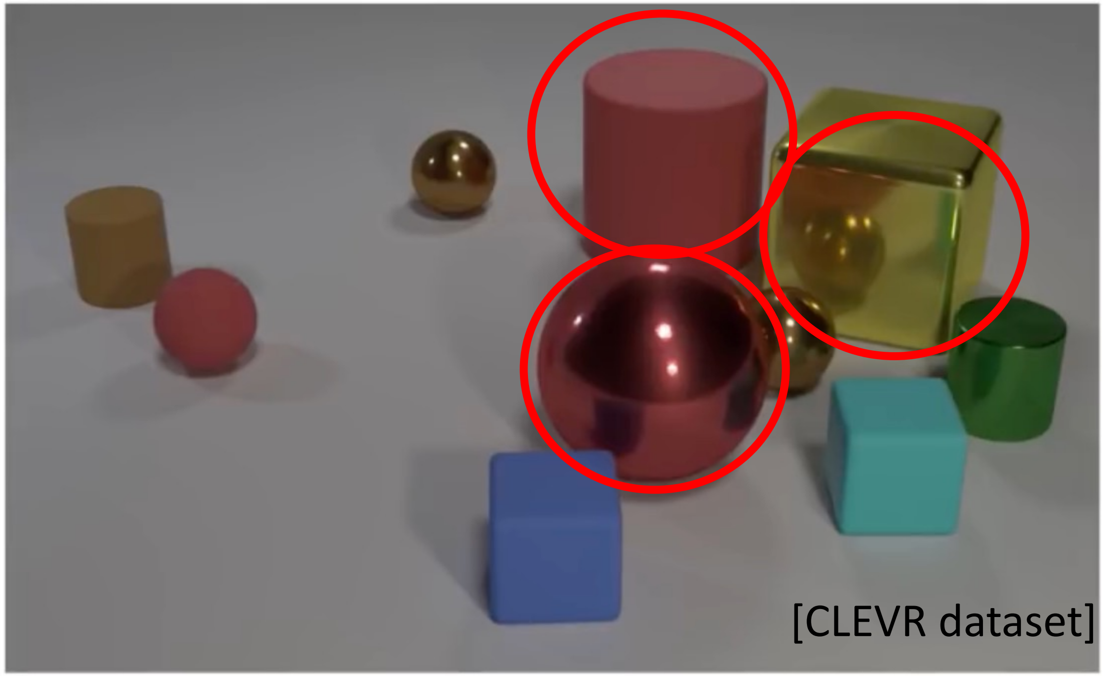
Neuro-Symbolic AI Example: Visual Reasoning



Question: *Are there an **equal number** of large things and metal spheres?*

Slide Adapted from MIT 6.S191: Neurosymbolic AI

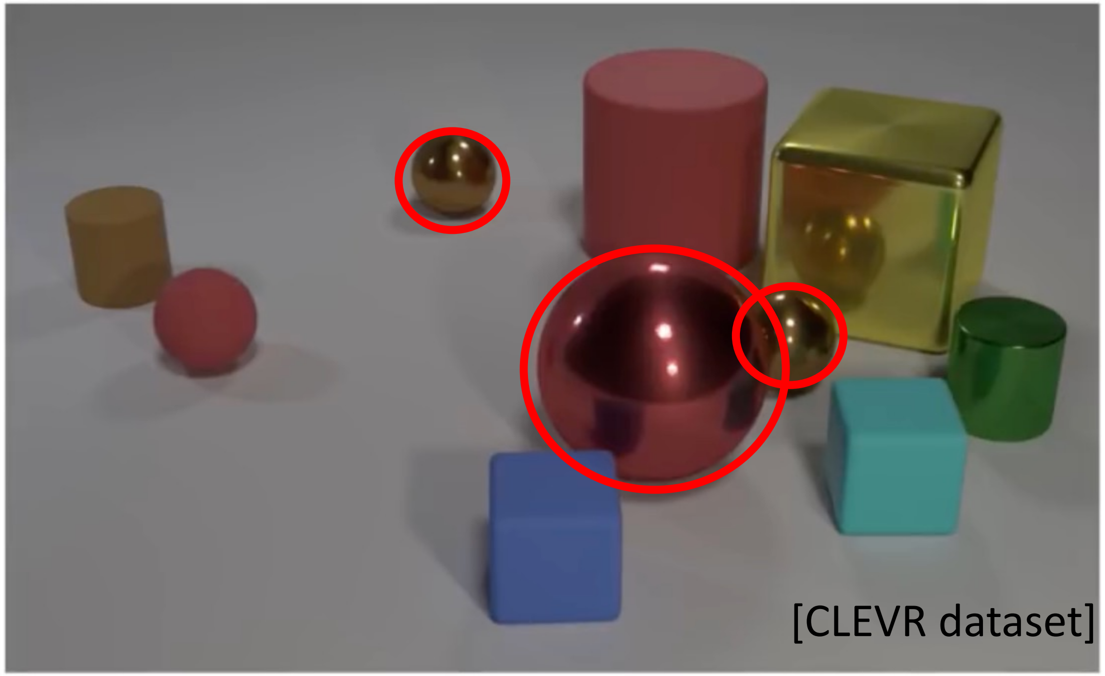
Neuro-Symbolic AI Example: Visual Reasoning



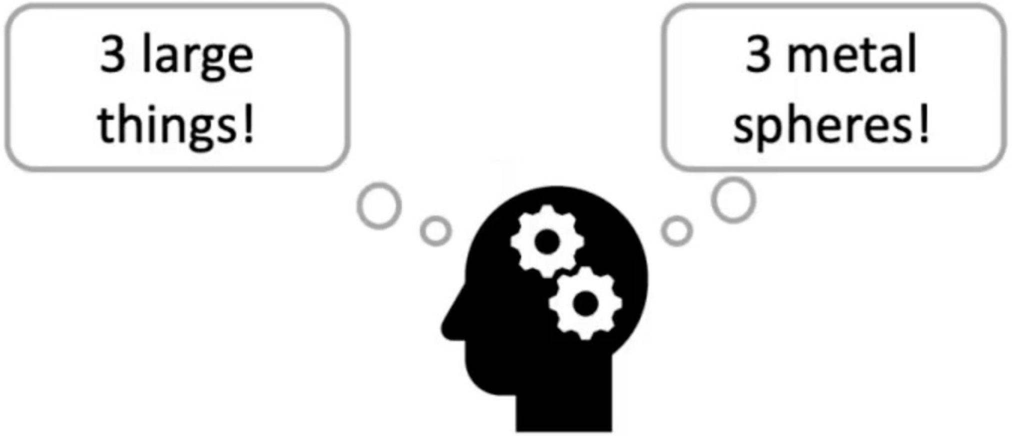
Question: *Are there an **equal number** of large things and metal spheres?*

Slide Adapted from MIT 6.S191: Neurosymbolic AI

Neuro-Symbolic AI Example: Visual Reasoning

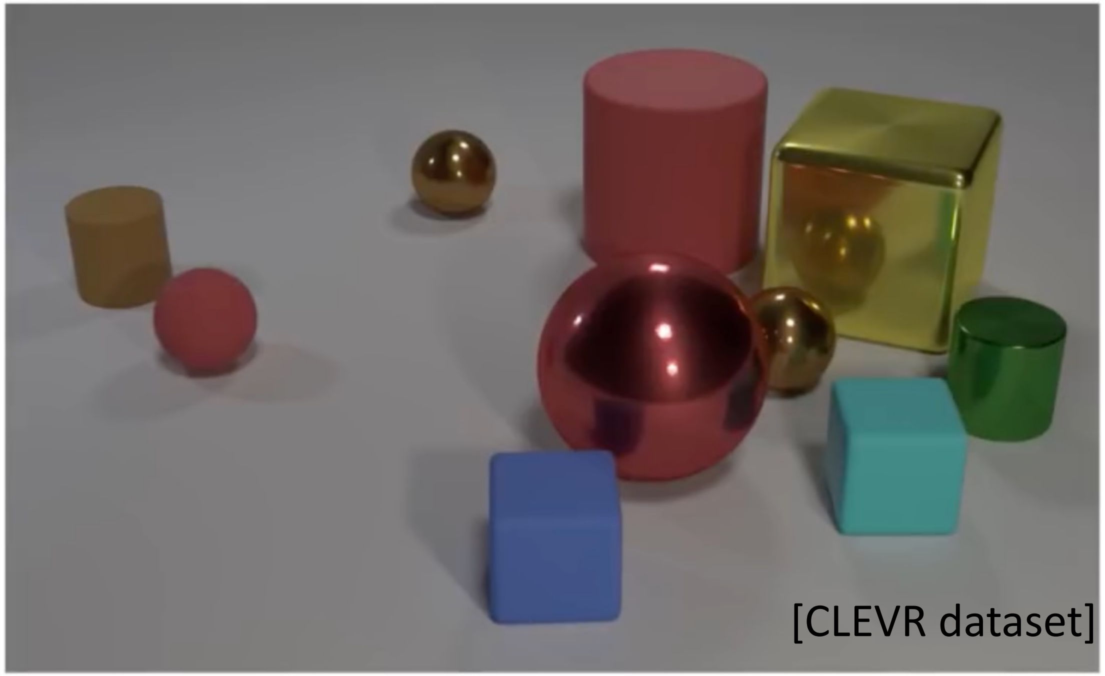


Question: *Are there an **equal number** of large things and metal spheres?*

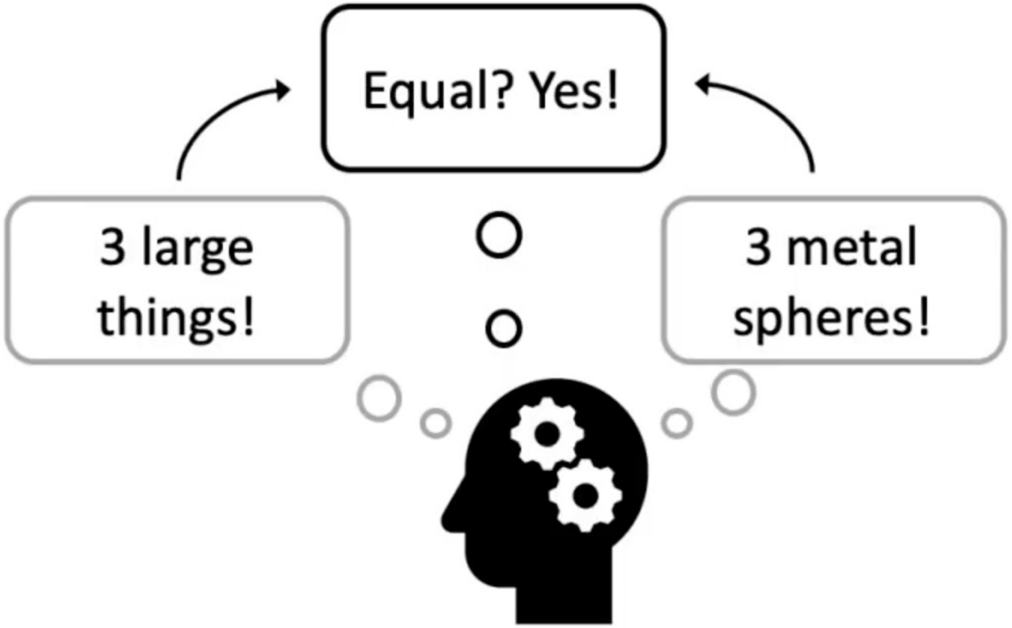


Slide Adapted from MIT 6.S191: Neurosymbolic AI

Neuro-Symbolic AI Example: Visual Reasoning

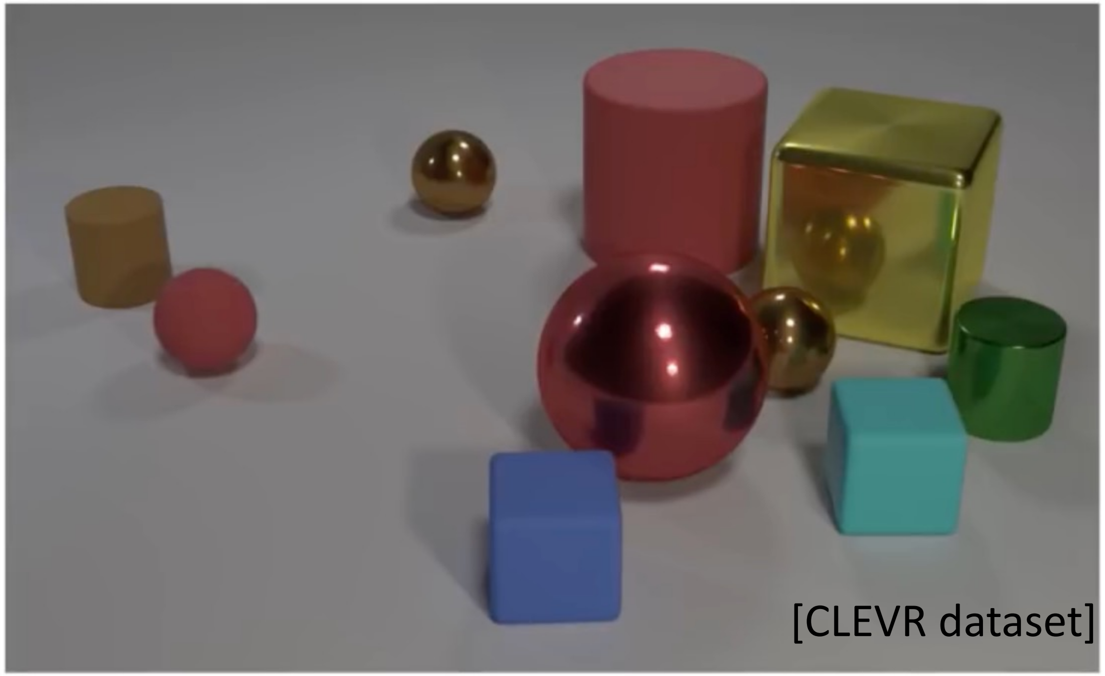


Question: *Are there an **equal number** of large things and metal spheres?*



Slide Adapted from MIT 6.S191: Neurosymbolic AI

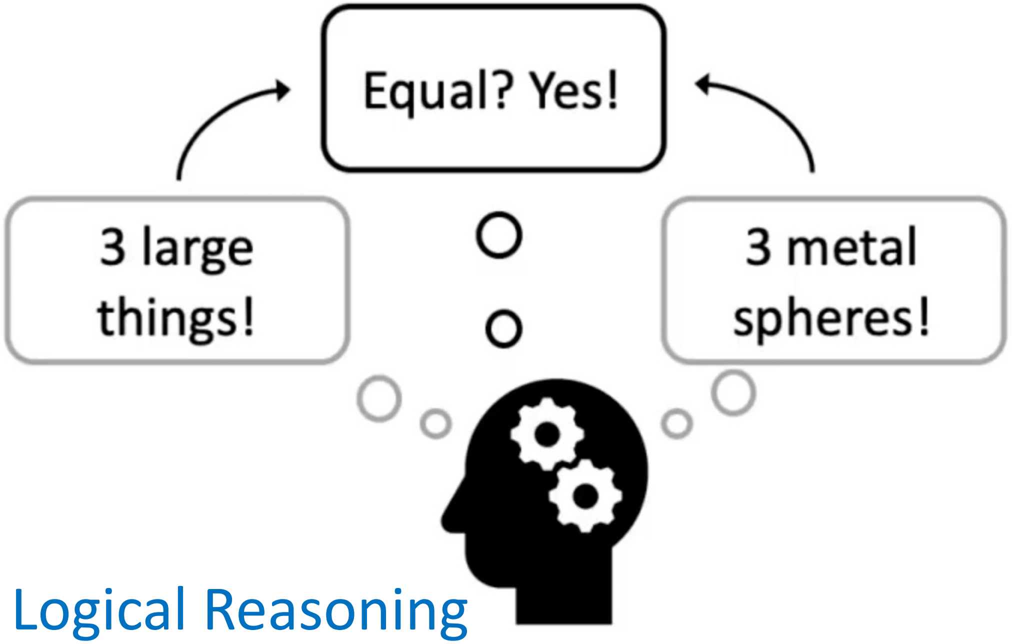
Neuro-Symbolic AI Example: Visual Reasoning



Visual Perception

Question Understanding

Question: *Are there an **equal number** of large things and metal spheres?*



Logical Reasoning

Other Examples

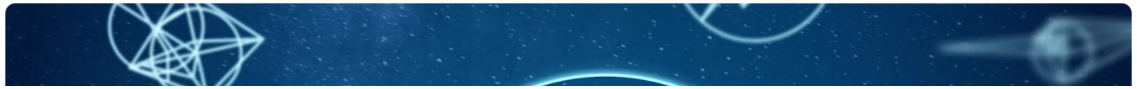
Google DeepMind

AlphaGeometry: An Olympiad-level AI system for geometry

17 JANUARY 2024

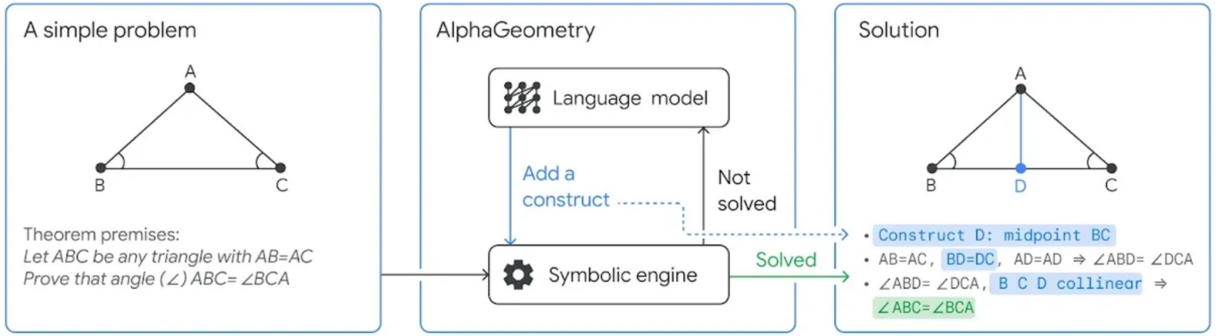
Trieu Trinh and Thang Luong

Share



AlphaGeometry adopts a neuro-symbolic approach

AlphaGeometry is a neuro-symbolic system made up of a neural language model and a symbolic deduction engine, which work together to find proofs for complex geometry theorems. Akin to the idea of "[thinking, fast and slow]", one system provides fast, "intuitive" ideas, and the other, more deliberate, rational decision-making.

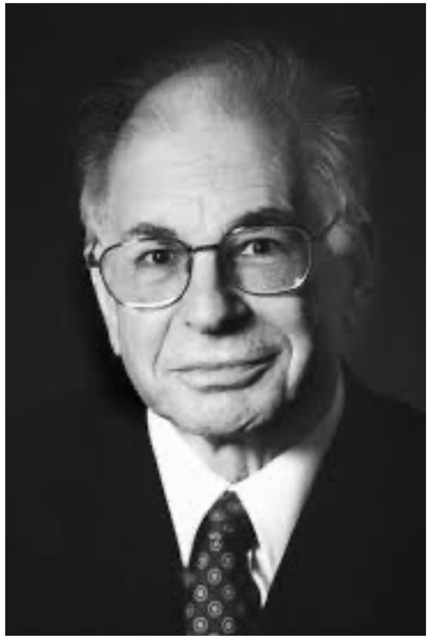


LLM: construct auxiliary points and lines
Symbolic: deductive reasoning

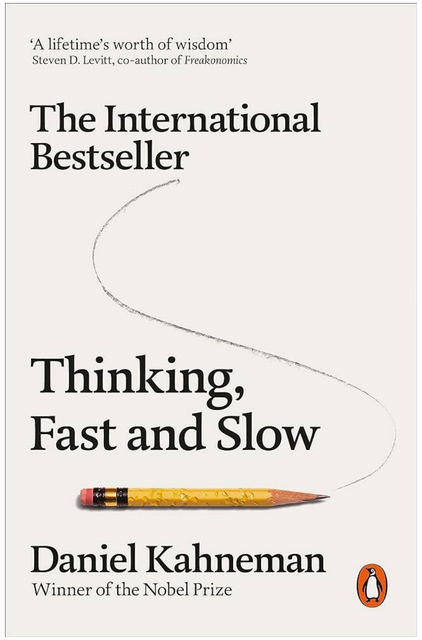
Eval on 30 Int. Math Olympics (IMO) problems:
- GPT-4: 8/30
- AlphaGeometry (Neuro-Symbolic): 25/30
- Human Gold Medalist: 26/30

Trinh et al, "Solving Olympiad Geometry without Human Demonstrations", Nature 2024

Relationship to Human Minds



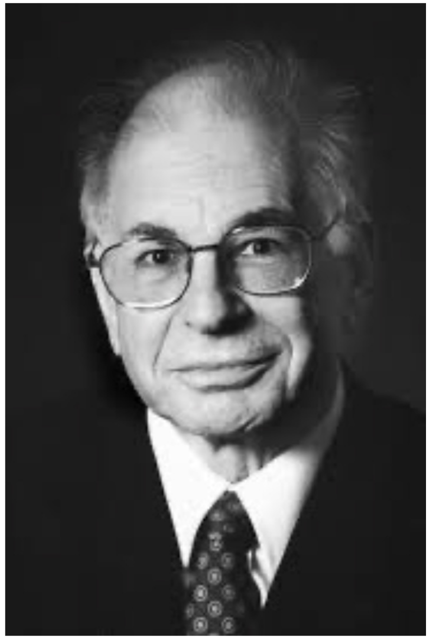
Daniel Kahneman (1934-2024)



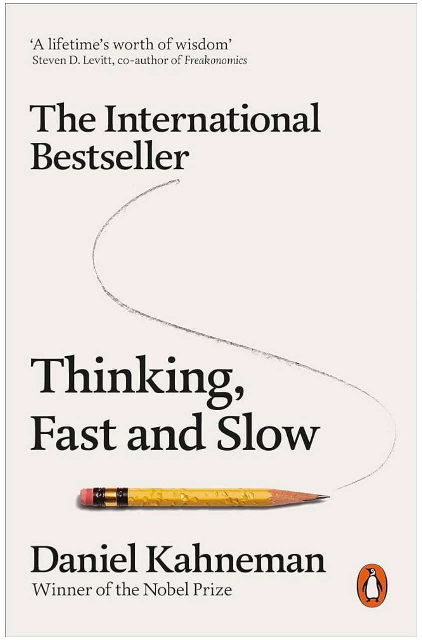
AlphaGeometry adopts a neuro-symbolic approach

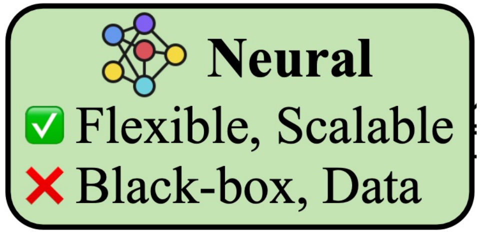
AlphaGeometry is a neuro-symbolic system made up of a neural language model and a symbolic deduction engine, which work together to find proofs for complex geometry theorems. Akin to the idea of "thinking, fast and slow", one system provides fast, "intuitive" ideas, and the other, more deliberate, rational decision-making.

Relationship to Human Minds



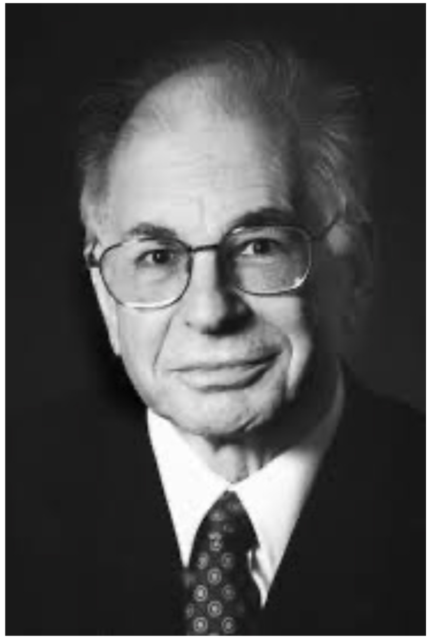
Daniel Kahneman
(1934-2024)



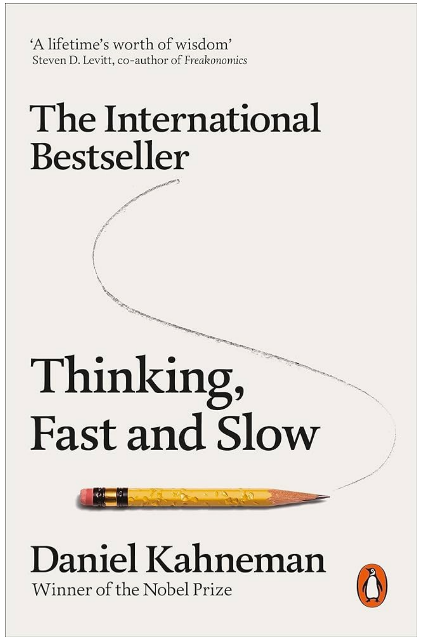


*System 1: thinking **fast***
(intuitive perception)

Relationship to Human Minds



Daniel Kahneman
(1934-2024)



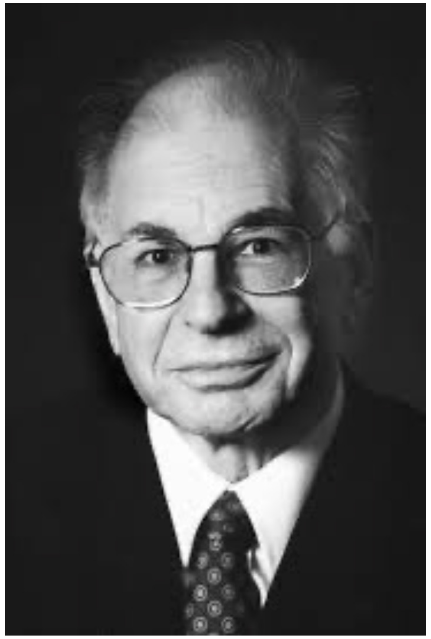
Neural
✅ Flexible, Scalable
❌ Black-box, Data

*System 1: thinking **fast***
(intuitive perception)

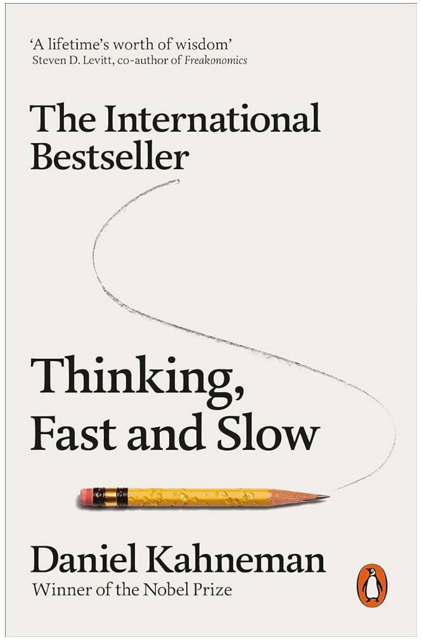
Symbolic
✅ Reasoning, Transparent
❌ Scalable, Learnable

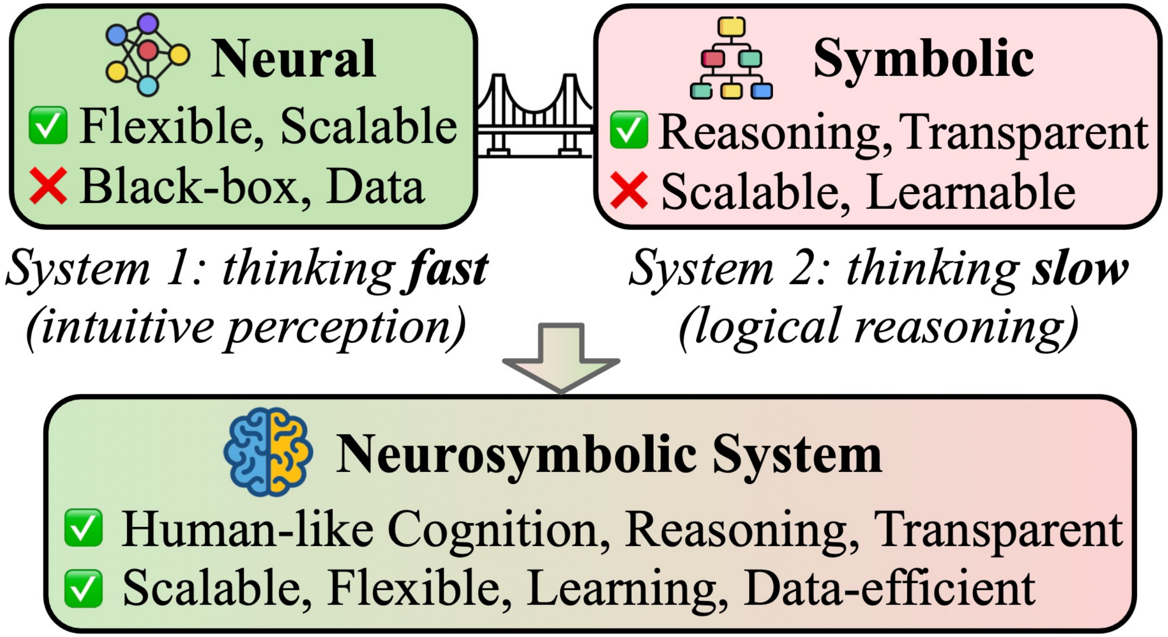
*System 2: thinking **slow***
(logical reasoning)

Relationship to Human Minds

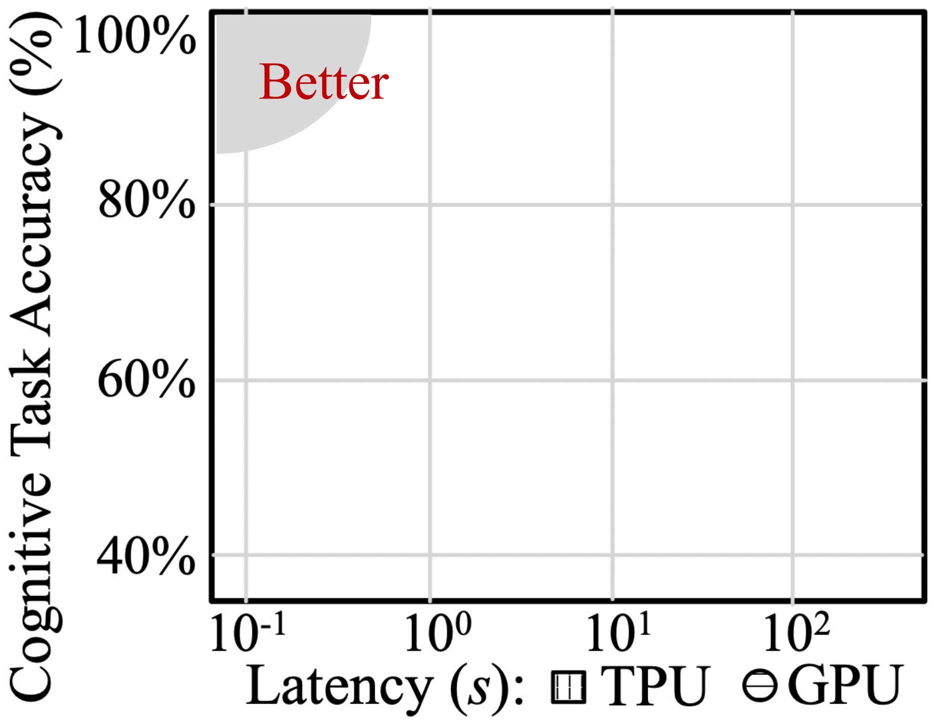


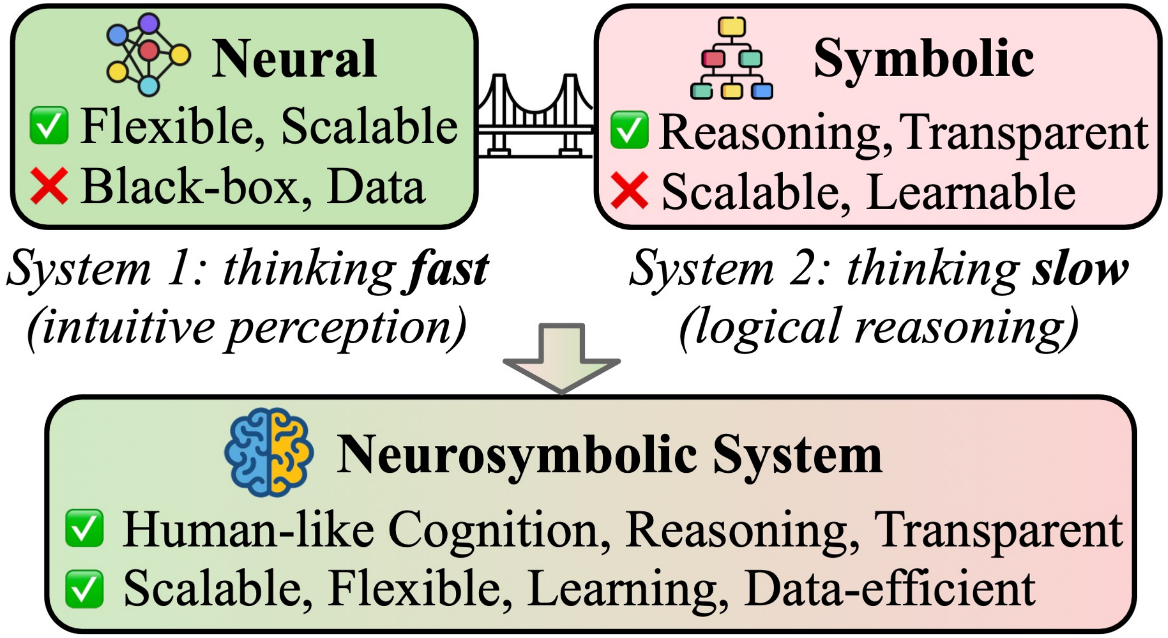
Daniel Kahneman
(1934-2024)



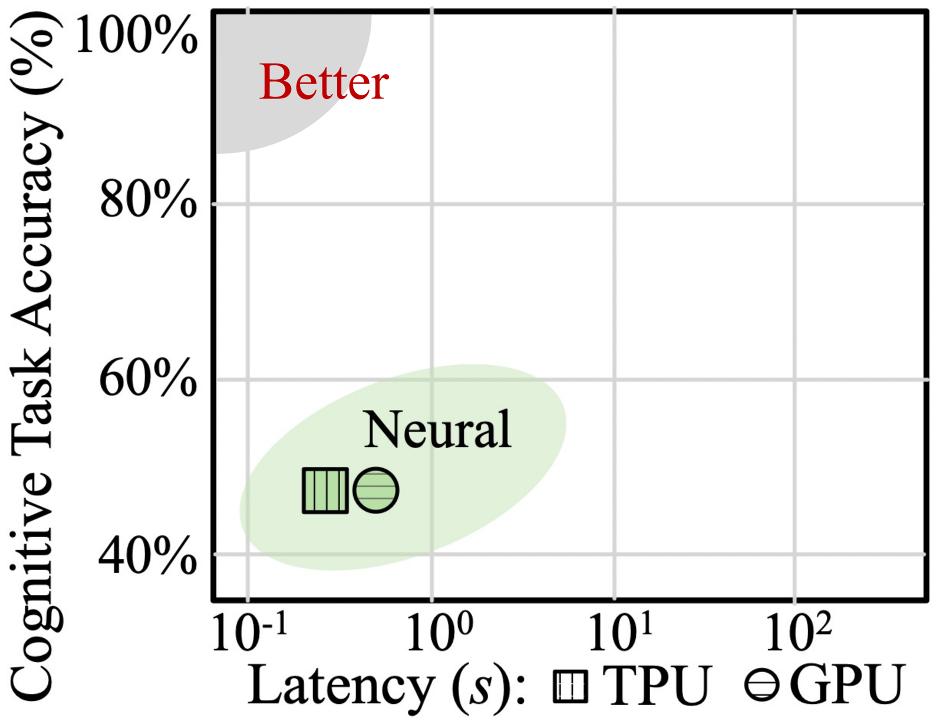


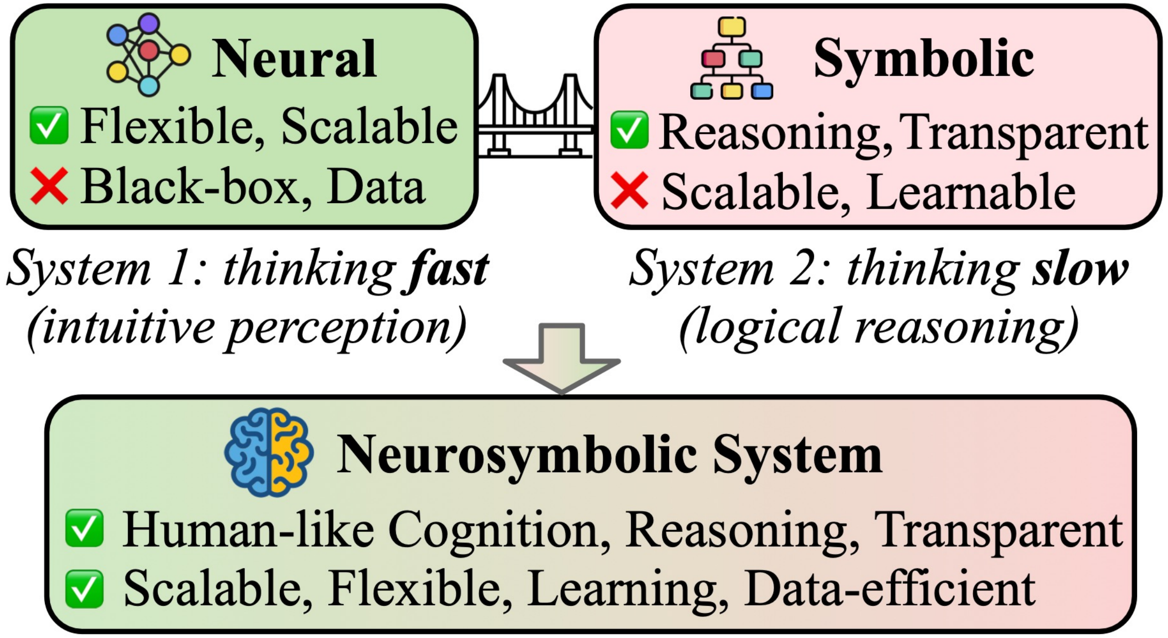
However.. From Computing Perspective



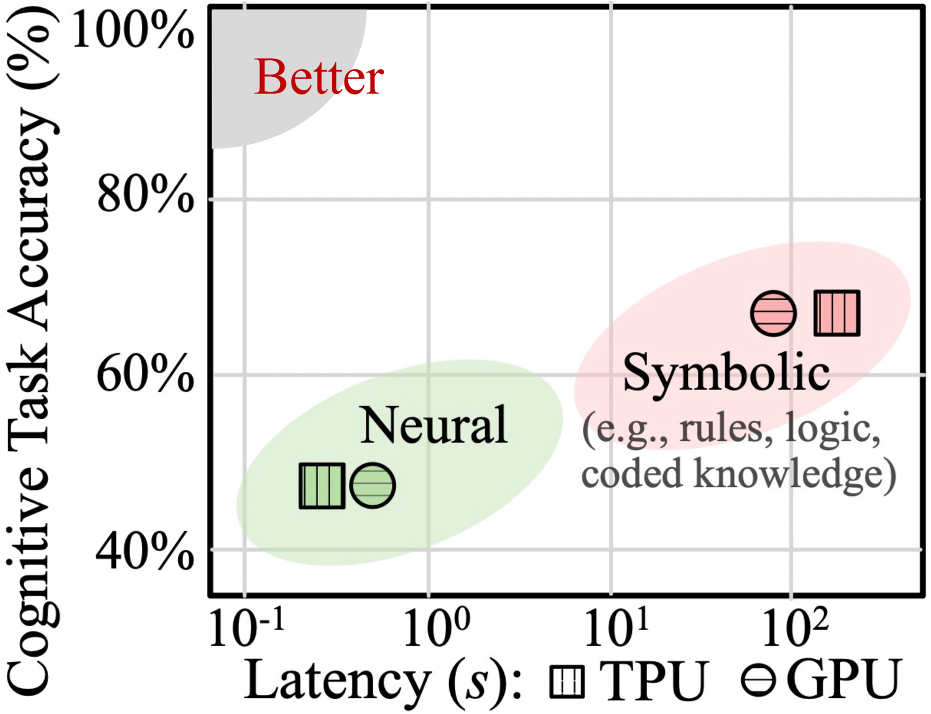


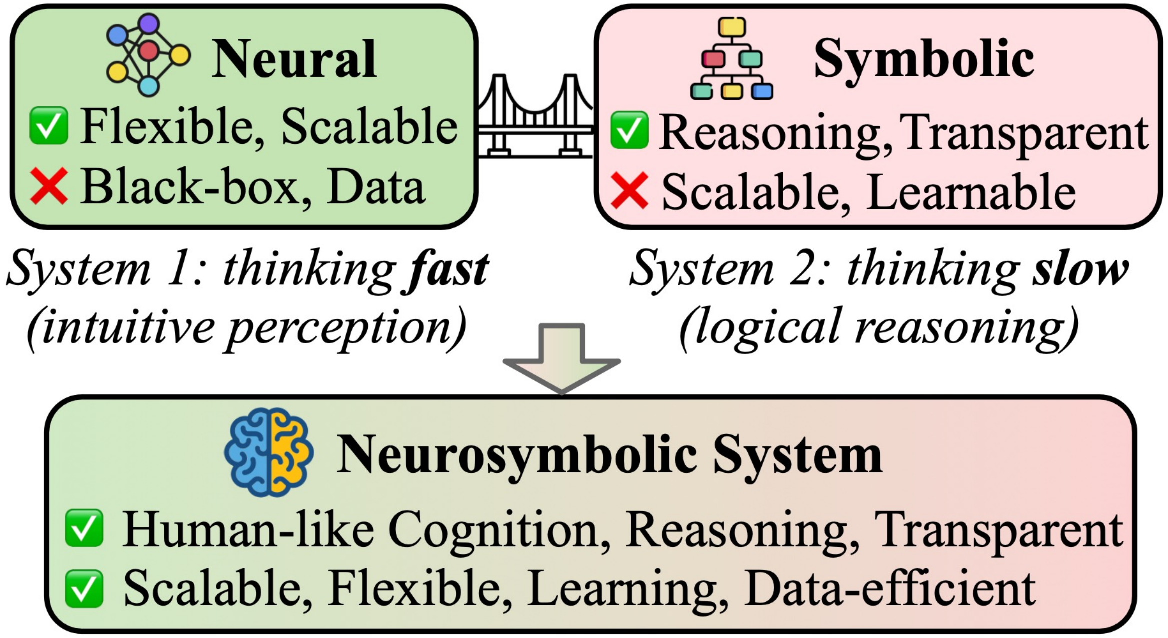
However.. From Computing Perspective



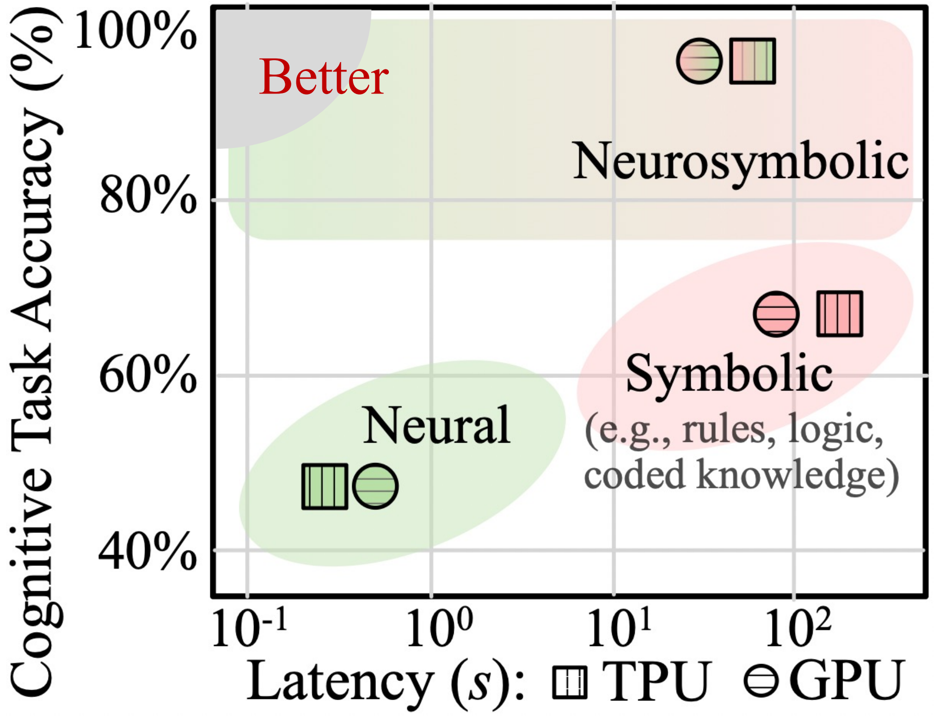


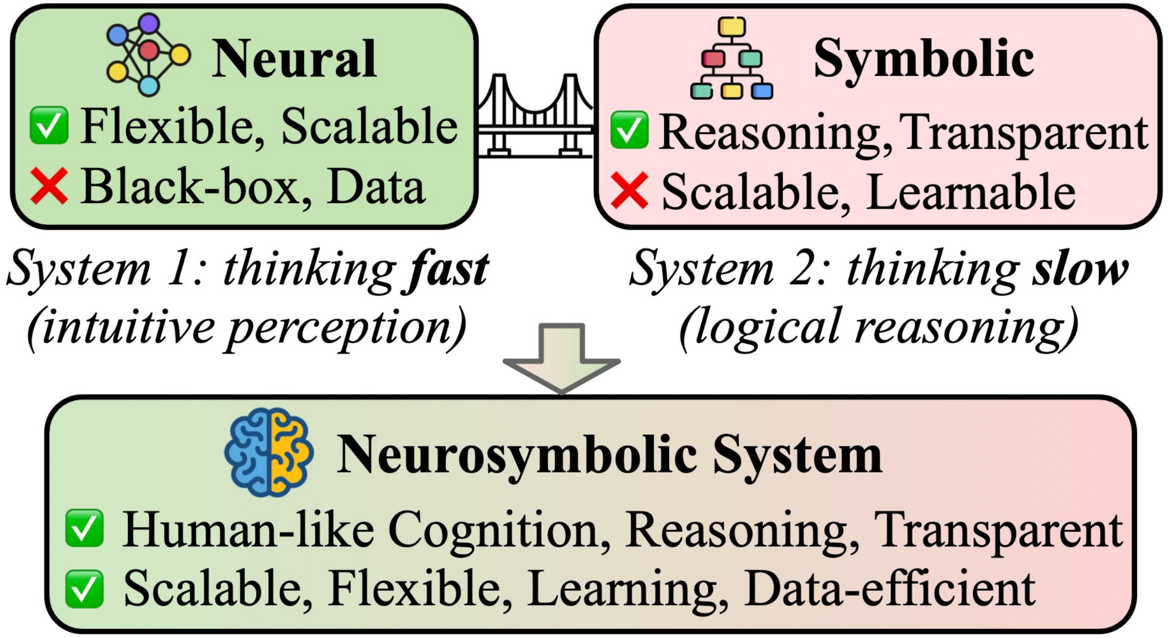
However.. From Computing Perspective



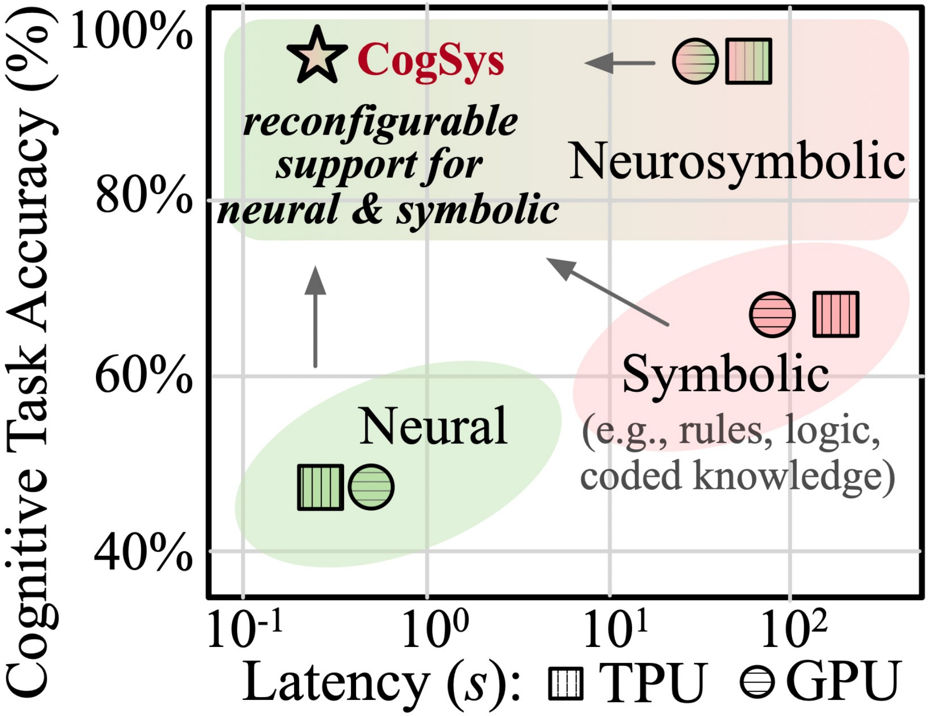


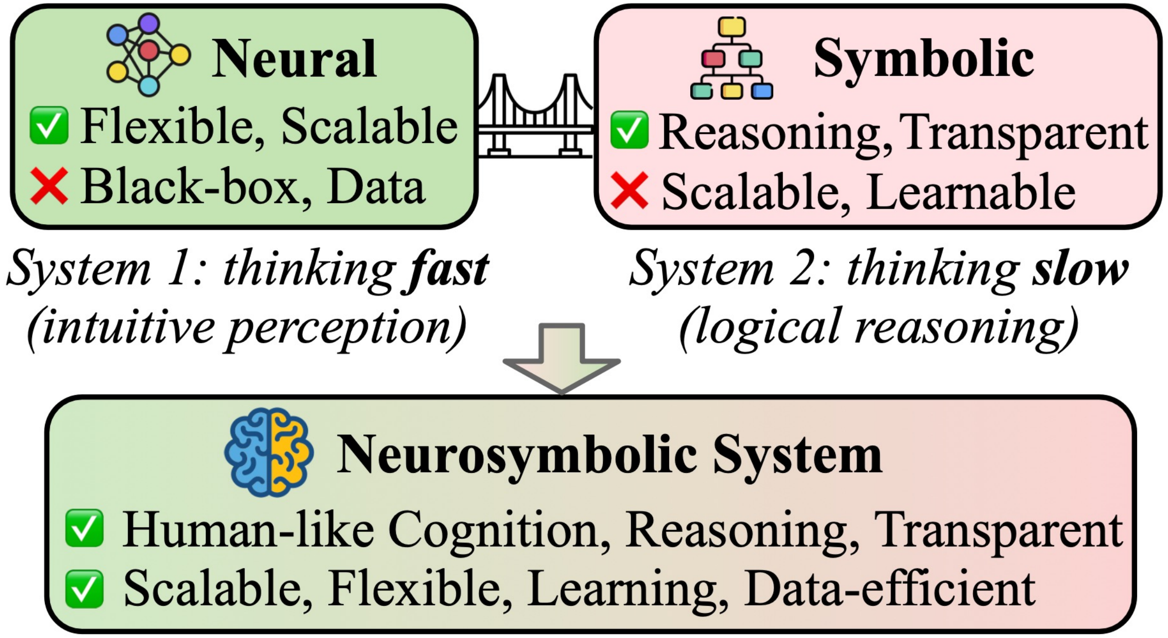
However.. From Computing Perspective





However.. From Computing Perspective



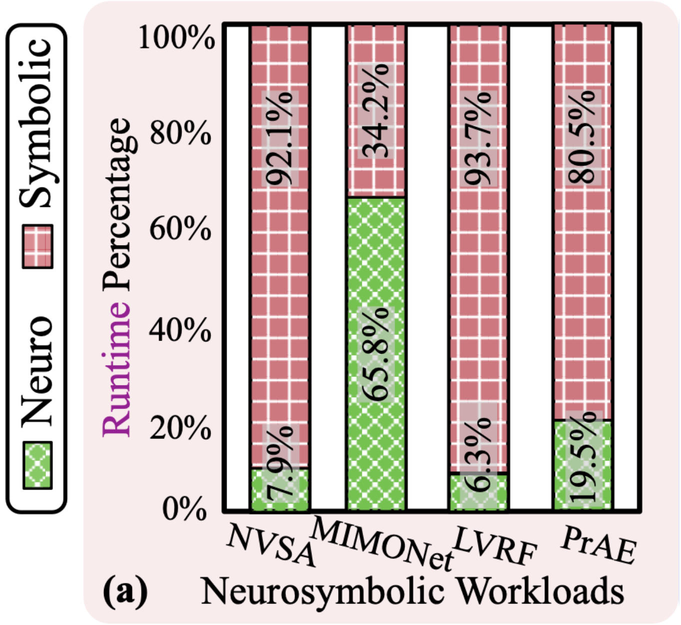


Research Question:

What’s the **system implications** of neuro-symbolic workloads?

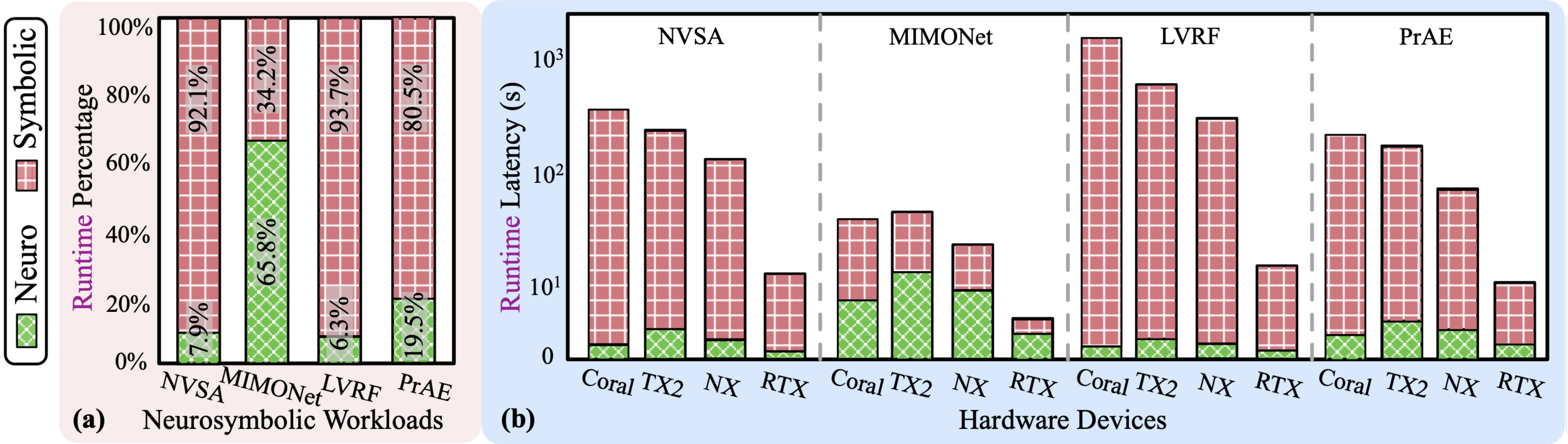
Why neuro-symbolic workloads are **inefficient** on off-the-shelf hardware?

Workload Profiling – Runtime



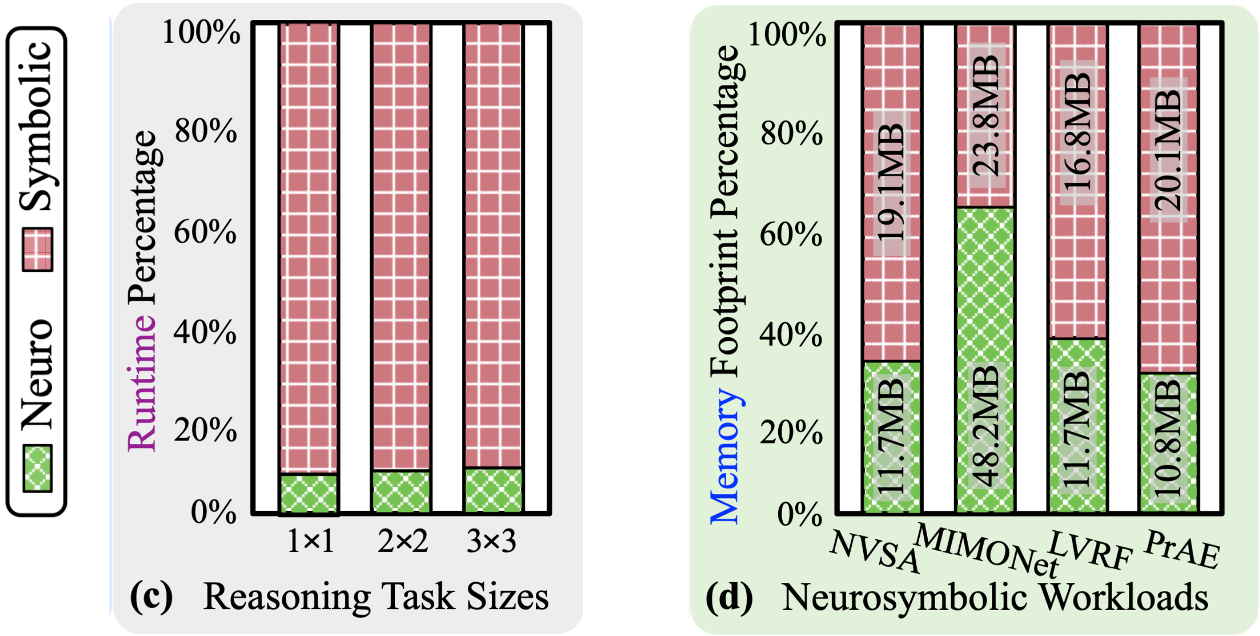
Neuro-symbolic workload exhibits **high latency** compared to neural models;

Workload Profiling – Runtime



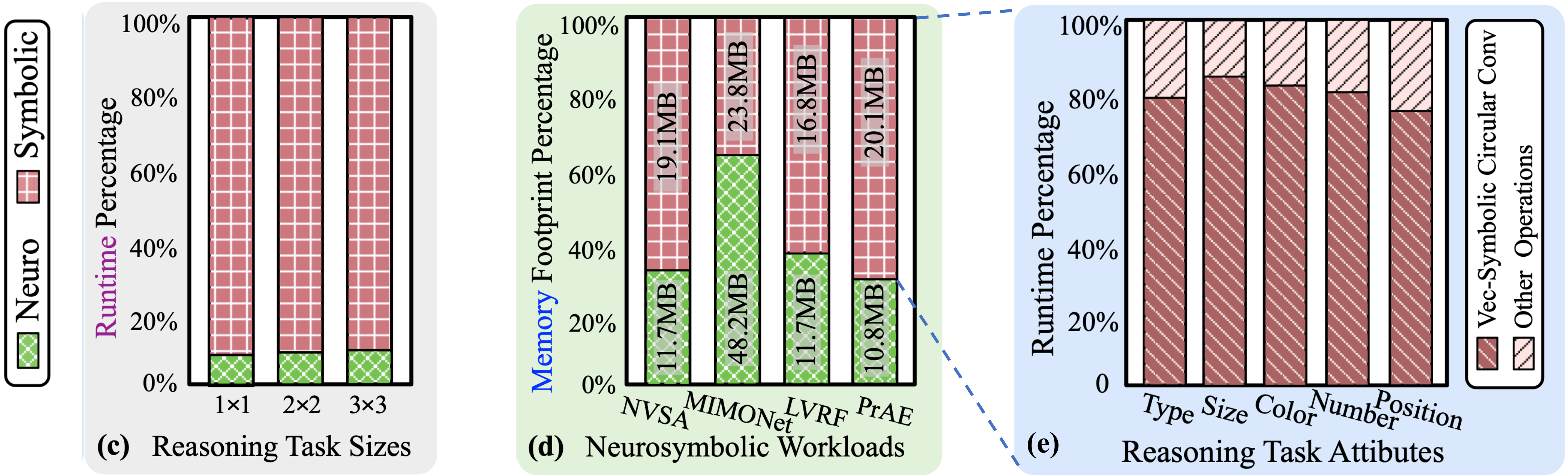
Neuro-symbolic workload exhibits **high latency** compared to neural models;
Symbolic component is executed **inefficiently** across off-the-shelf CPU/GPUs

Workload Profiling – Memory & Operator



Symbolic components exhibit **large memory footprint**;

Workload Profiling – Memory & Operator



Symbolic components exhibit **large memory footprint**;
Symbolic operations are dominated by **vector-symbolic circular convolutions**

Workload Profiling – Kernel Behavior

	Neuro Kernel		Symbolic Kernel	
	segmm_nn	relu_nn	vectorized	elementwise
Runtime Percentage (%)				
Compute Throughput (%)				
ALU Utilization (%)				
L1 Cache Hit Rate (%)				
L2 Cache Hit Rate (%)				
L1 Cache Throughput (%)				
L2 Cache Throughput (%)				
DRAM BW Utilization (%)				

Why system Inefficiency?

Workload Profiling – Kernel Behavior

	Neuro Kernel		Symbolic Kernel	
	segmm_nn	relu_nn	vectorized	elementwise
Runtime Percentage (%)	18.2	10.4	37.5	12.4
Compute Throughput (%)	**95.1**	**92.9**	3.0	2.3
ALU Utilization (%)	**90.1**	**48.3**	**5.9**	**4.5**
L1 Cache Hit Rate (%)				
L2 Cache Hit Rate (%)				
L1 Cache Throughput (%)				
L2 Cache Throughput (%)				
DRAM BW Utilization (%)				

Symbolic exhibits low ALU utilization,

Workload Profiling – Kernel Behavior

	Neuro Kernel		Symbolic Kernel	
	segmm_nn	relu_nn	vectorized	elementwise
Runtime Percentage (%)	18.2	10.4	37.5	12.4
Compute Throughput (%)	**95.1**	**92.9**	3.0	2.3
ALU Utilization (%)	**90.1**	**48.3**	**5.9**	**4.5**
L1 Cache Hit Rate (%)	1.6	**51.6**	**29.5**	**33.3**
L2 Cache Hit Rate (%)	**86.8**	**65.5**	**48.6**	**34.3**
L1 Cache Throughput (%)				
L2 Cache Throughput (%)				
DRAM BW Utilization (%)				

Symbolic exhibits low ALU utilization, low cache hit rate,

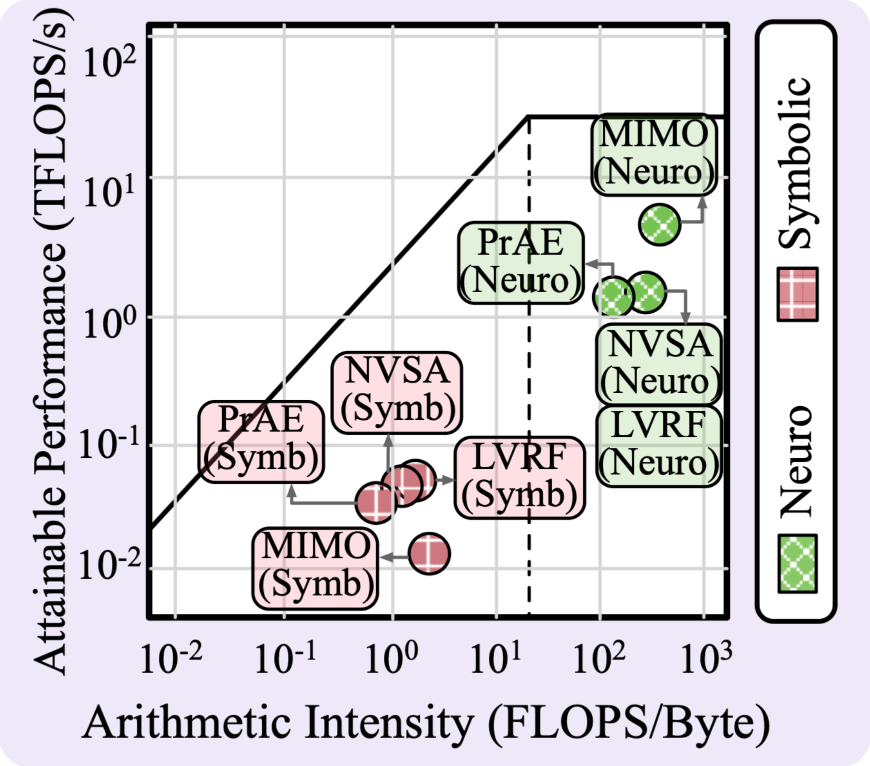
Workload Profiling – Kernel Behavior

	Neuro Kernel		Symbolic Kernel	
	segmm_nn	relu_nn	vectorized	elementwise
Runtime Percentage (%)	18.2	10.4	37.5	12.4
Compute Throughput (%)	**95.1**	**92.9**	3.0	2.3
ALU Utilization (%)	**90.1**	**48.3**	**5.9**	**4.5**
L1 Cache Hit Rate (%)	1.6	**51.6**	**29.5**	**33.3**
L2 Cache Hit Rate (%)	**86.8**	**65.5**	**48.6**	**34.3**
L1 Cache Throughput (%)	79.7	82.6	28.4	10.8
L2 Cache Throughput (%)	19.2	17.5	29.8	22.8
DRAM BW Utilization (%)	14.9	24.2	**90.9**	**78.4**

Symbolic exhibits low ALU utilization, low cache hit rate, massive data transfer, low data reuse, resulting in hardware underutilization and inefficiency

Workload Profiling – Roofline Analysis

	Neuro Kernel		Symbolic Kernel	
	segmm_nn	relu_nn	vectorized	elementwise
Runtime Percentage (%)	18.2	10.4	37.5	12.4
Compute Throughput (%)	95.1	92.9	3.0	2.3
ALU Utilization (%)	90.1	48.3	5.9	4.5
L1 Cache Hit Rate (%)	1.6	51.6	29.5	33.3
L2 Cache Hit Rate (%)	86.8	65.5	48.6	34.3
L1 Cache Throughput (%)	79.7	82.6	28.4	10.8
L2 Cache Throughput (%)	19.2	17.5	29.8	22.8
DRAM BW Utilization (%)	14.9	24.2	90.9	78.4



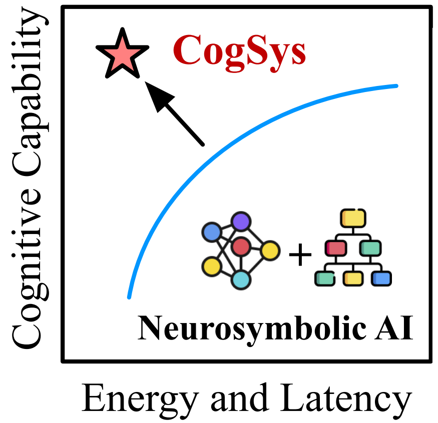
Symbolic exhibits low ALU utilization, low cache hit rate, massive data transfer, low data reuse, resulting in hardware underutilization and inefficiency

Research Question:

How to enhance the **efficiency and scalability** of neuro-symbolic systems?

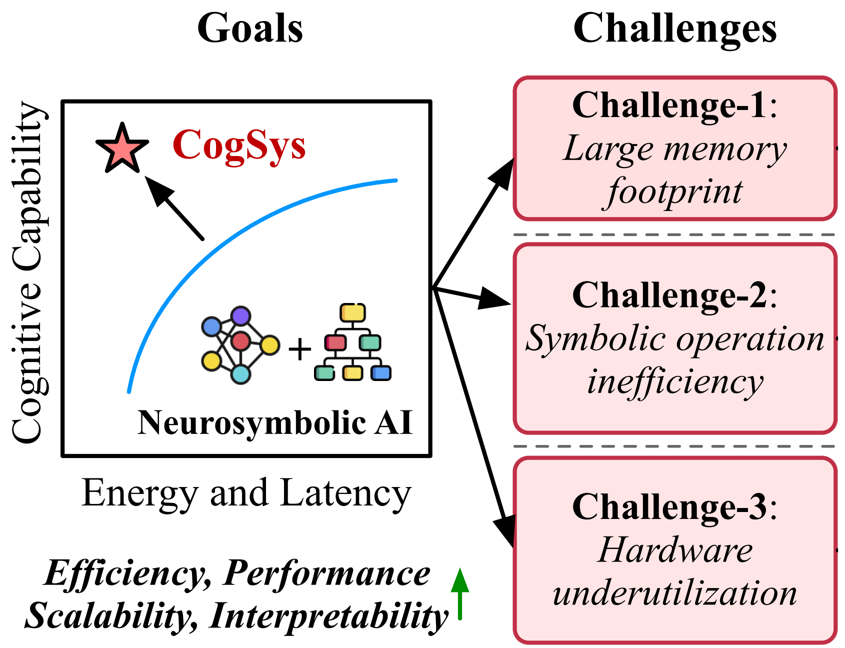
Our Methodology

Goals

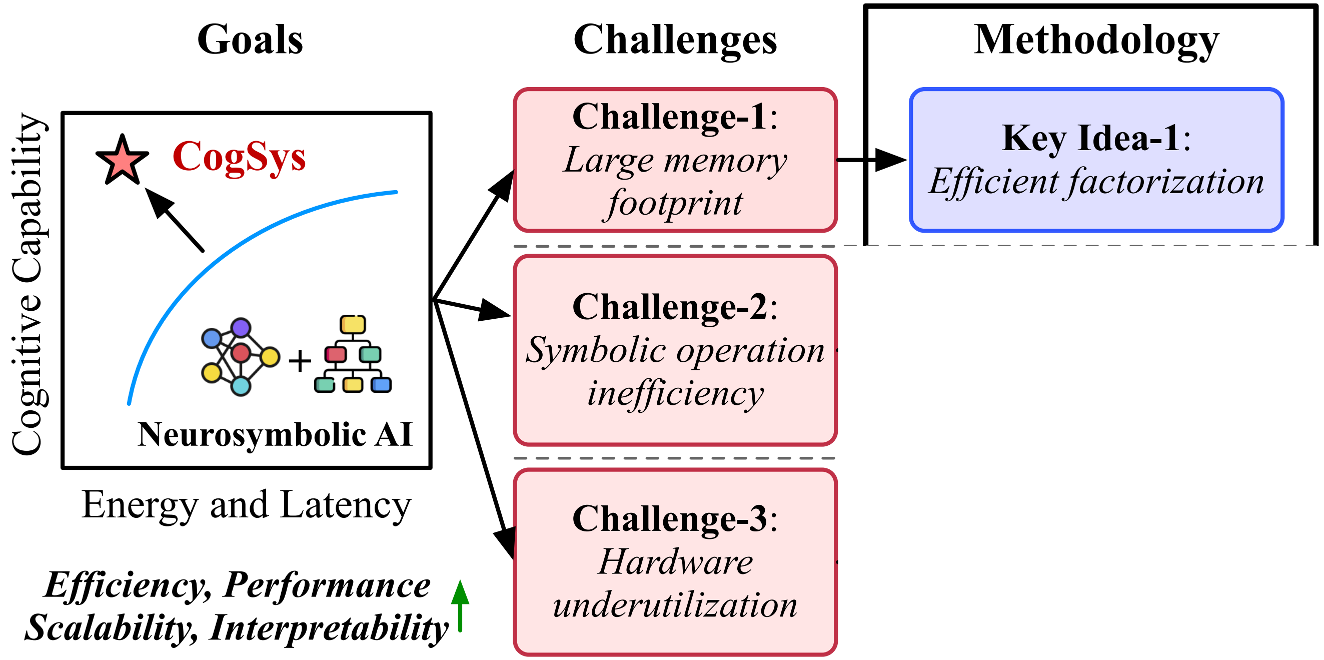


Efficiency, Performance
Scalability, Interpretability ↑

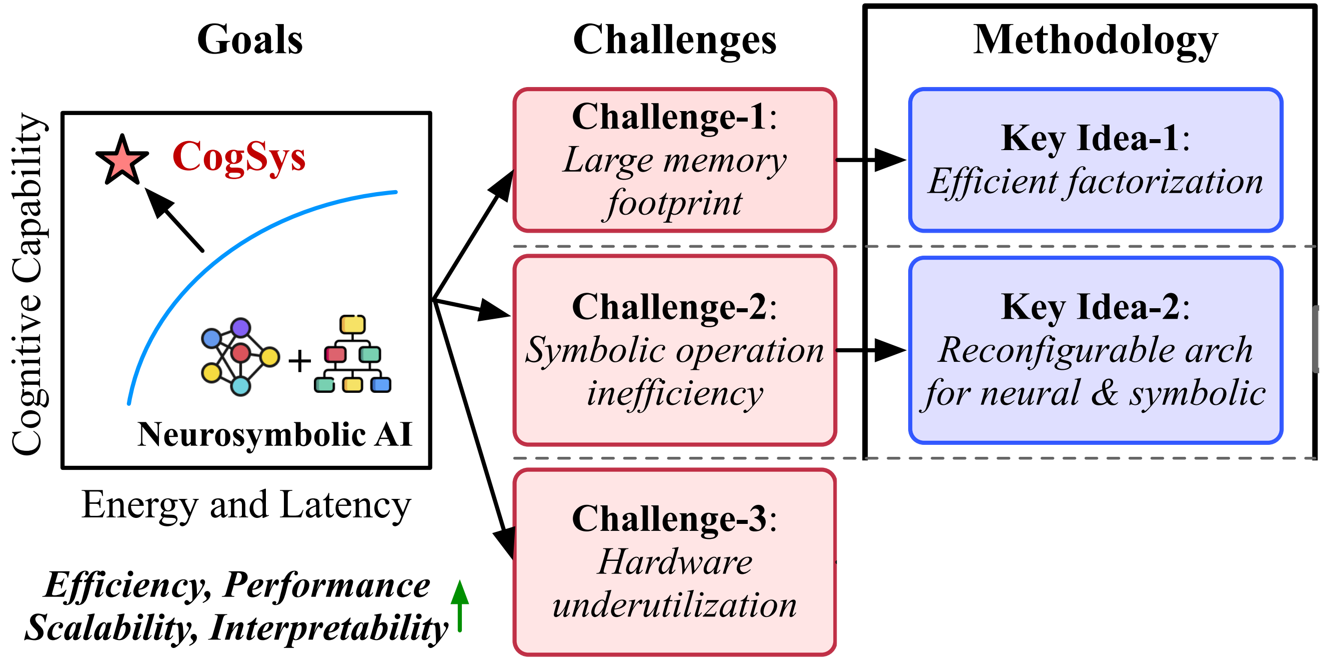
Our Methodology



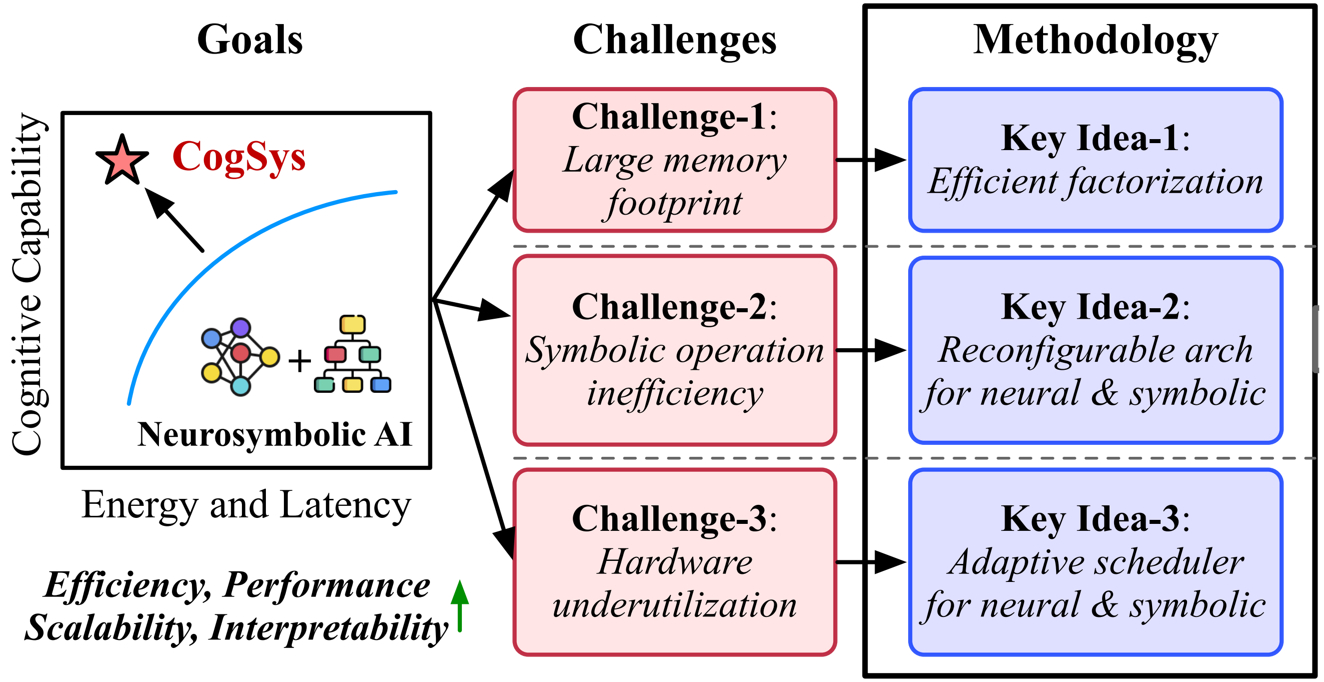
Our Methodology



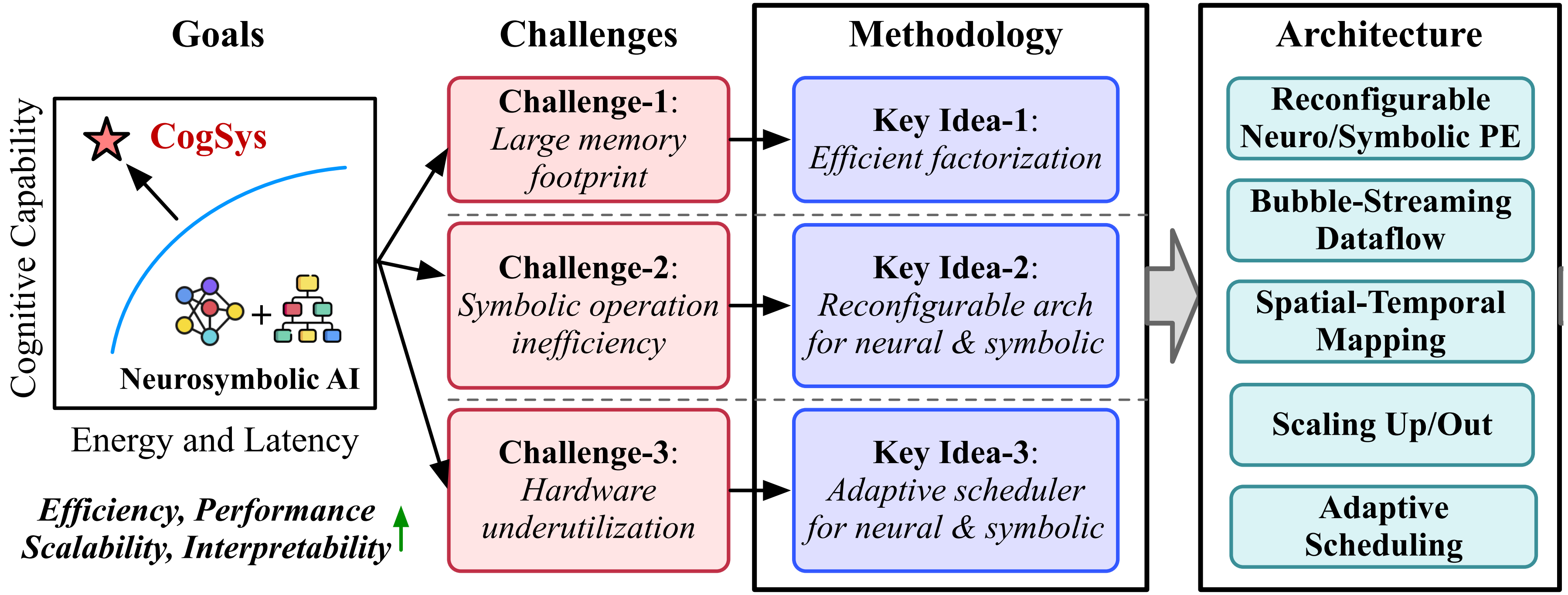
Our Methodology



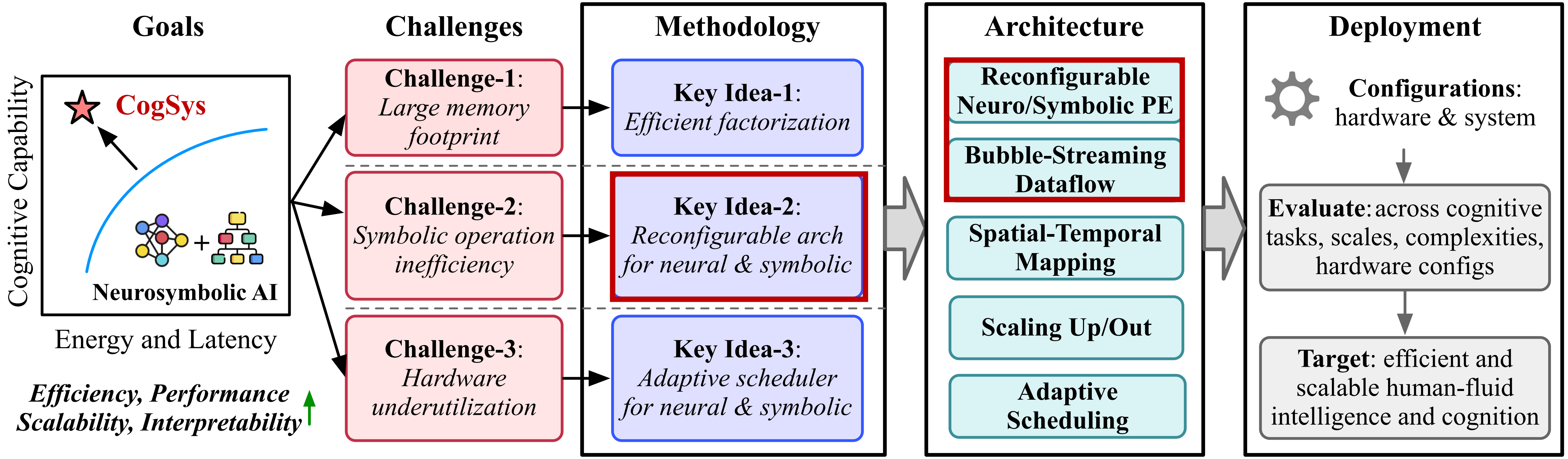
Our Methodology



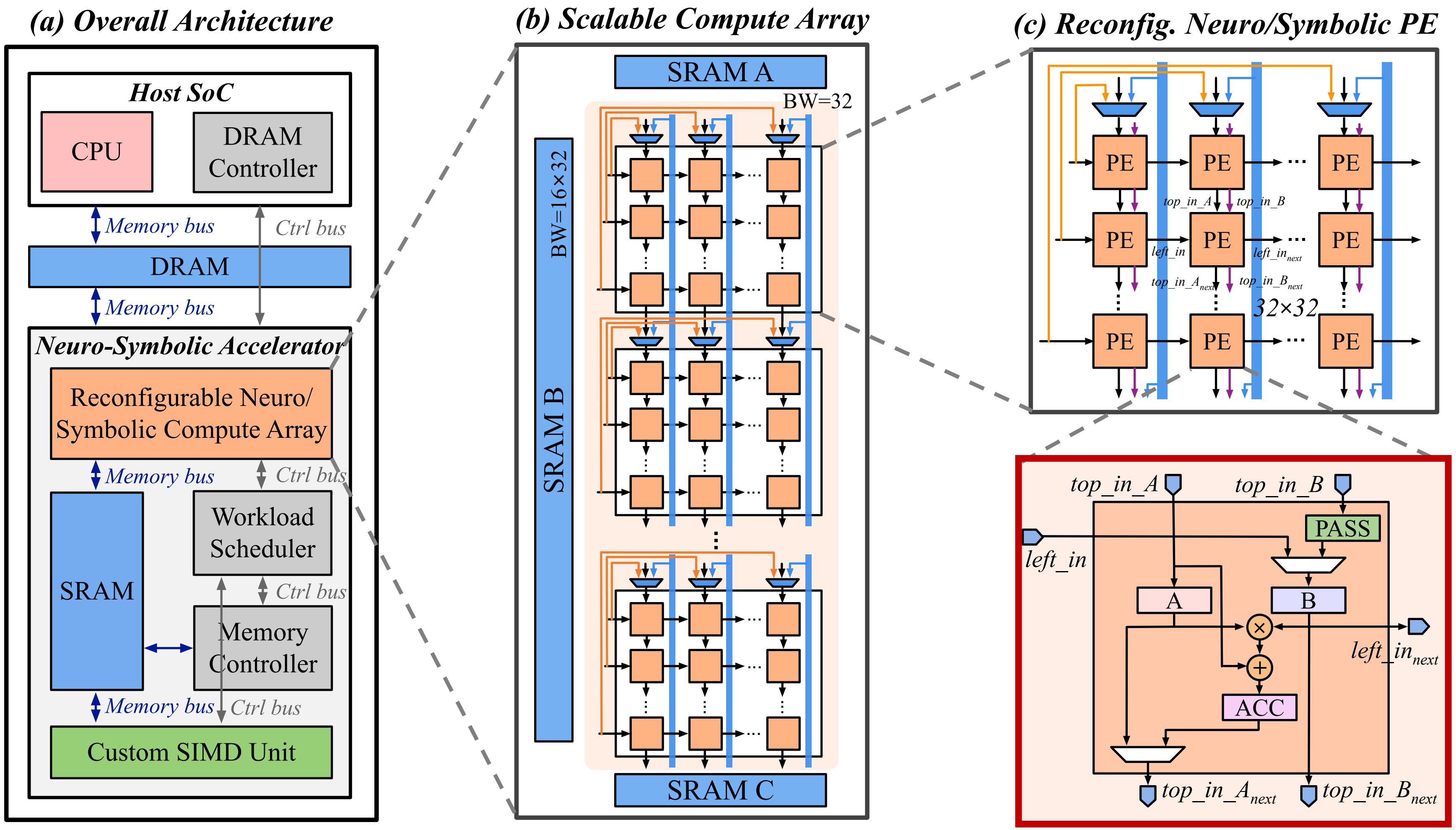
Our Methodology



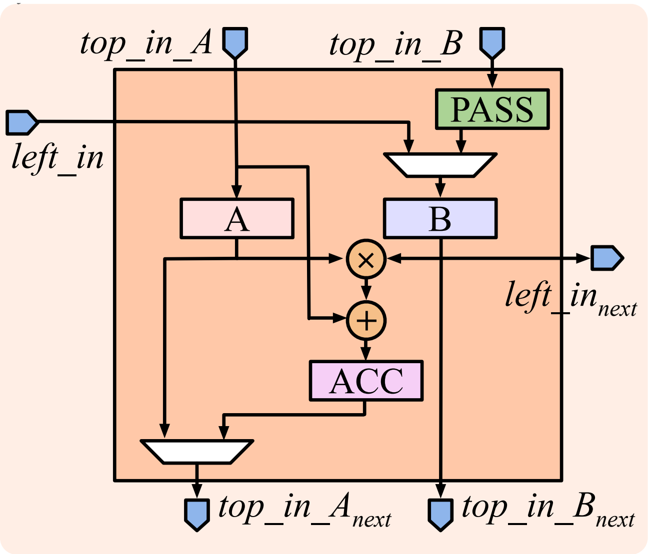
Our Methodology



Hardware Architecture Overview



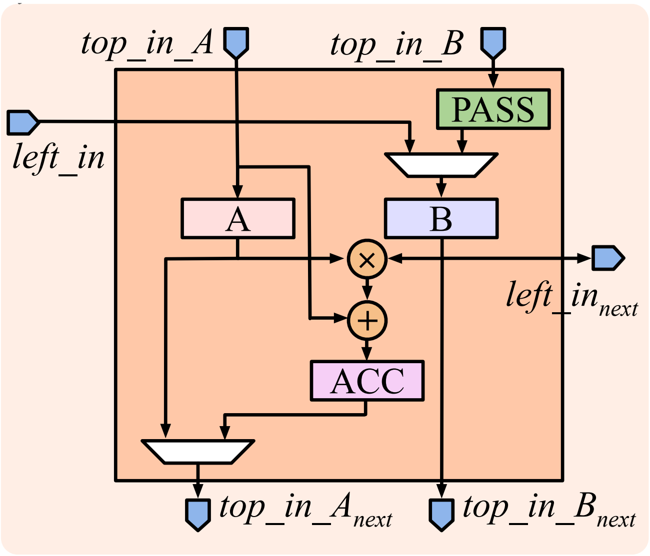
Reconfigurable Neuro/Symbolic PE



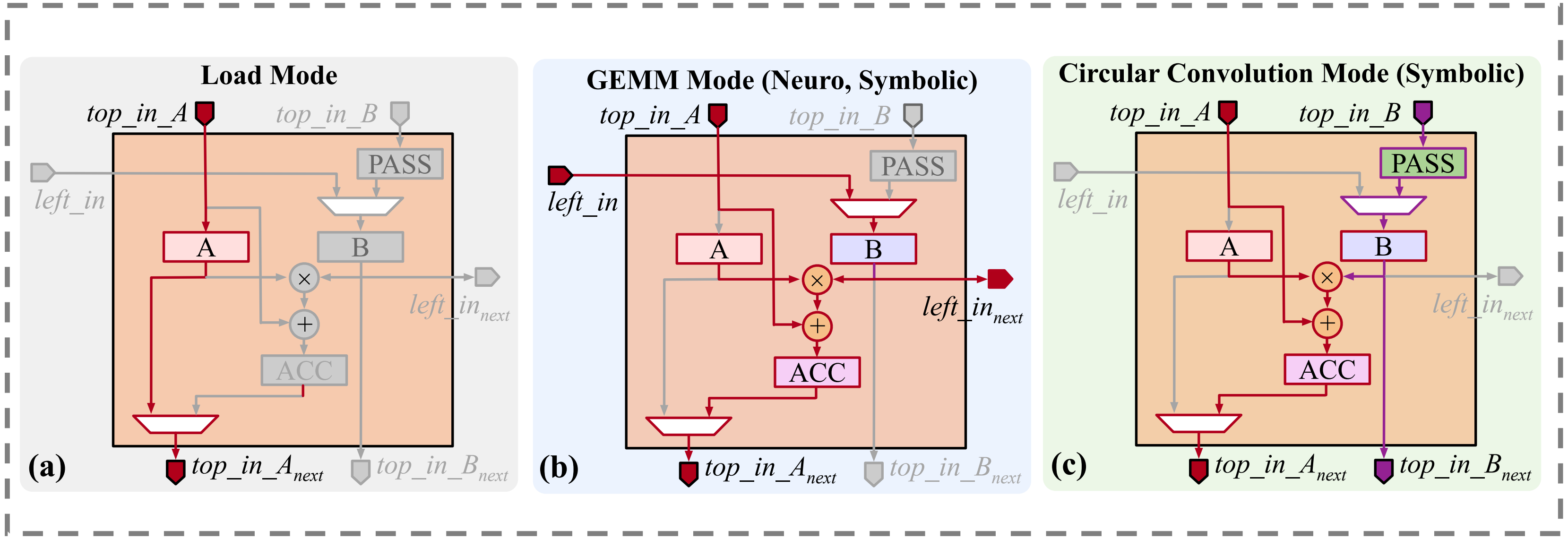
Micro-architecture of reconfigurable neuro/symbolic PE

Reconfigurable neuro/symbolic PE incurs **low area overhead** based on systolic array PE;

Reconfigurable Neuro/Symbolic PE



Micro-architecture of
reconfigurable neuro/symbolic PE

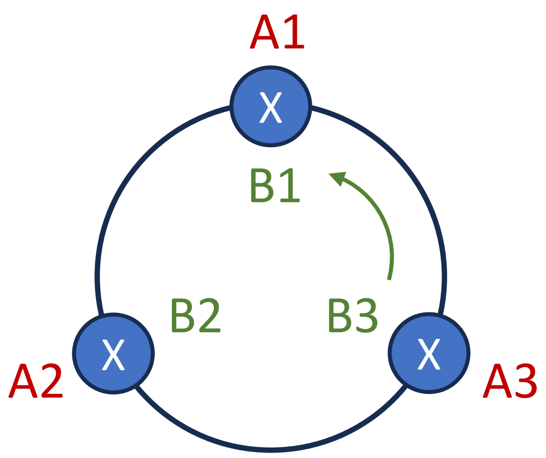


Operation mode of
reconfigurable neuro/symbolic PE

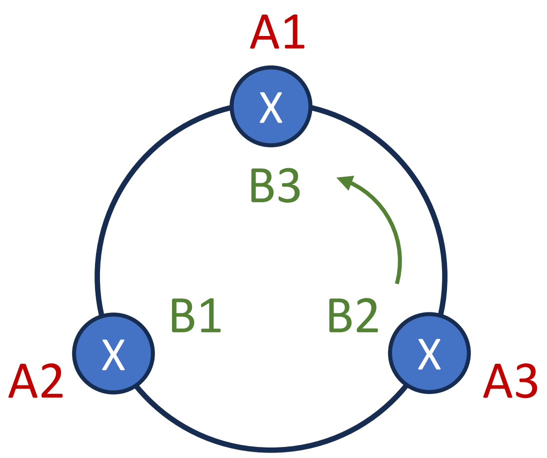
Reconfigurable neuro/symbolic PE incurs **low area overhead** based on systolic array PE;
The PE is reconfigurable for **three operation modes**: load, neuro, symbolic

What is Circular Convolution?

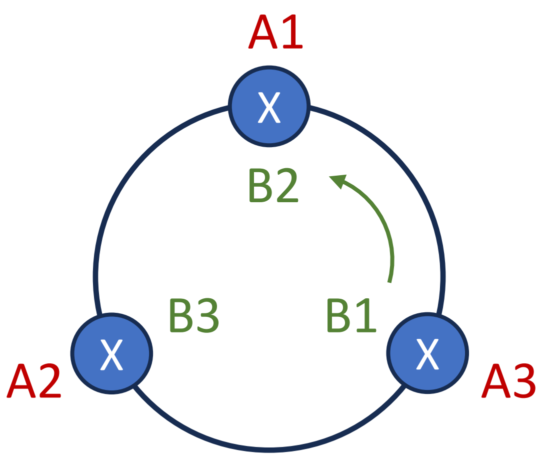
$$\begin{pmatrix} A1 \\ A2 \\ A3 \end{pmatrix} \odot \begin{pmatrix} B1 \\ B2 \\ B3 \end{pmatrix} = \begin{pmatrix} A1B1+A2B2+A3B3 \\ A1B3+A2B1+A3B2 \\ A1B2+A2B3+A2B1 \end{pmatrix}$$



A1B1+A2B2+A3B3



A1B3+A2B1+A3B2



A1B2+A2B3+A2B1

Bubble Streaming Dataflow

Vector-Symbolic Circular Convolution Example (3 CircConv):

CircConv #1: (A1, A2, A3) ⊙ (B1, B2, B3)

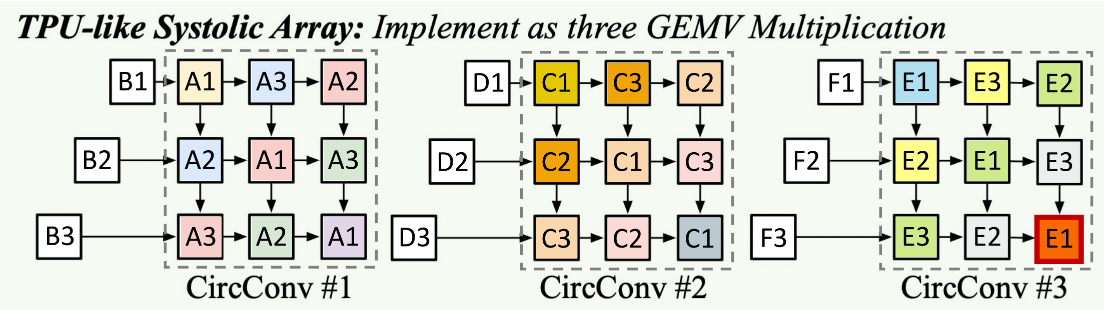
CircConv #2: (C1, C2, C3) ⊙ (D1, D2, D3)

CircConv #3: (E1, E2, E3) ⊙ (F1, F2, F3)

CircConv #1 Computation:

(A1, A2, A3) ⊙ (B1, B2, B3) =

(A1B1+A2B2+A3B3, A1B3+A2B1+A3B2, A1B2+A2B3+A2B1)



Cycles:

TPU: Finish at (3n+15) = 24 cycles

For symbolic operation:

- TPU-like array **suffers from** low parallelism & high memory access;

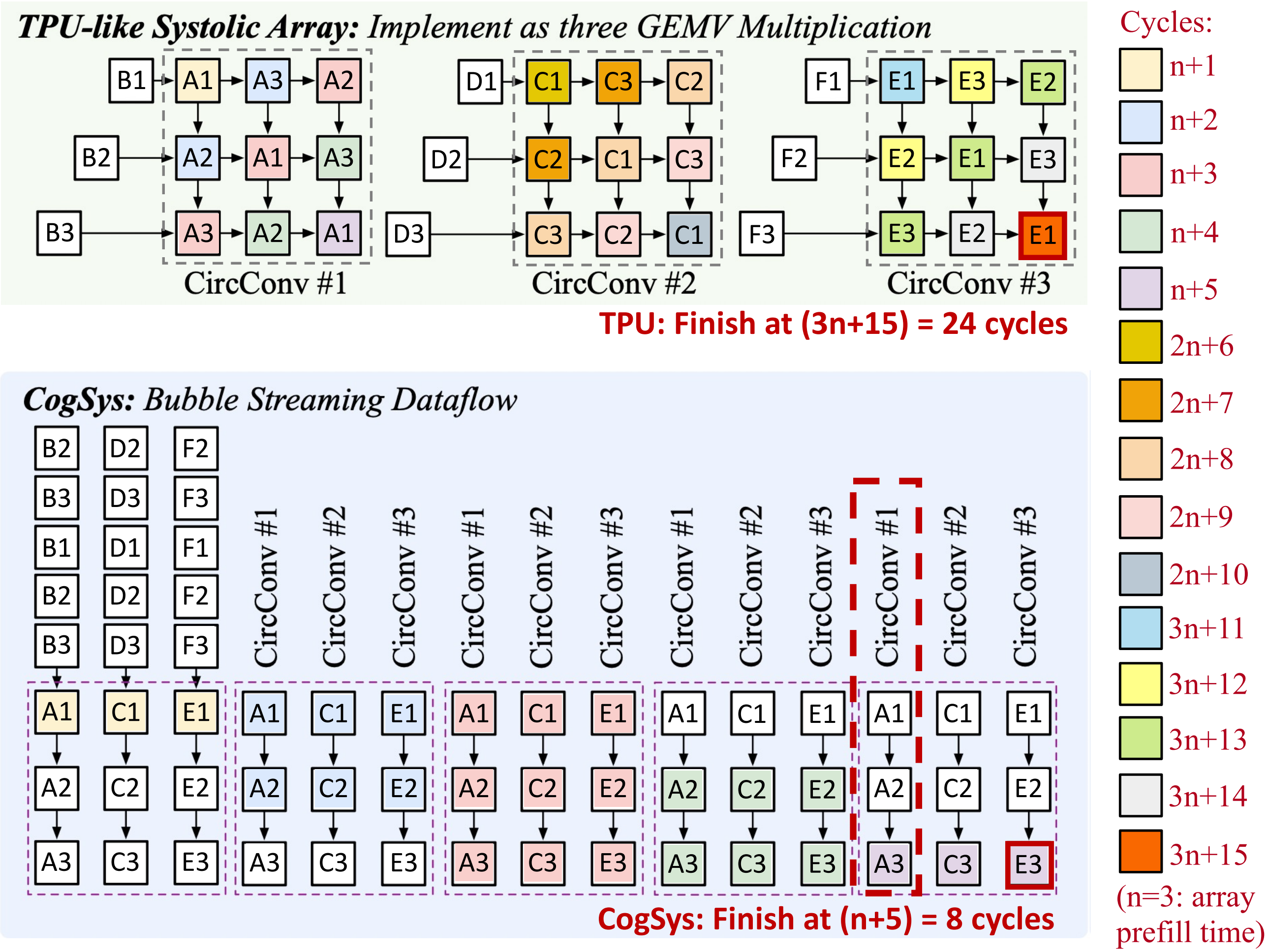
Bubble Streaming Dataflow

Vector-Symbolic Circular Convolution Example (3 CircConv):
CircConv #1: (A1, A2, A3)⊙(B1, B2, B3)
CircConv #2: (C1, C2, C3)⊙(D1, D2, D3)
CircConv #3: (E1, E2, E3) ⊙(F1, F2, F3)

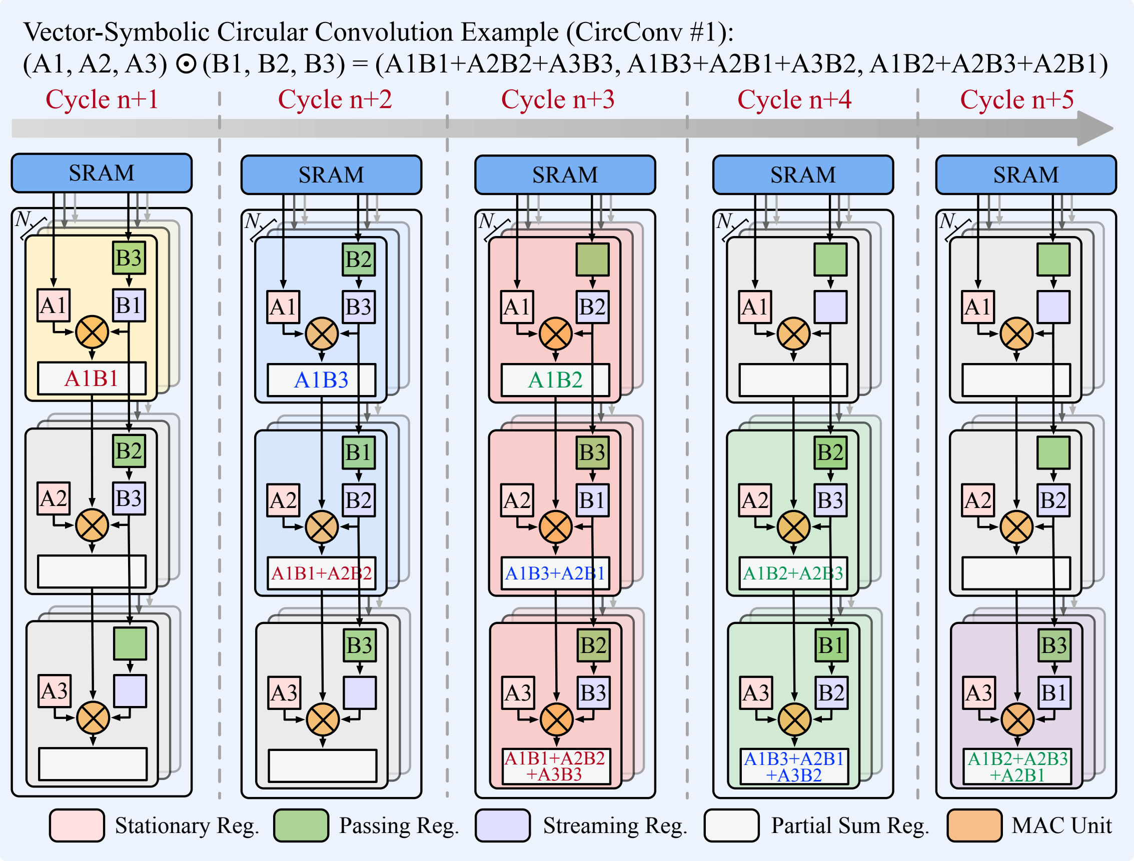
CircConv #1 Computation:
(A1, A2, A3) ⊙ (B1, B2, B3) =
(A1B1+A2B2+A3B3, A1B3+A2B1+A3B2, A1B2+A2B3+A2B1)

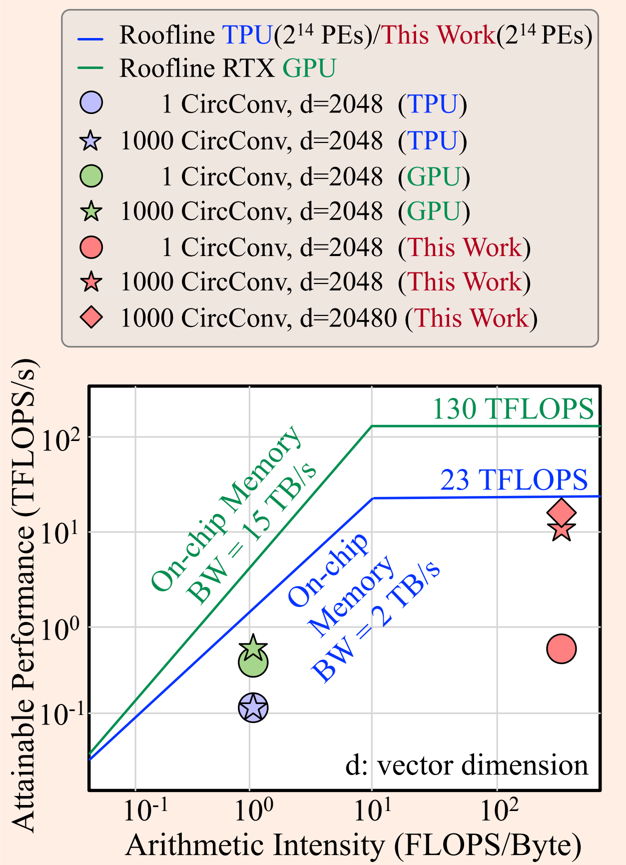
For symbolic operation:

- TPU-like array **suffers from** low parallelism & high memory access;
- Bubble streaming dataflow **improve parallelism, arithmetic intensity, and data reuse.**



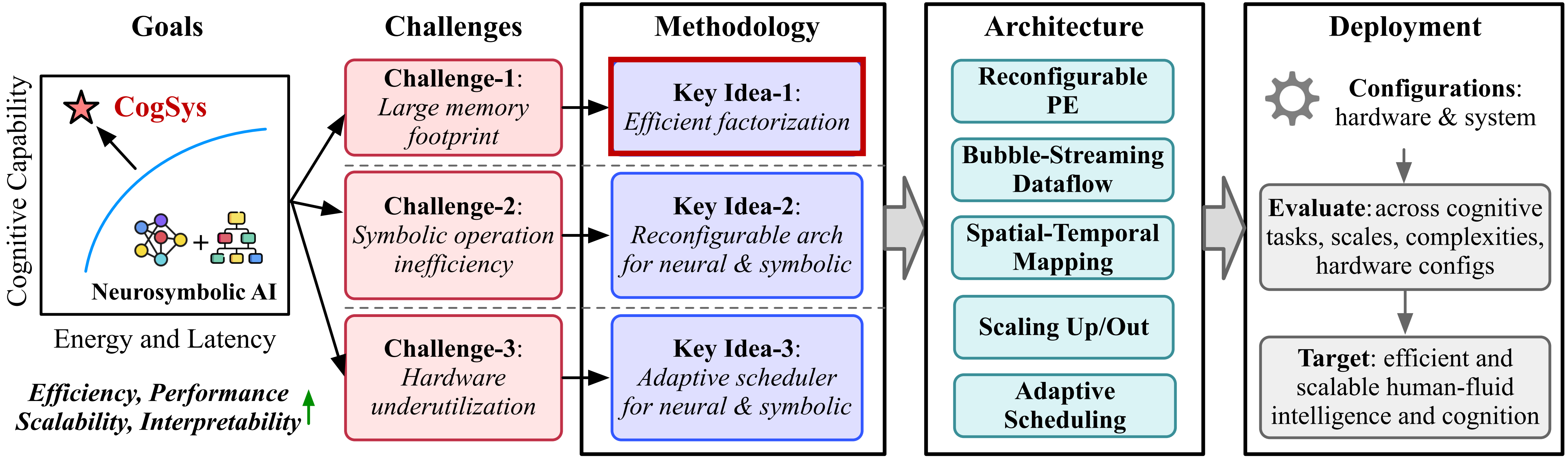
Bubble Streaming Dataflow



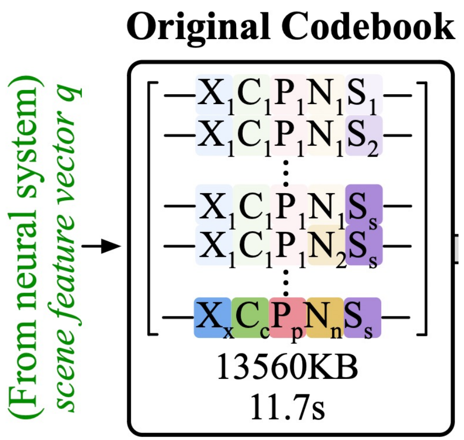


Bubble streaming dataflow flow improve parallelism, arithmetic intensity, and data reuse

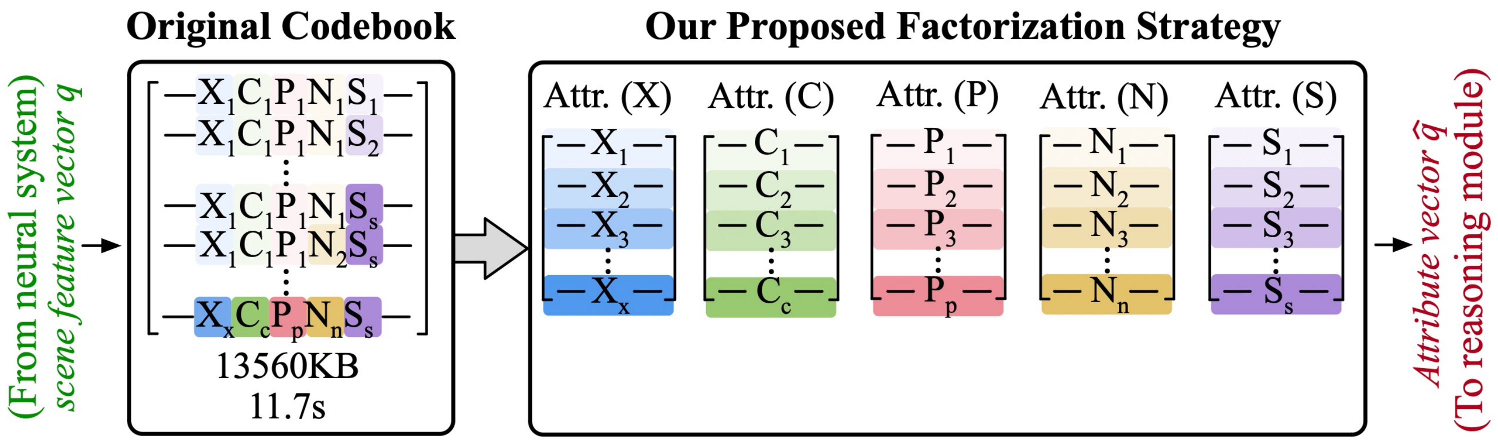
Our Methodology



Algorithm Optimization – Efficient Factorization

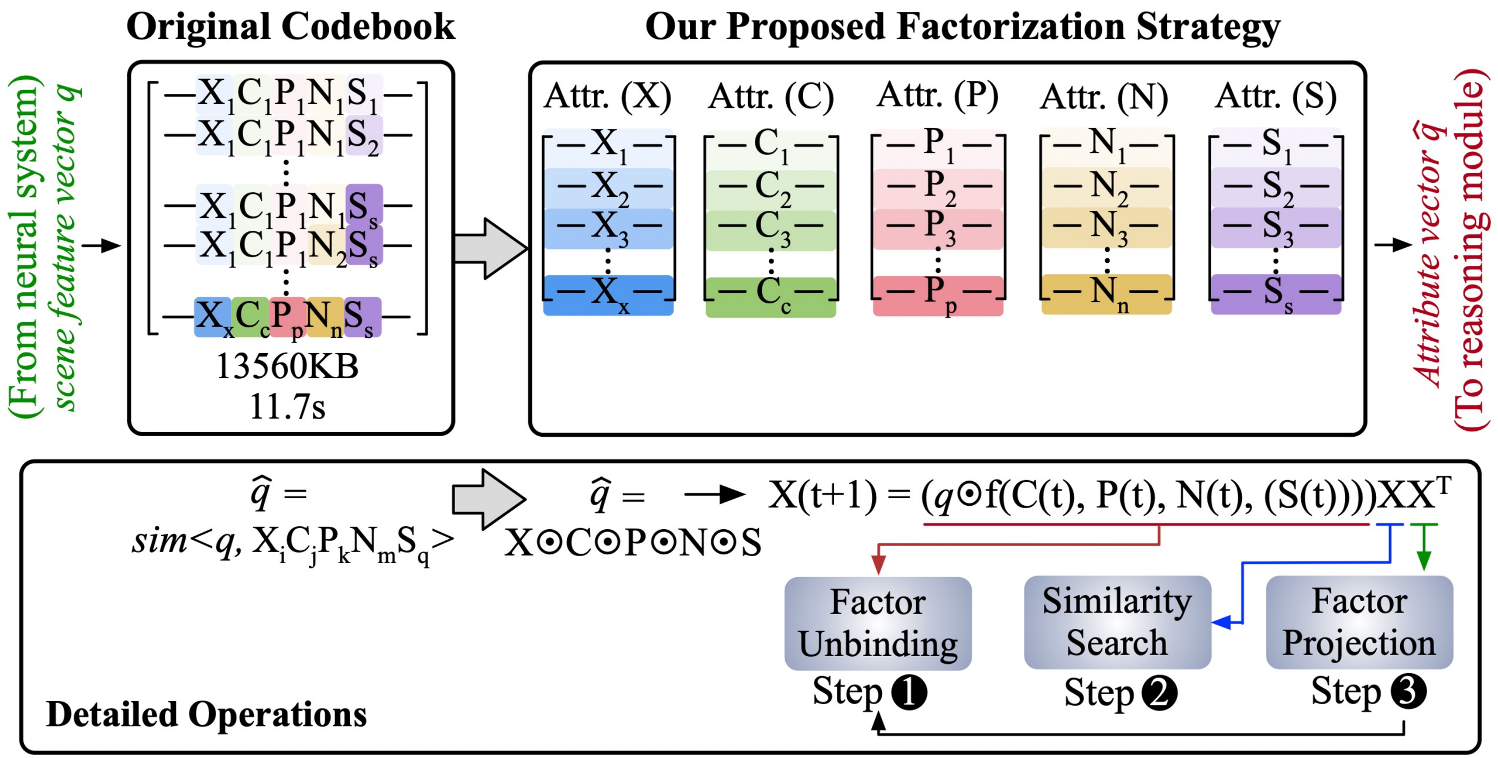


Algorithm Optimization – Efficient Factorization



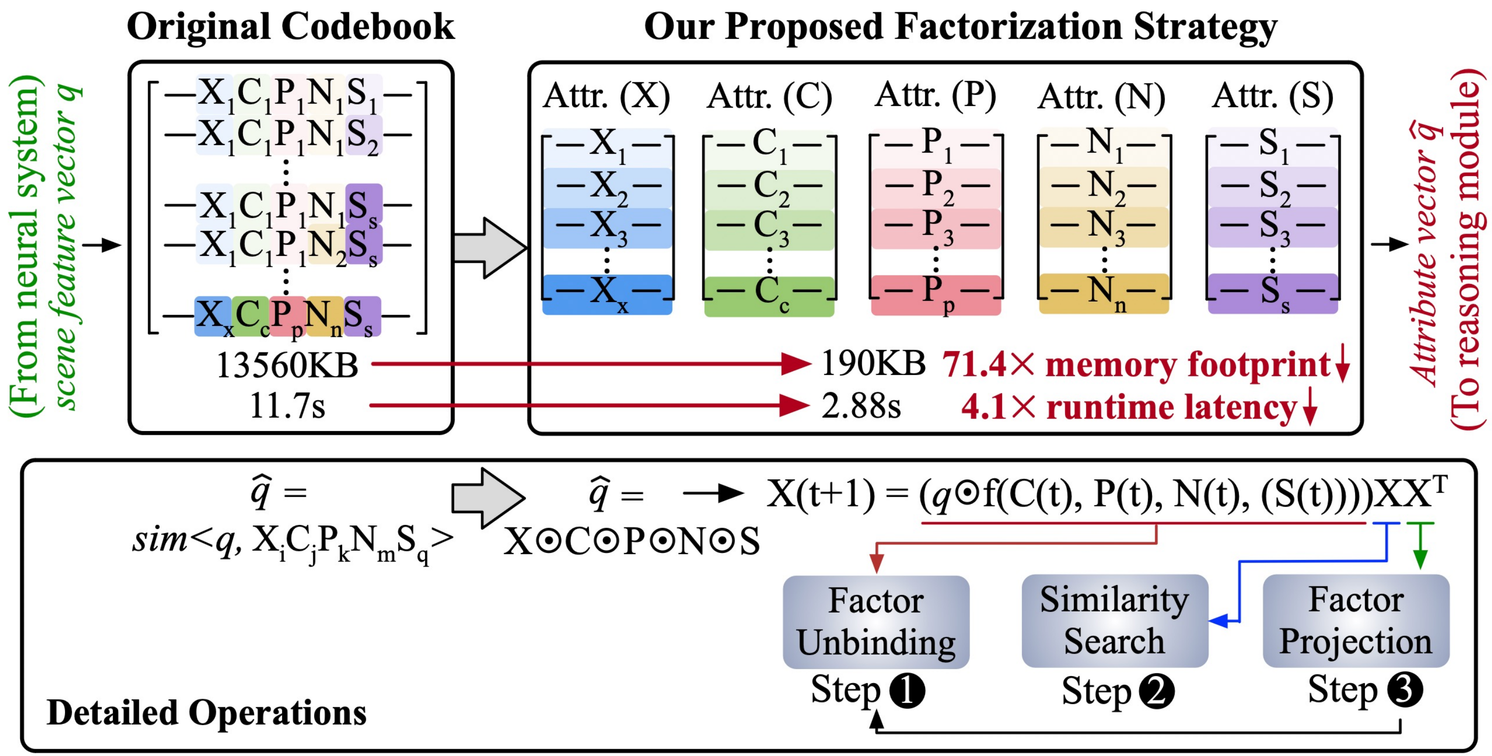
Factorization **disentangles** large symbolic knowledge codebook into small volume of attributes

Algorithm Optimization – Efficient Factorization



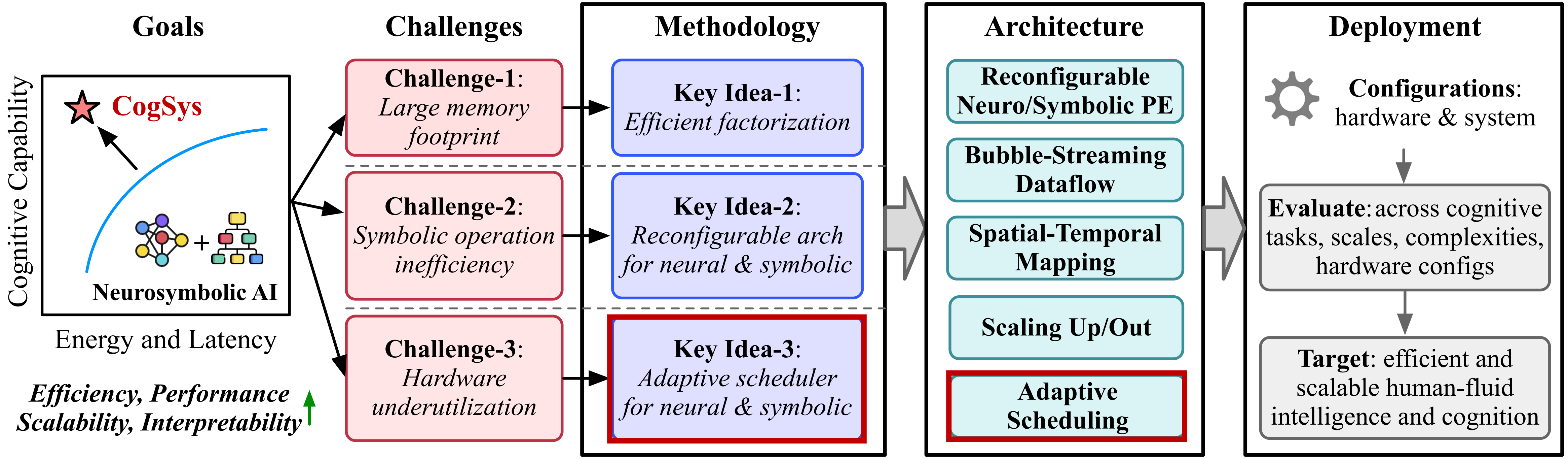
Factorization **disentangles** large symbolic knowledge codebook into small volume of attributes

Algorithm Optimization – Efficient Factorization

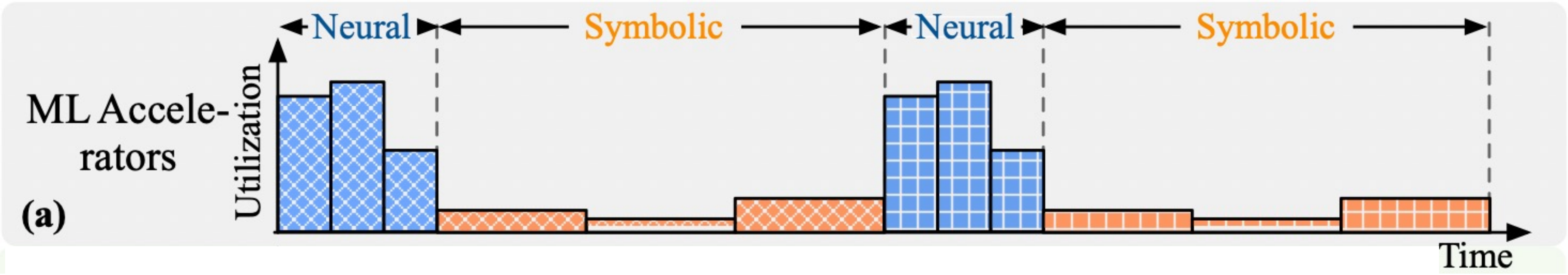


Factorization **disentangles** large symbolic knowledge codebook into small volume of attributes, thus **reducing computational time and space complexity**

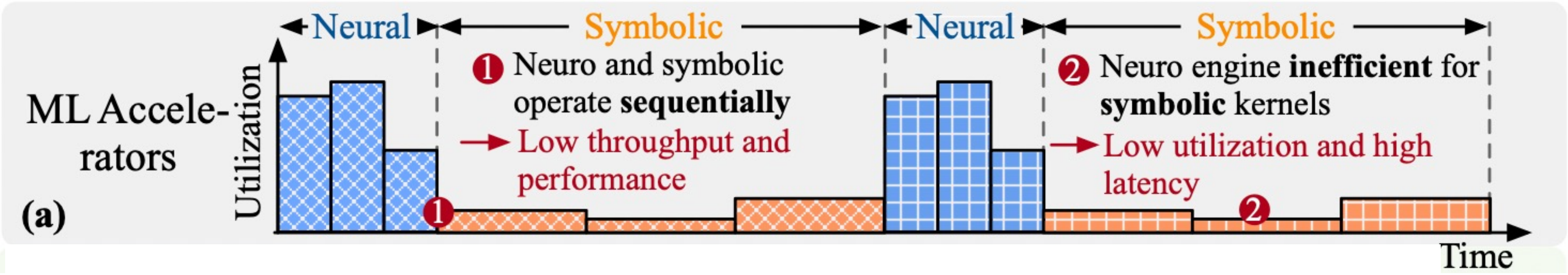
Our Methodology



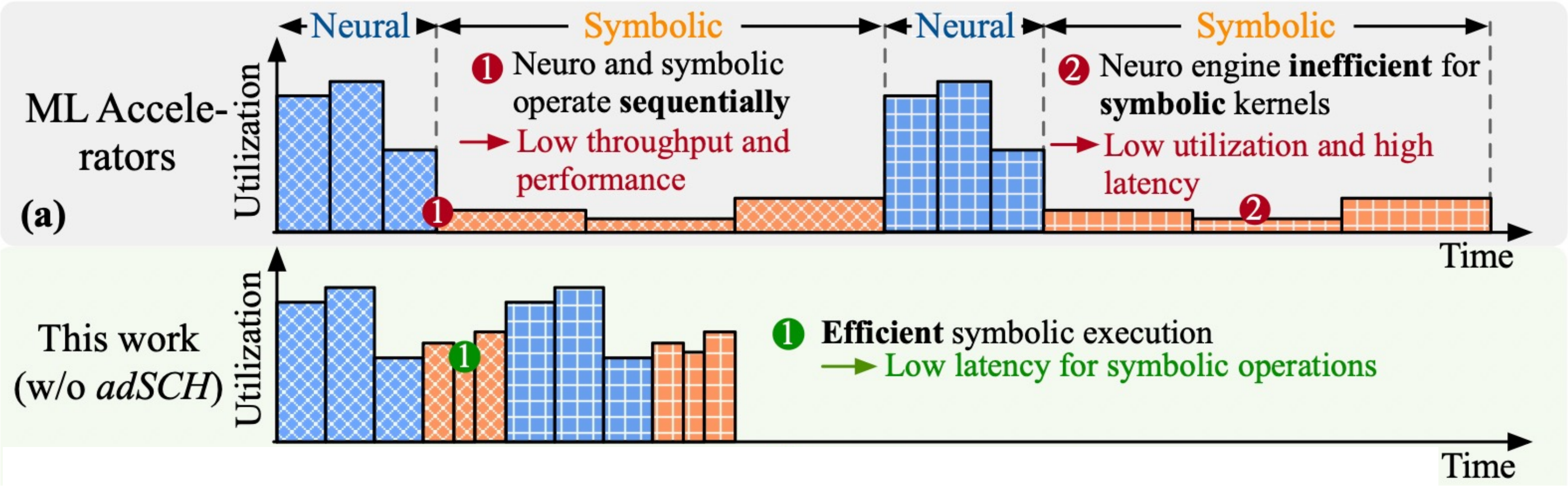
System Optimization - Adaptive Scheduling



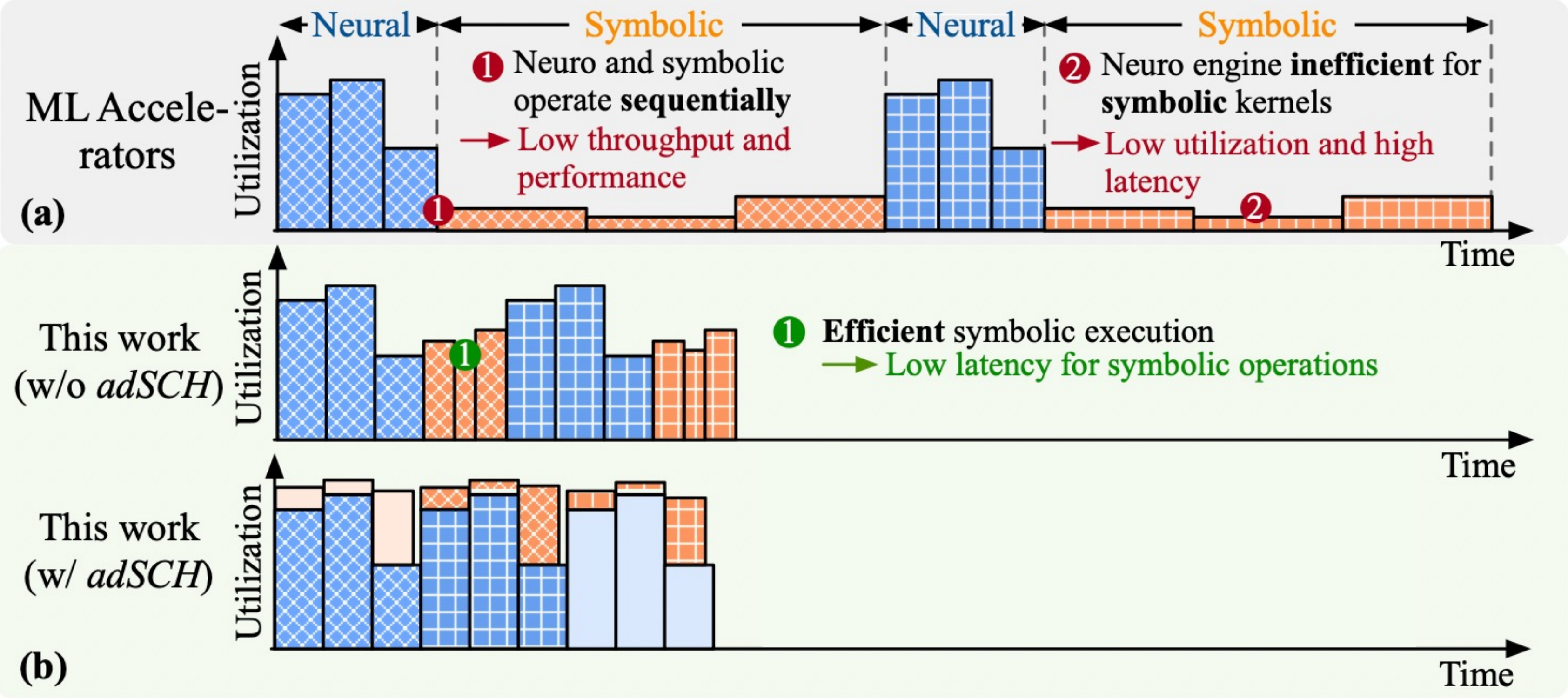
System Optimization - Adaptive Scheduling



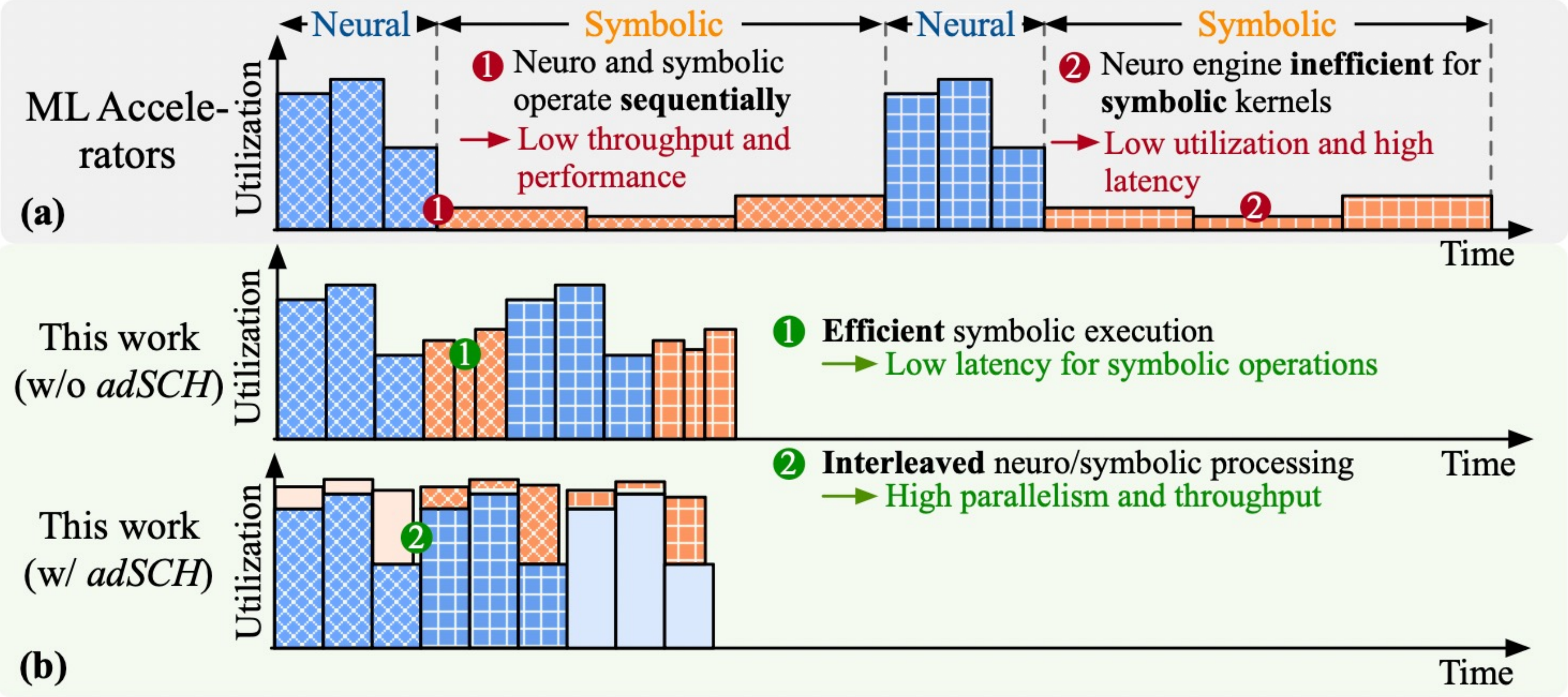
System Optimization - Adaptive Scheduling



System Optimization - Adaptive Scheduling

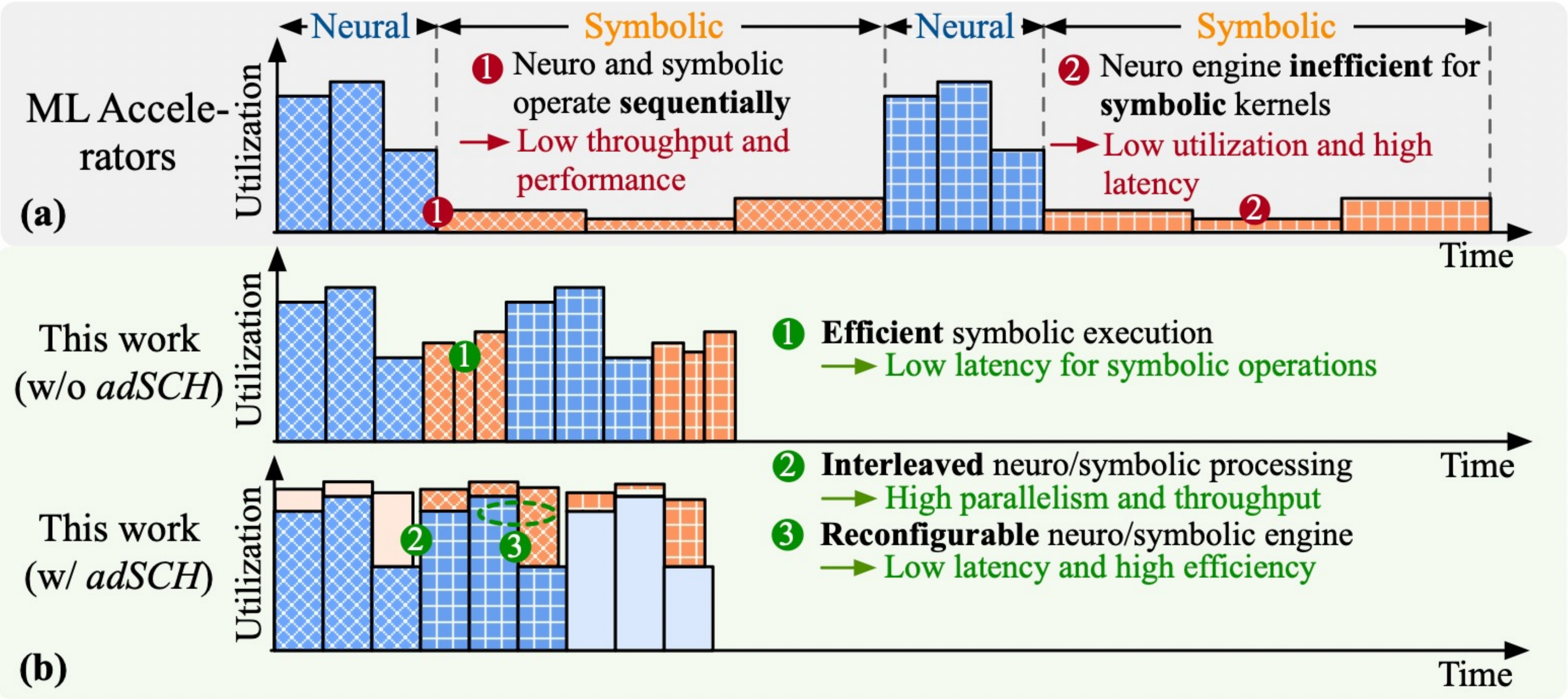


System Optimization - Adaptive Scheduling



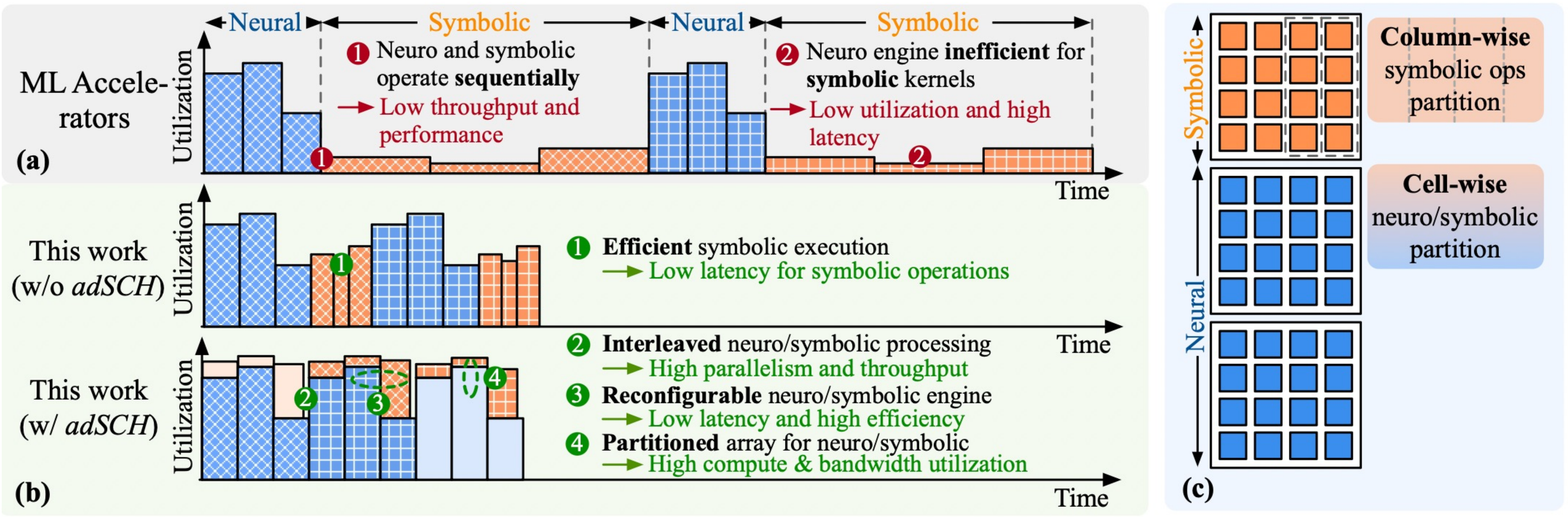
Adaptive scheduling enables **interleaved**

System Optimization - Adaptive Scheduling



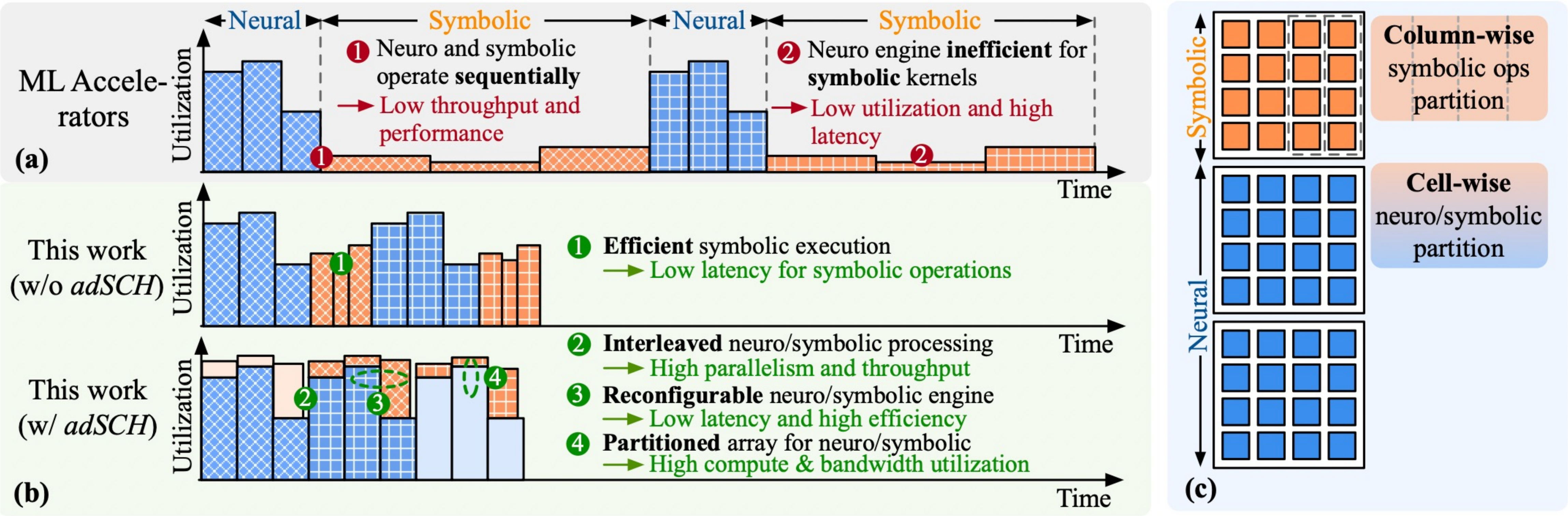
Adaptive scheduling enables **interleaved** and **reconfigurable** neuro/symbolic processing

System Optimization - Adaptive Scheduling



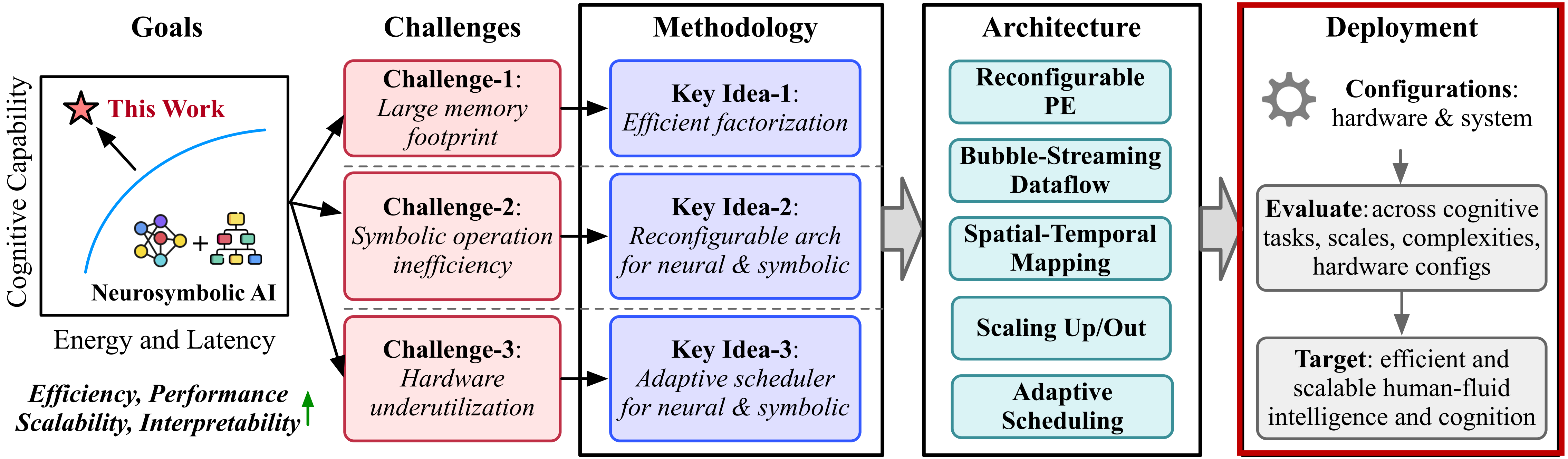
Adaptive scheduling enables **interleaved** and **reconfigurable** neuro/symbolic processing with **partitioned array**

System Optimization - Adaptive Scheduling



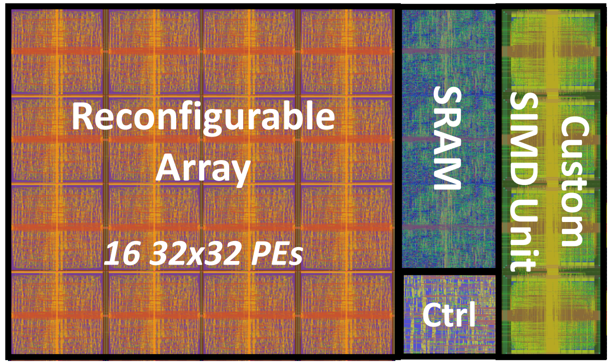
Adaptive scheduling enables **interleaved** and **reconfigurable** neuro/symbolic processing with **partitioned array**, improving parallelism, latency, efficiency, and utilization

Our Methodology



Evaluation – Setup and Accelerator Layout

Layout of Neuro-Symbolic Accelerator



Accelerator Specs

Technology	28 nm	Frequency	600 MHz
#Arrays	16	Voltage	1 V
Size of Each Array	32x32	Power	1.48 W
SRAM	4.5 MB	Area	4.9 mm^2

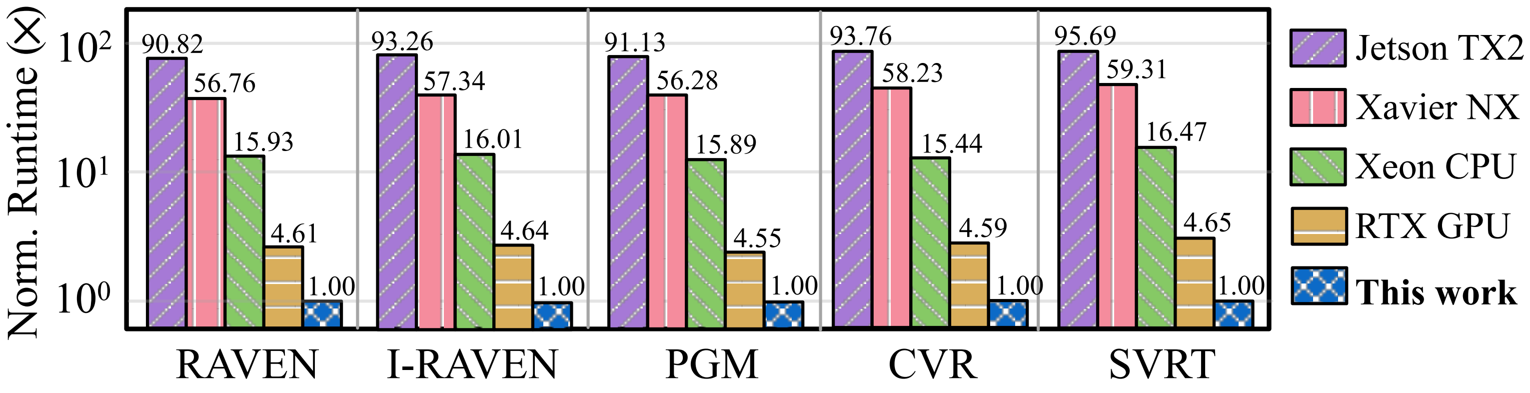
- **Task**: Cognitive reasoning tasks
- **Reasoning datasets**:
  - RAVEN, I-RAVEN, PGM, CVR, SVRT
- **Neuro-symbolic workloads**:
  - NVSA, MIMONet, LVRF
- **Hardware baseline**:
  - Jetson TX2, Xavier NX, RTX GPU, Xeon CPU
  - ML accelerators (TPU, MTIA, Gemmini)

Evaluation – Algorithm Performance

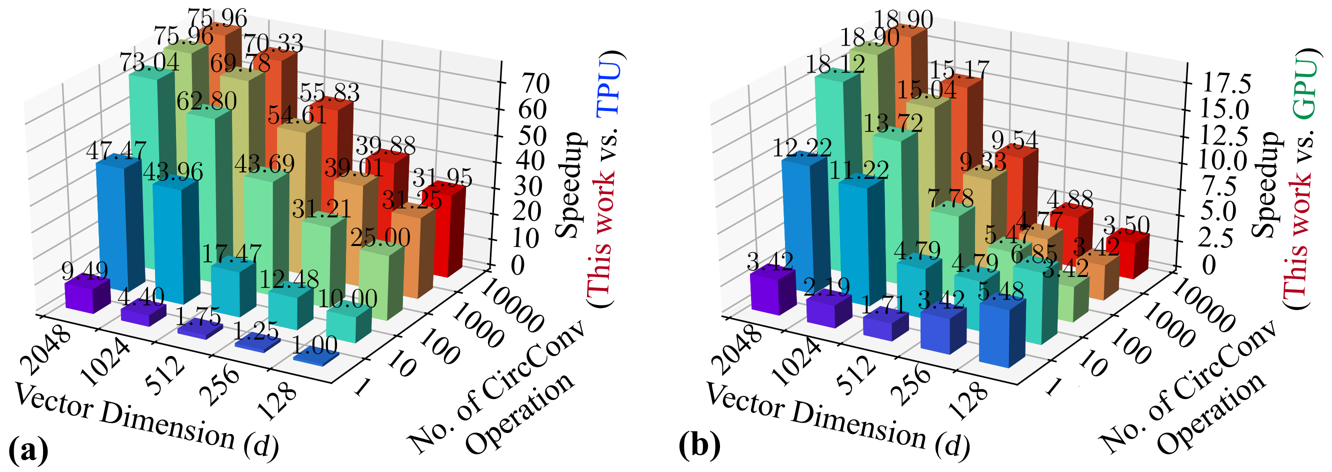
Dataset Accuracy	Neurosymbolic Model			Non-neurosymbolic		Human
	NVSA	**Our Design (+Algo Opt.)**	**Our Design (+Quant.)**	ResNet18	GPT-4	
RAVEN	98.5%	98.9%	98.7%	53.4%	89.0%	84.4%
I-RAVEN	99.0%	99.0%	98.8%	40.3%	86.0%	78.6%
PGM	68.3%	68.7%	68.4%	36.8%	56.0%	N/A
#Parameters	38 MB	32 MB	8 MB	42 MB	1.7 TB	N/A

- **Better Reasoning Capability:** neurosymbolic methods achieve high accuracy across reasoning tasks than NNs and human.
- **Smaller Memory Footprint:** neurosymbolic methods consume much less #parameter than NNs (e.g., LLM).

Evaluation – Hardware Performance

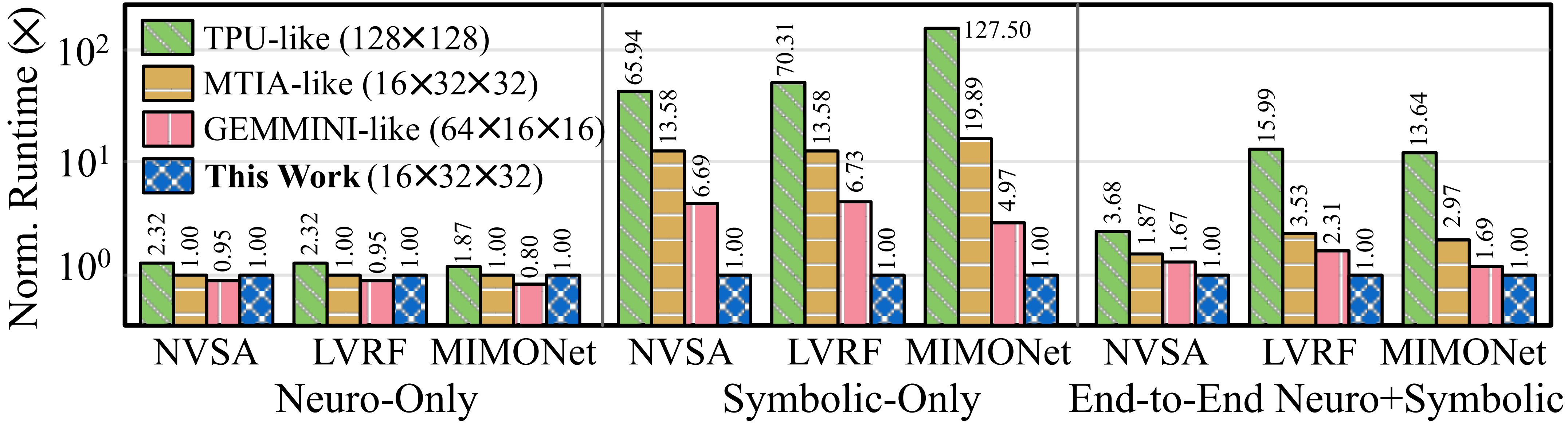


4x - 90x speedup compared to CPU/GPU



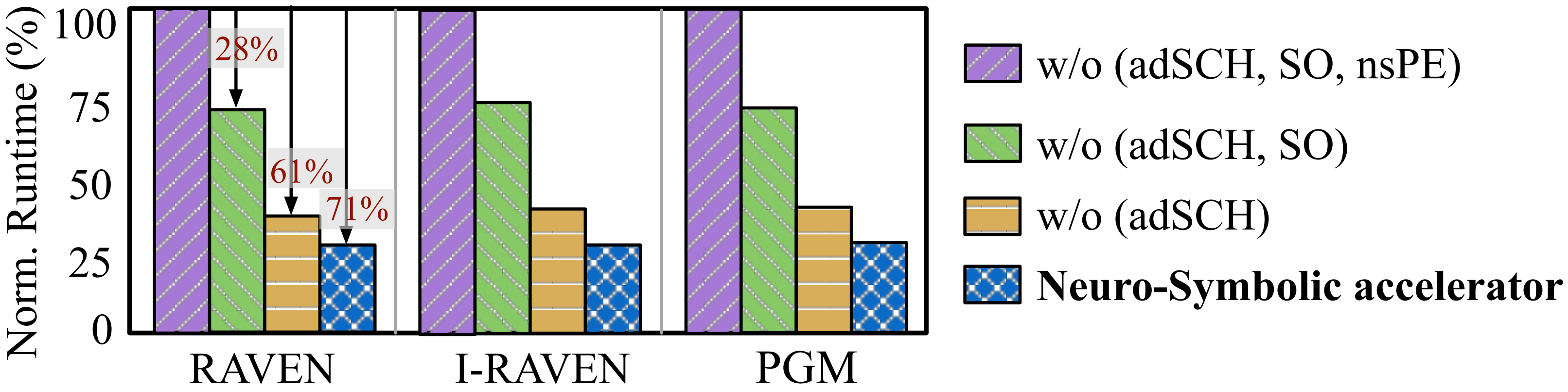
Symbolic operation:
75x speedup to TPU
18x speedup to GPU

Evaluation – Hardware Performance



Compared with ML accelerators: similar neuro latency, **7-120x symbolic** speedup, **2-16x end-to-end neuro-symbolic** speedup

Evaluation – Ablation Study



Proposed **scheduling**, reconfigurable **PE**, bubble streaming **dataflow** are effective

Neurosymbolic Cognitive Solution Algorithm @ Hardware	Normalized Runtime (%) on RAVEN	I-RAVEN	PGM	CVR	SVRT
NVSA @ Xavier NX	100	100	100	100	100
Proposed Algorithm @ Xavier NX	89.5%	88.9%	90.7%	87.6%	88.4%
Proposed Algorithm @ Proposed Accelerator	1.76%	1.74%	1.78%	1.72%	1.69%

Algorithm-system-hardware co-design is critical

Key Observations:

Compared with systolic arrays that only support neural, our design provides **reconfigurable support for neural and symbolic** operations with **only 4.8% area overhead**.

Our design achieves **0.3s latency** per cognition task, with **1.18W power** consumption.

CogSys Summary

- **Neuro-symbolic AI** is a compositional method to improve reasoning and interpretability.

- In this work,
  - Characterize **system implications**
  - Propose **algorithm-system-hardware co-design**
    - **Algorithm**: efficient factorization
    - **System**: adaptive scheduling
    - **Hardware Architecture**: reconfigurable neuro/symbolic PE, dataflow, mapping, and scaling strategy
  - Achieve **efficient and scalable neuro-symbolic** execution across reasoning tasks