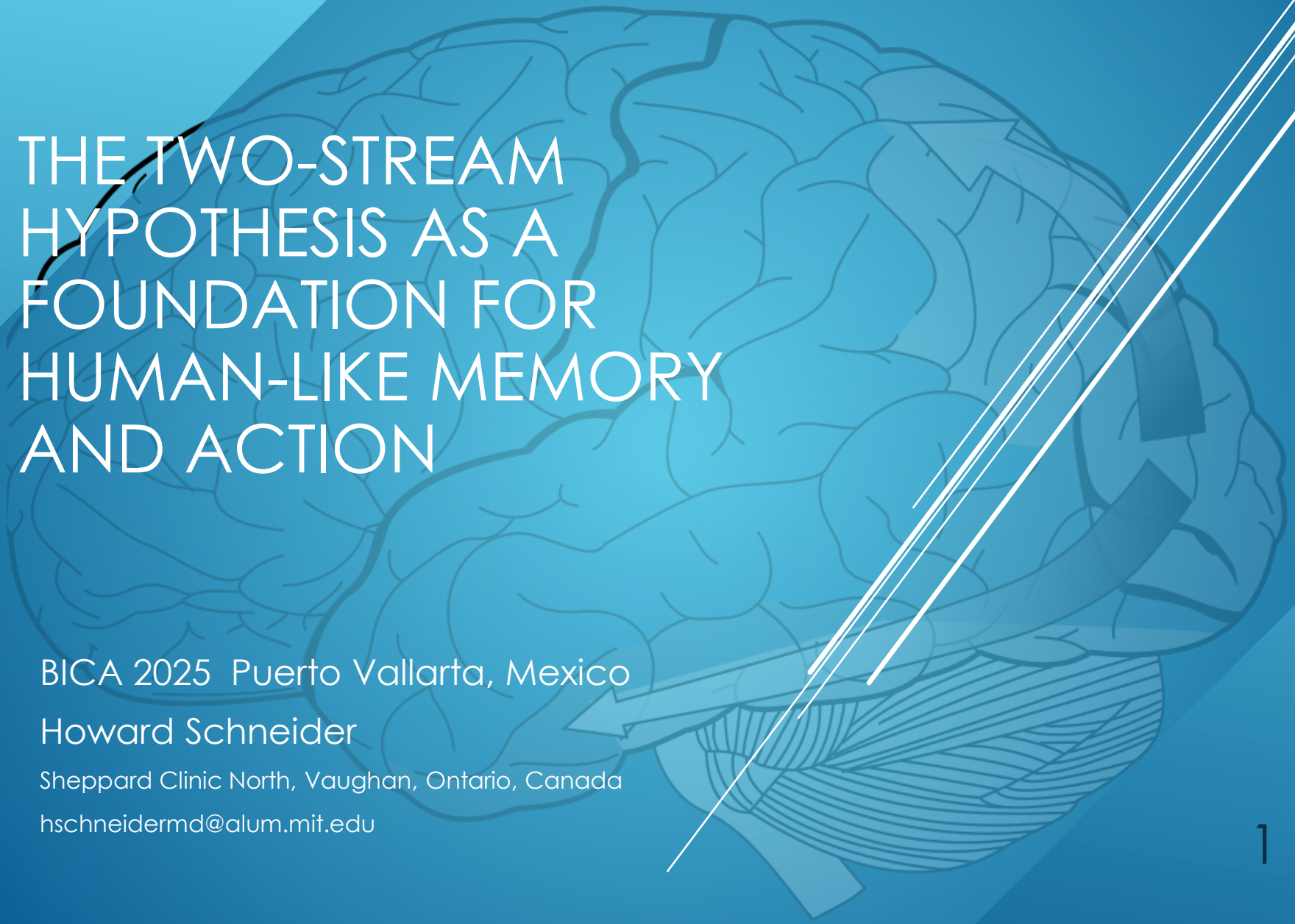
THE TWO-STREAM HYPOTHESIS AS A FOUNDATION FOR HUMAN-LIKE MEMORY AND ACTION



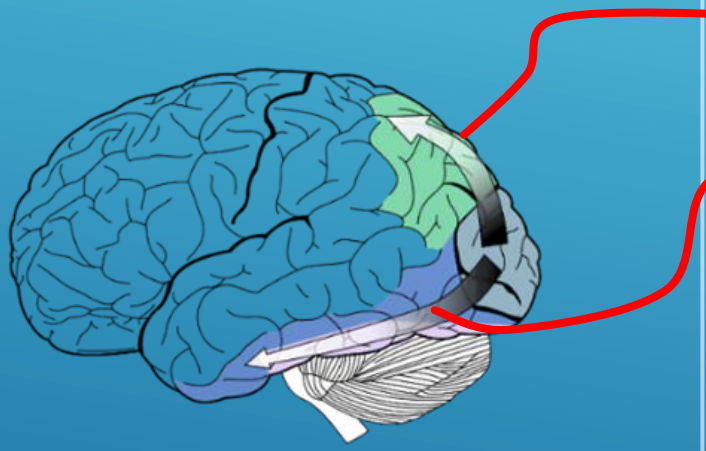
BICA 2025 Puerto Vallarta, Mexico

Howard Schneider

Sheppard Clinic North, Vaughan, Ontario, Canada

hschneidermd@alum.mit.edu

INTRODUCTION



►- The two-stream hypothesis in neuroscience posits separate visual pathways:

►- Dorsal stream: "where/how" (spatial/motor guidance)

► - Ventral stream: "what" (object identity)

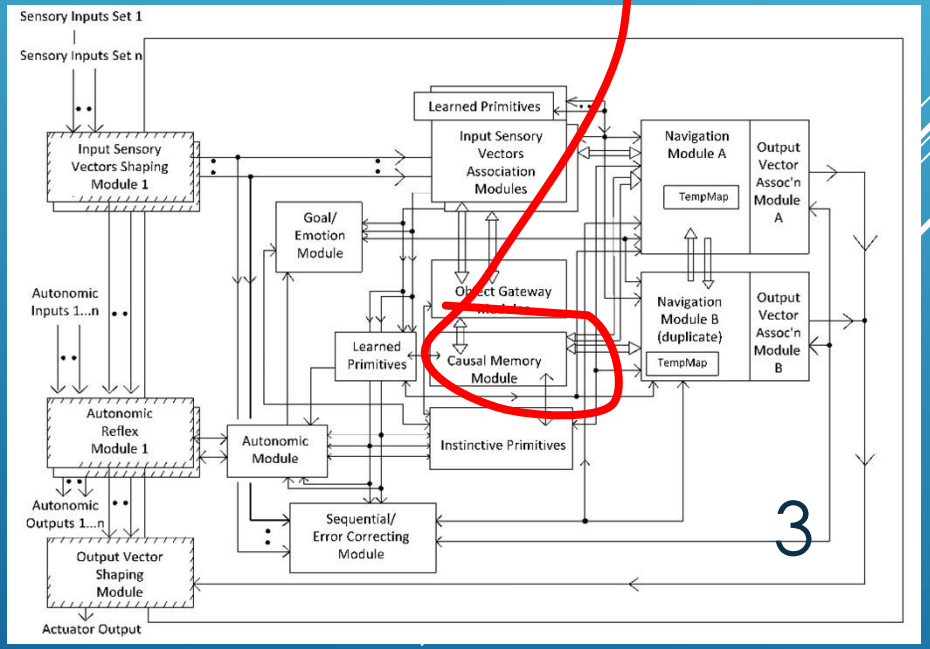
►- Similar dorsal/ventral separation seen in auditory pathways

►- Found in all mammals

►- **Question: Is this division a biological artifact or a functional advantage?**

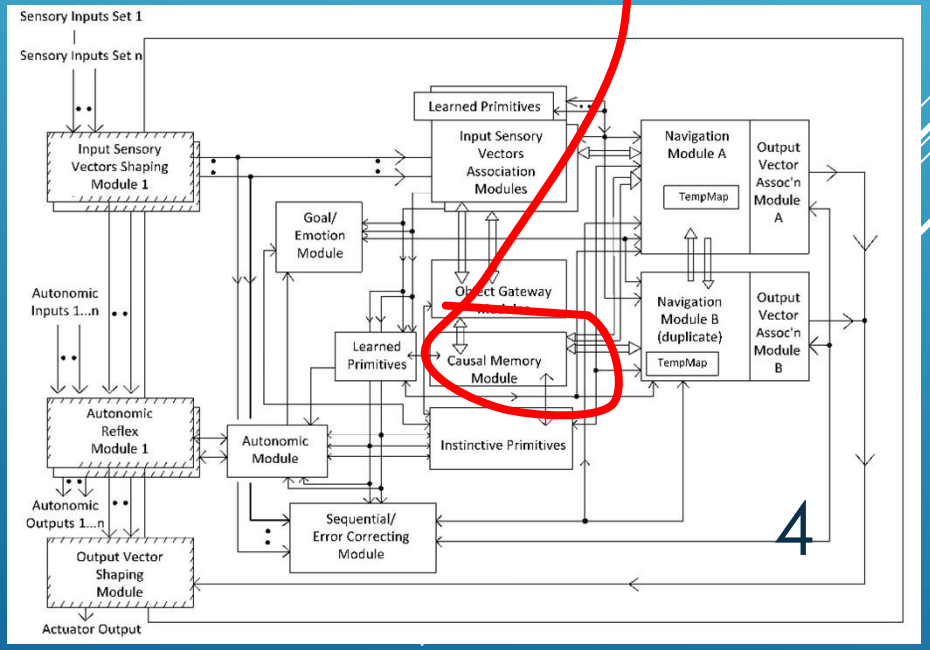
MORE EFFICIENT TO HOLD ‘WHAT’ AND ‘WHERE/HOW’ INFORMATION IN ONE LOCATION?

OR ADVANTAGE TO SPLITTING THE INFORMATION?



MORE EFFICIENT TO HOLD 'WHAT' AND 'WHERE/HOW' INFORMATION IN ONE LOCATION?

→ RELEVANT TO COGNITIVE ARCHITECTURE DESIGN



RELATED WORK

- ACT-R cognitive architecture -> ACT-R/PM with “visual-location” and “visual-object”, but model
  trade-offs are unclear
- Work by Mishkin et al. (macaque lesions), Goodale and Milner (humans)

APPROACH

- An advantage of splitting identification circuits ("what") and spatial guidance circuits ("where/how") is to parallelize and specialize resource usage
- A disadvantage of the dorsal/ventral split is the need to integrate information

APPROACH

- We create and evaluate a simulation model of a **"split-store" model** (i.e., "what" and "where/how" circuits separate)

versus

- a **"unified store" model** (i.e., "what" and "where/how" data in same location)

APPROACH

- Unified store: both object and spatial data in one location (e.g., CCA7 architecture)
- Split store: separate circuits for object identification and spatial localization (e.g., brain)
- Hypothesis: separation may enable specialization and parallel processing
- **Key question: does this separation provide performance benefits in the brain? In cognitive architectures?**

METHODS

- **Python-based simulation** (Python 3.11.4)
- Environment:
  - 100 million position 2D grid
  - 5000 unique objects with random features
- Workload: 10,000 queries per condition
  - Canonical (i.e., default) workload: 40% "what", 40% "where", 20% integrated
- Code available: https://github.com/howard8888/bica25/blob/main/two_stream_ver01.pyj

```
    results = []
    for run in range(TRIALS):
        logging.info(f"Starting simulation run {run+1}")
        print(f"\nStarting simulation run {run+1}")

        # Create one ToM agent and one non-ToM agent for each
        agent_true: Optional[CCA7Agent] = CCA7Agent(True)
        agent_false: Optional[CCA7Agent] = CCA7Agent(False)
        cycles_true, cycles_false = 0, 0  #number of cycles t

        try:
            for cycle in range(1000):
                #set up global scarcity/plenty conditions for
                if CCA7Agent.global_energy_supply < 0.8:
                    CCA7Agent.replenish_energy(0.2)

                #simulates an agent's chance (set at 50%) of
                presence = np.random.choice([0, 1]) #0 or 1 c

                if agent_true is not None and agent_true.cog_
                    #note that once agent_true is expelled an
```

COMPARISON MODELS

- **Unified model:** one structure for both object and spatial data
- **Split model:** two structures, one for "what" and one for "where/how"

→ *Each model evaluated on identical queries and object distributions*

RESULTS

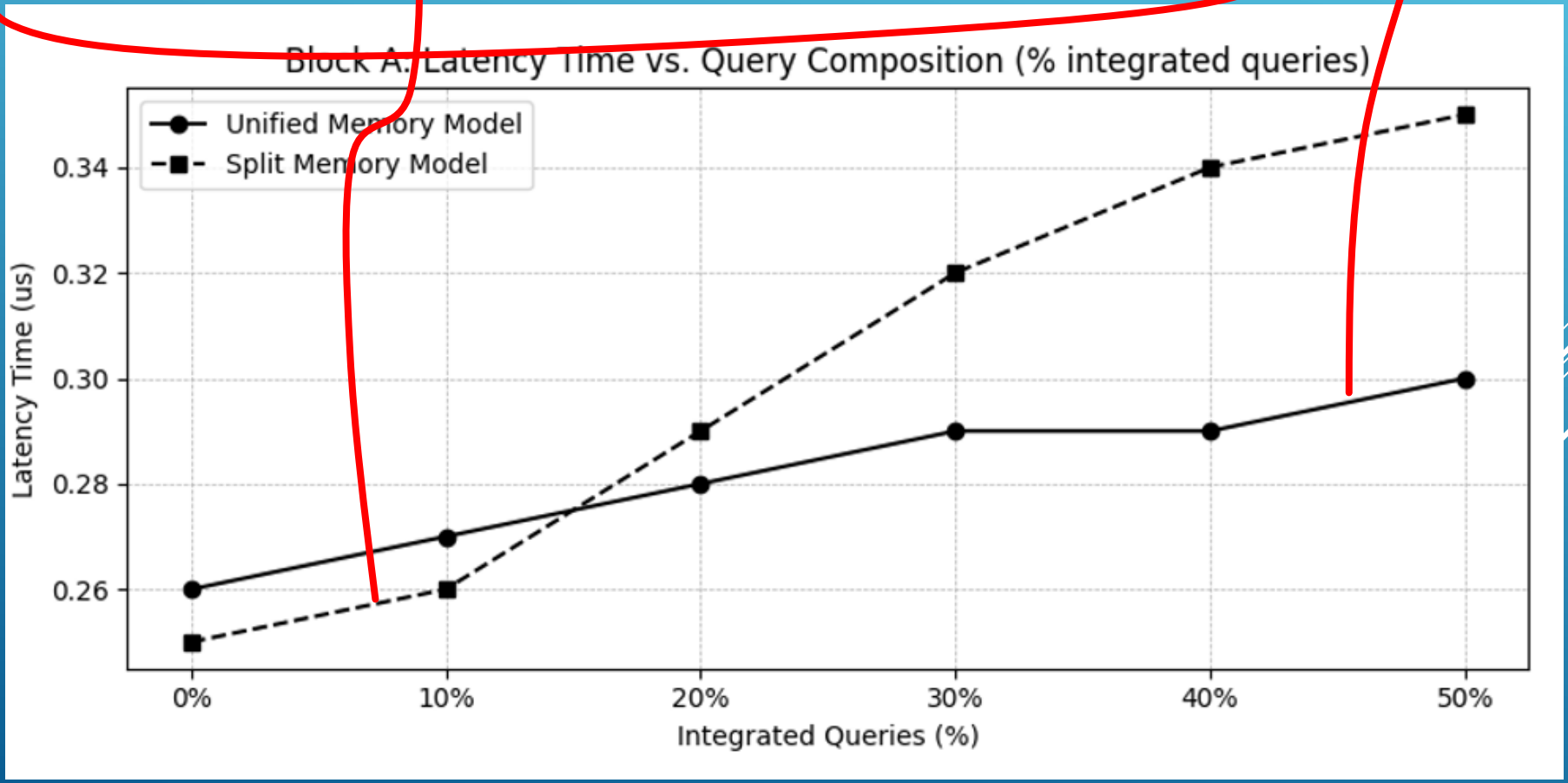
1. Baseline timings for canonical workload (40%/40%/20%)
2. Parameter-sweeps:
   2A. Sweep (i.e., vary) composition (Block A)
   2B. Sweep number of objects (Block B)
   2C. Sweep the amount of parallelism (Block C)

1. QUERY TIMES ON A CANONICAL 40% WHERE/40% WHAT /20% INTEGRATED MIX OF 10,000 QUERIES FOR THE UNIFIED VS SPLIT

Model	$\mu \pm \sigma$ query time (latency in µs)	Relative Δ
Unified	0.29 ± 0.02	—
Split	0.34 ± 0.03	+ 17 % slower

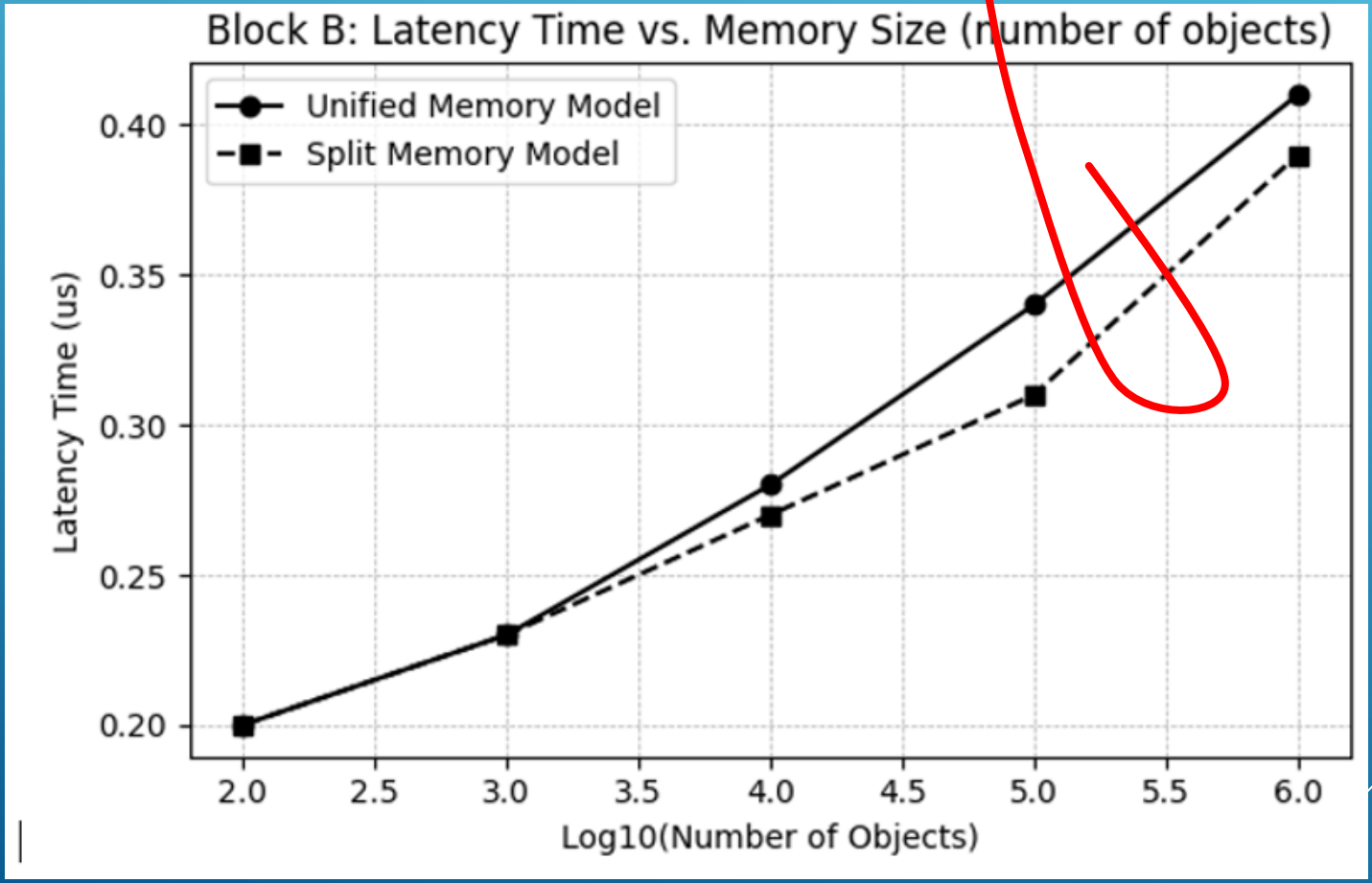
2A. TIME VS %INTEGRATED QUERIES

- Unified model faster with more integrated queries
- Split model outperforms with more non-integrated queries
- Integration overhead is costly in split model



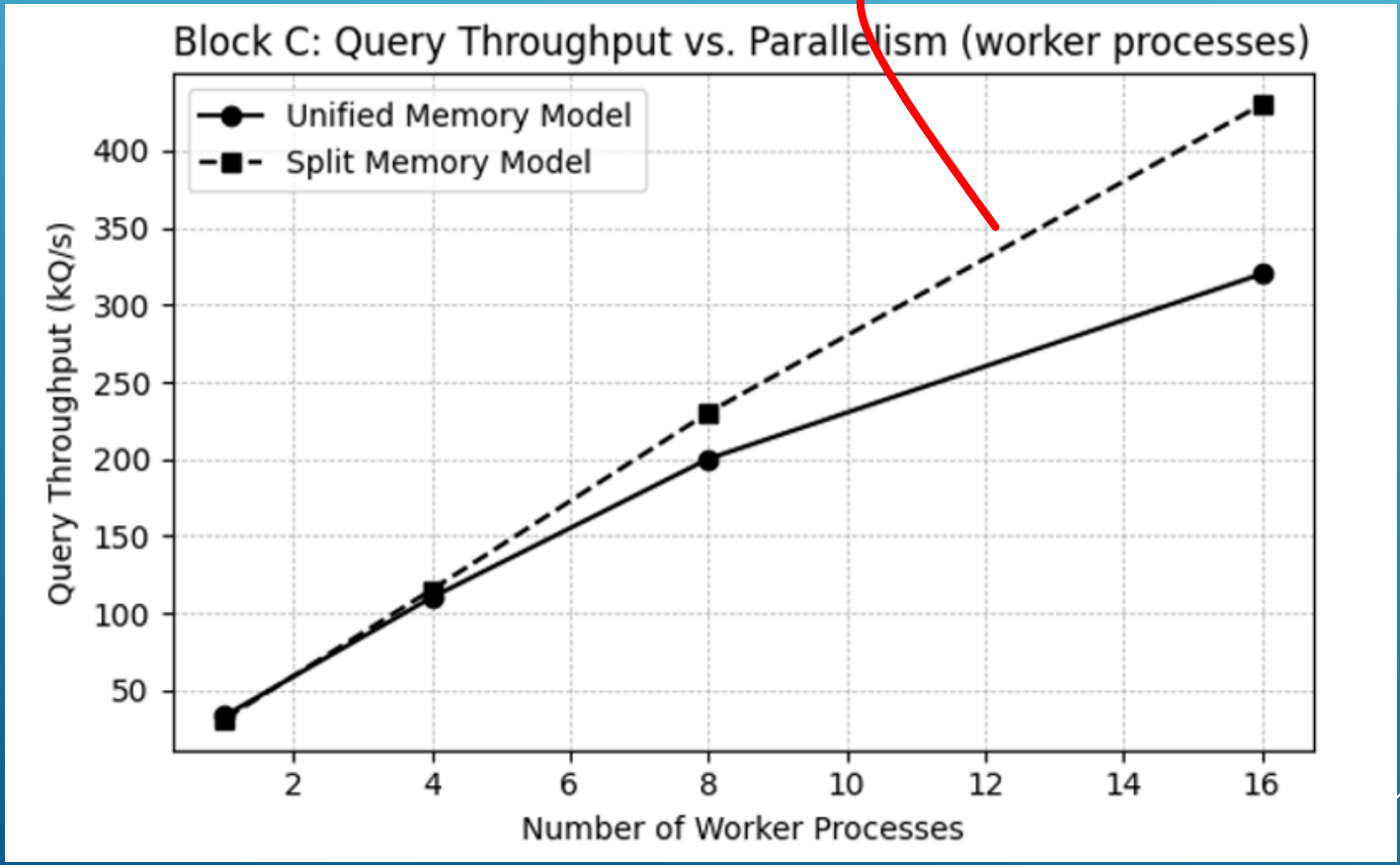
2B. TIME VS NUMBER OF OBJECTS

- Increasing object count slows unified model significantly
- Split model shows better scalability with object count



2C. THROUGHPUT VS. PARALLELISM

- Parallelism reduces latency in split model
- Unified model less able to benefit from parallel query processing
- Split model becomes faster with high parallelism



RESULTS SUMMARY

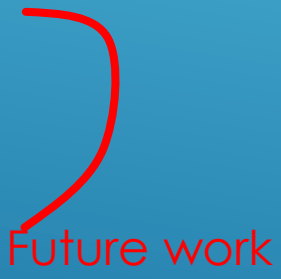
- The unified model was faster when there were substantial number of integrated queries, although for **higher % routine non-integrated queries, the split-model was faster**
- For larger number of objects in the envr't, the unified model slows down slightly
- With more **parallel processing, the split-model was faster**

DISCUSSION

From a brain evolutionary approach, having different “what” and “where/how” circuits **allows each to become more specialized** – “where/how” needs a fast visuomotor loop while “what” needs more richly structured data

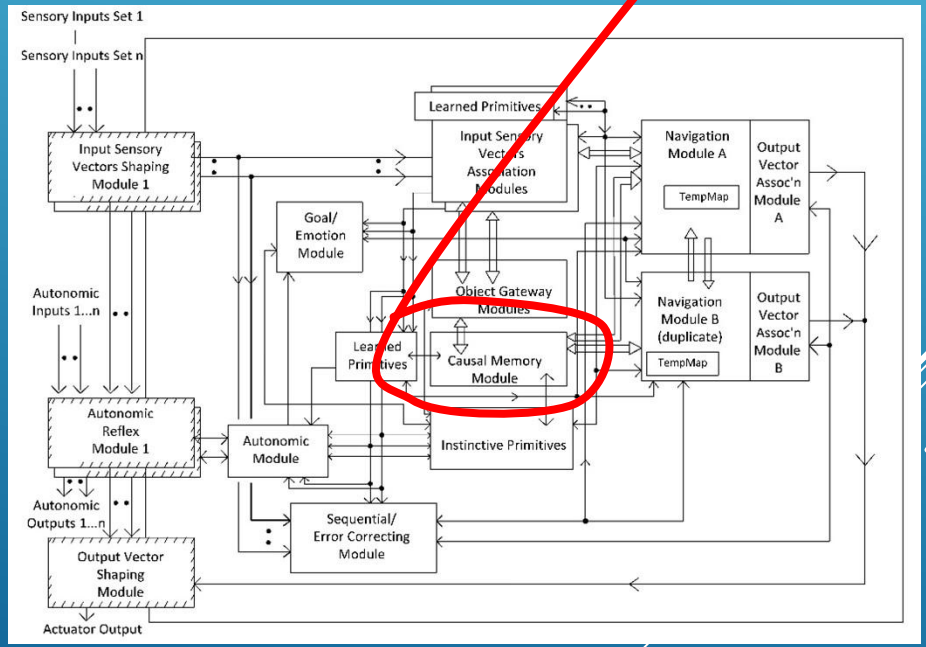
Simulation shows -- with **massive parallelism**, as exists in the brain, the **split-model is no longer slow** compared to the unified model

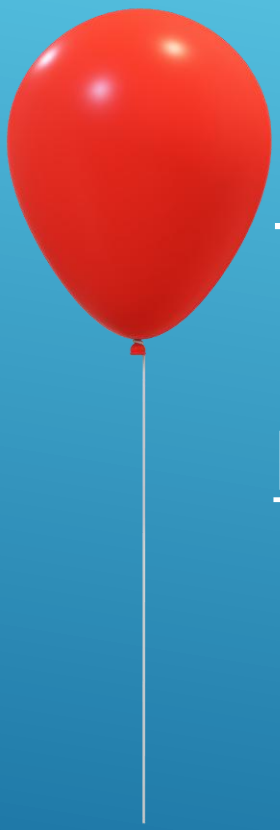
From a cognitive point of view, **the misalignment between "what" and "where" memories forces the brain to perform binding operations, possibly allowing compositional and creative results to emerge**



The **dorsal-ventral (“where/how”-”what”) pathways are not just a biological artifact, but a viable design principle for future brain-inspired cognitive architectures (BICAs)**

Future work





Thank you

hschneidermd@alum.mit.edu