



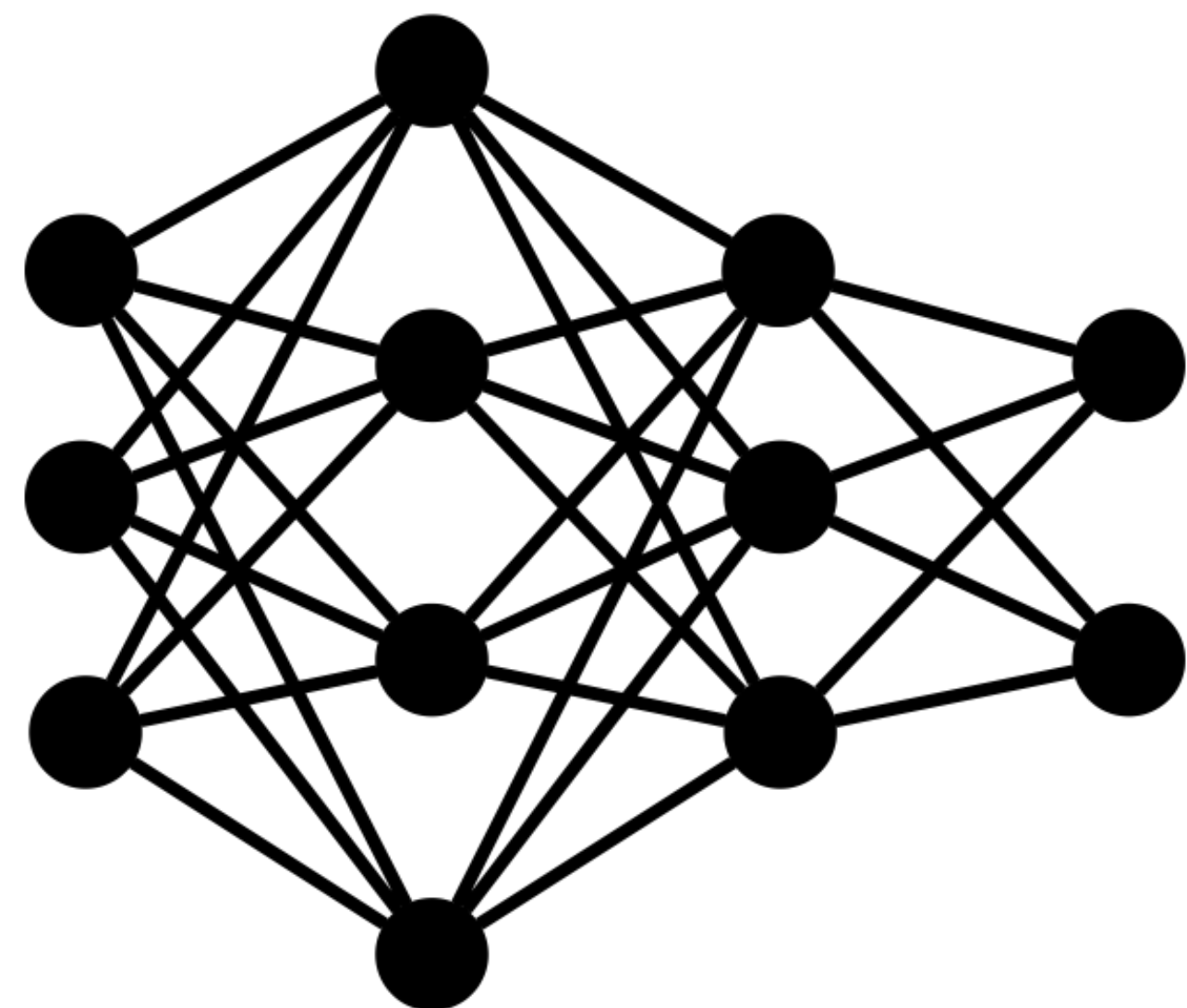
# Mind the Gap: Attainable Data Movement and Operational Intensity Bounds for Tensor Algorithms

Qijing Huang, Po-An Tsai, Joel S Emer, Angshuman Parashar

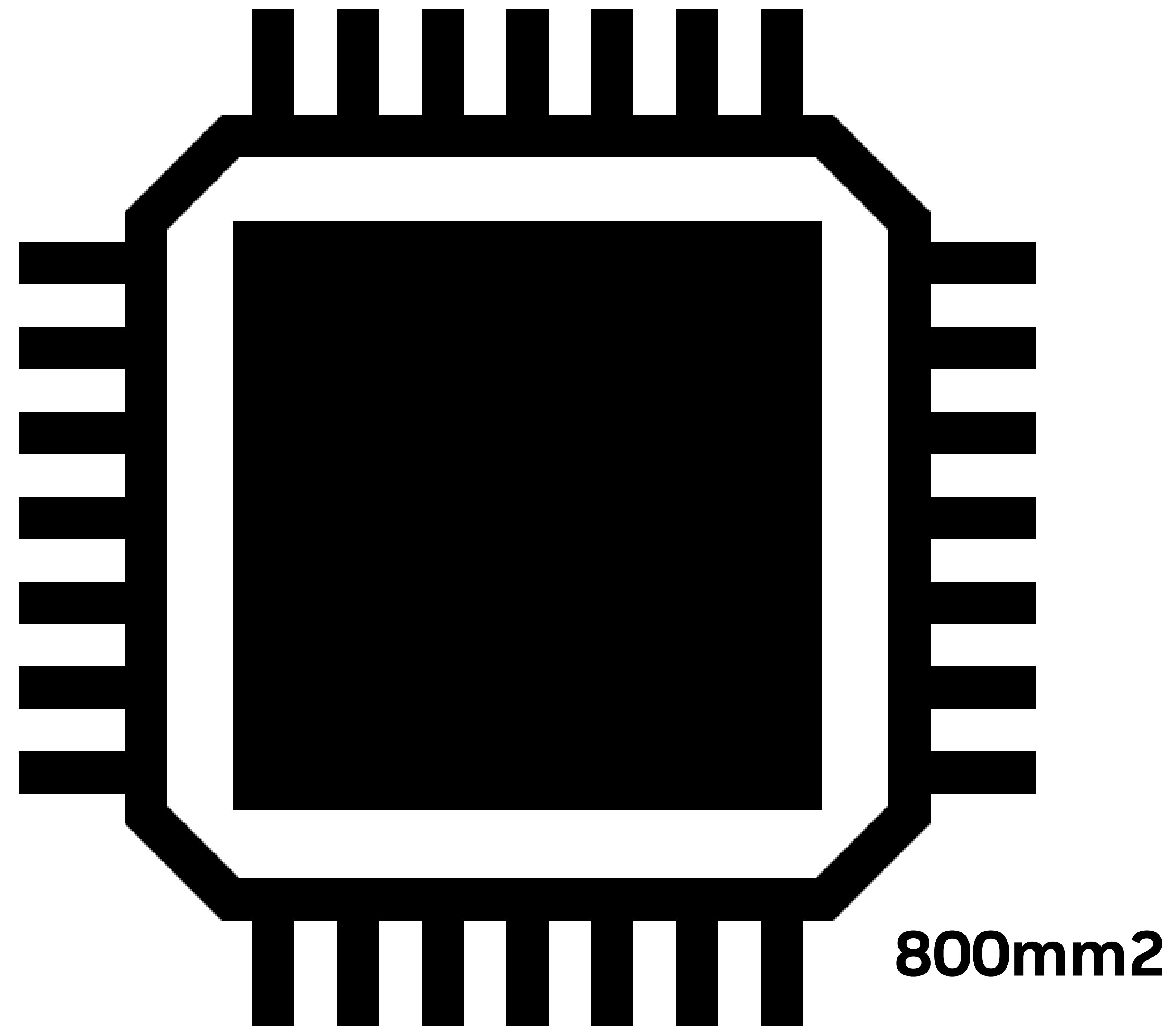
NVIDIA, MIT CSAIL

# Motivation: A design challenge

Algorithm

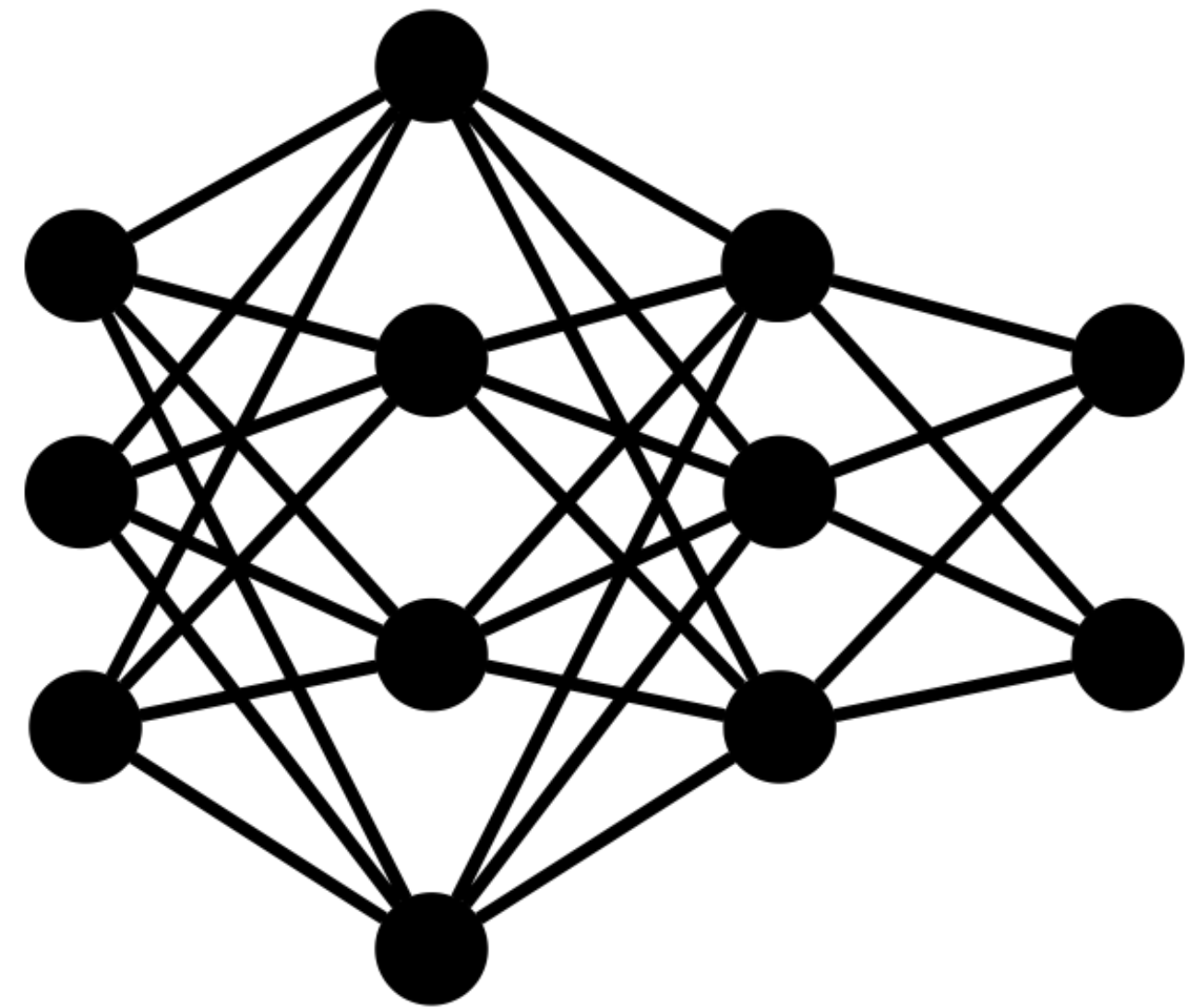


**GPTx**

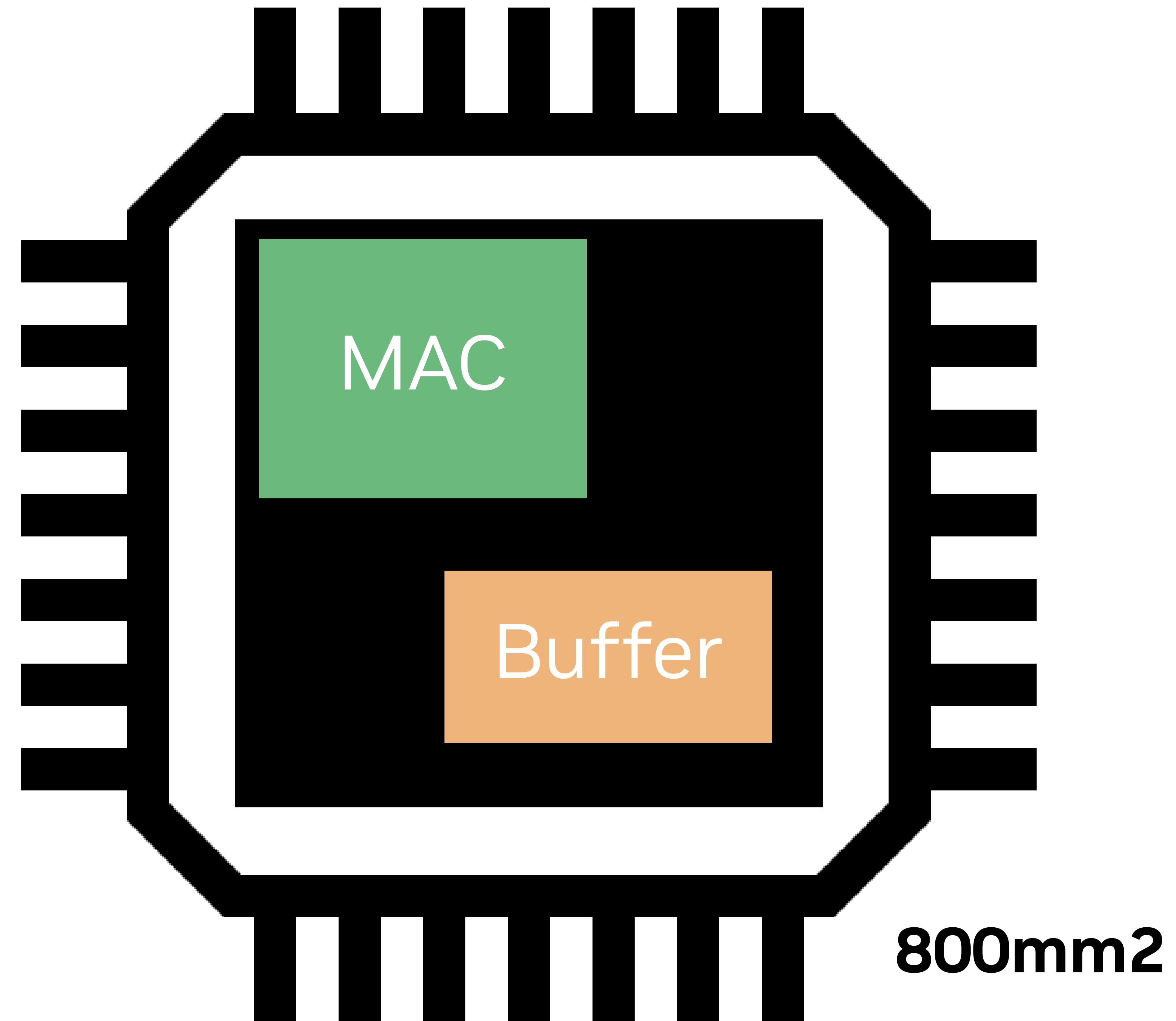


# Motivation: A design challenge

Algorithm

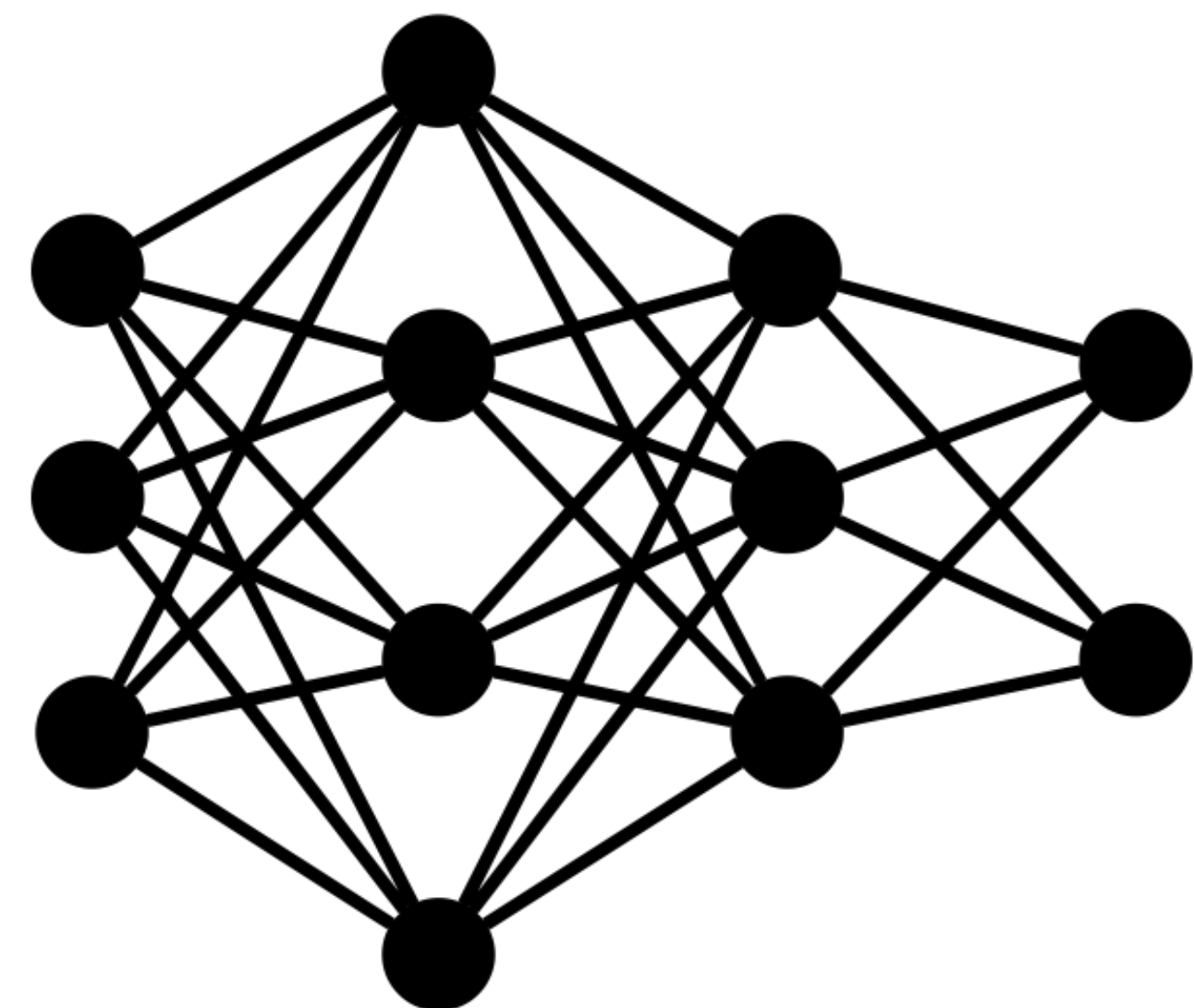


**GPTx**

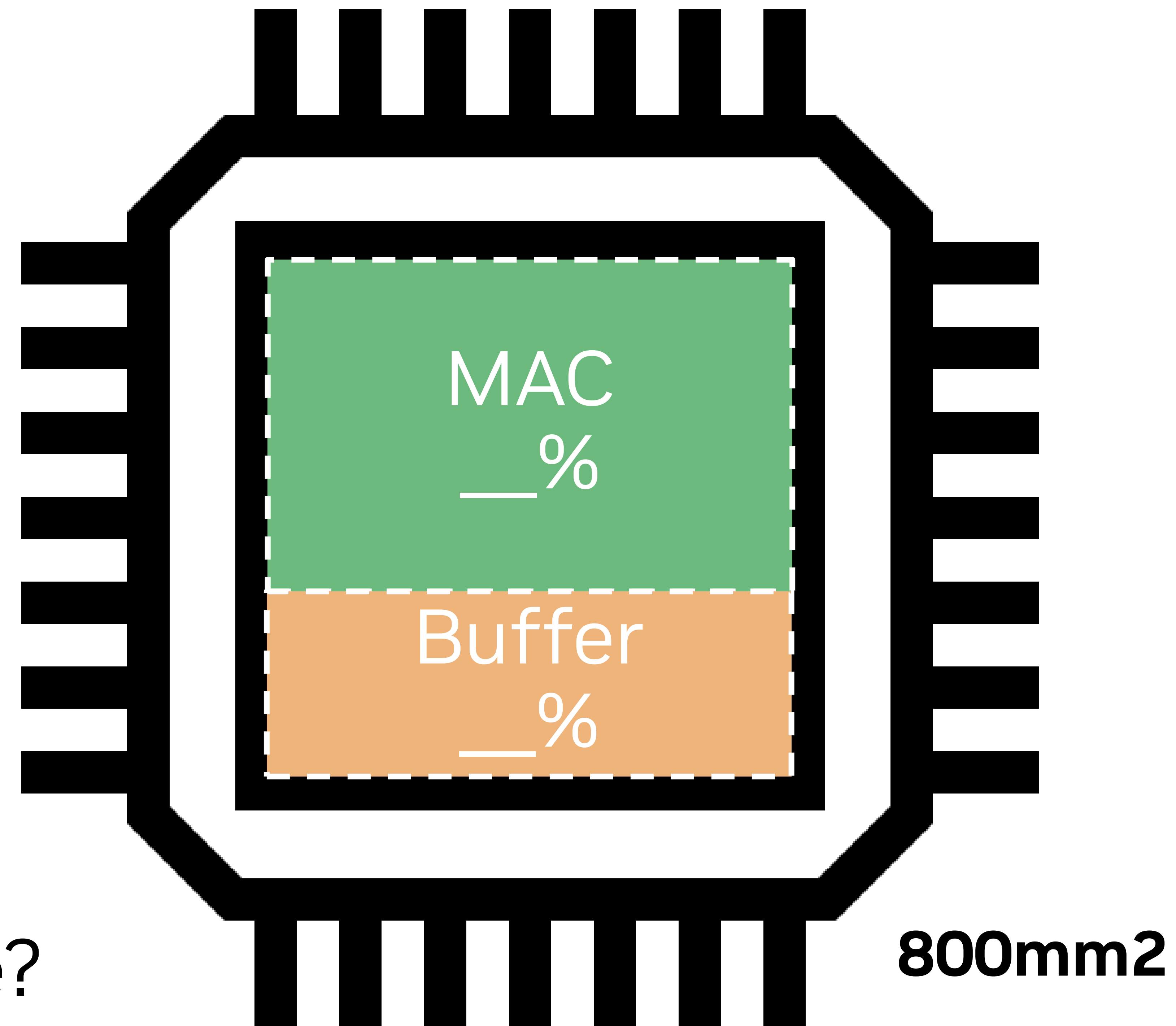


# Motivation: A design challenge

Algorithm

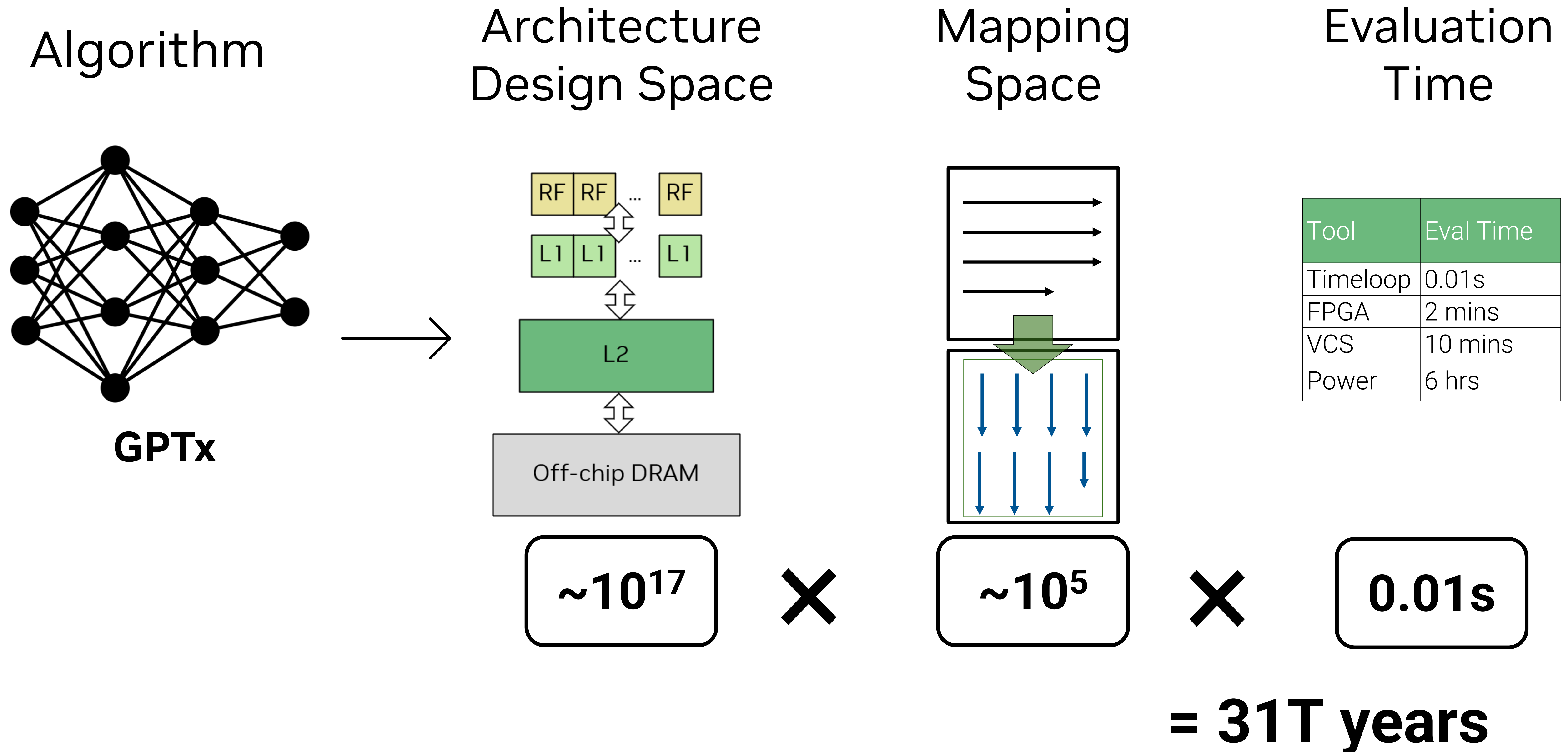


**GPTx**



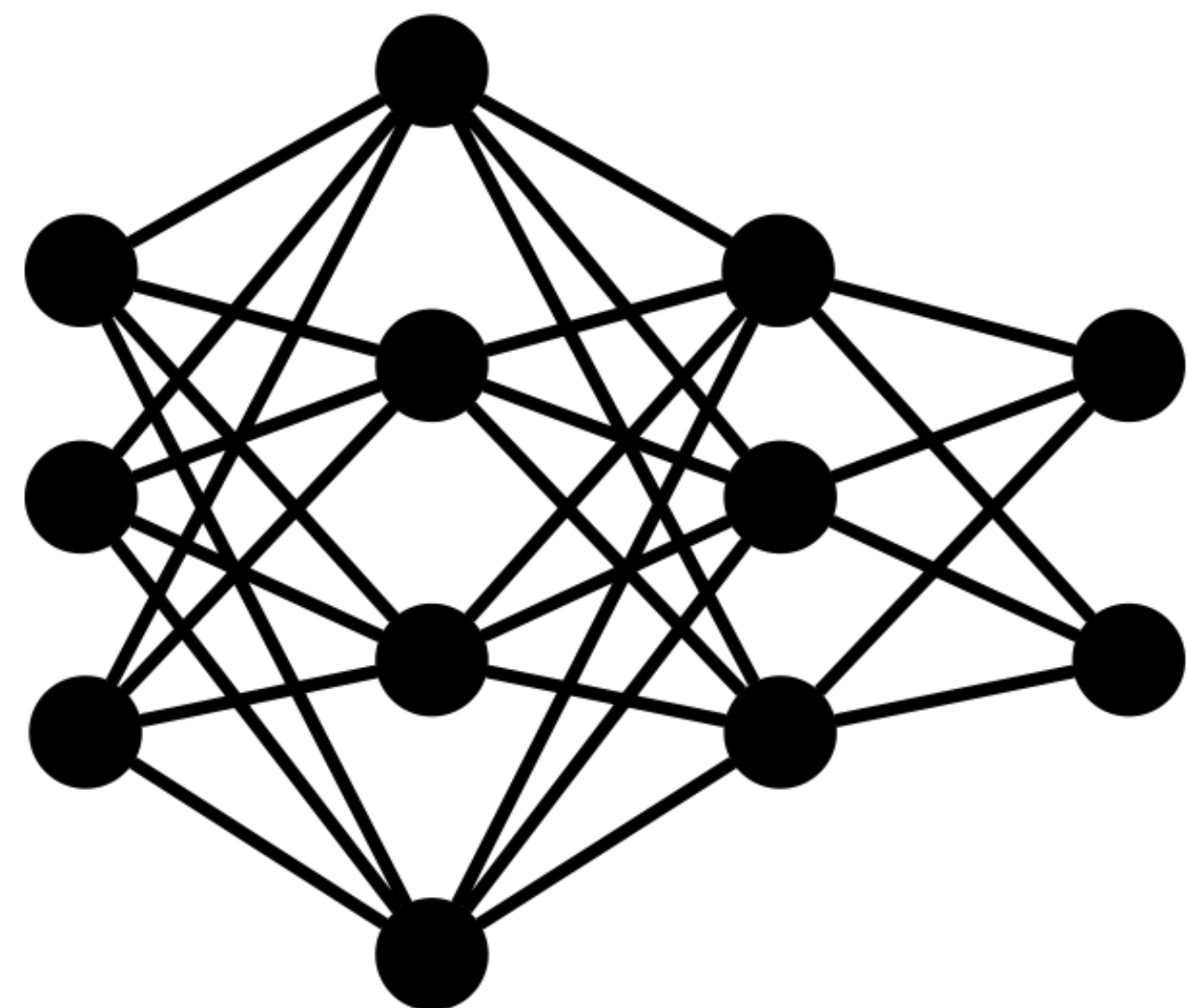
How to provision chip area  
between storage and compute?

# Approach 1: Design space exploration



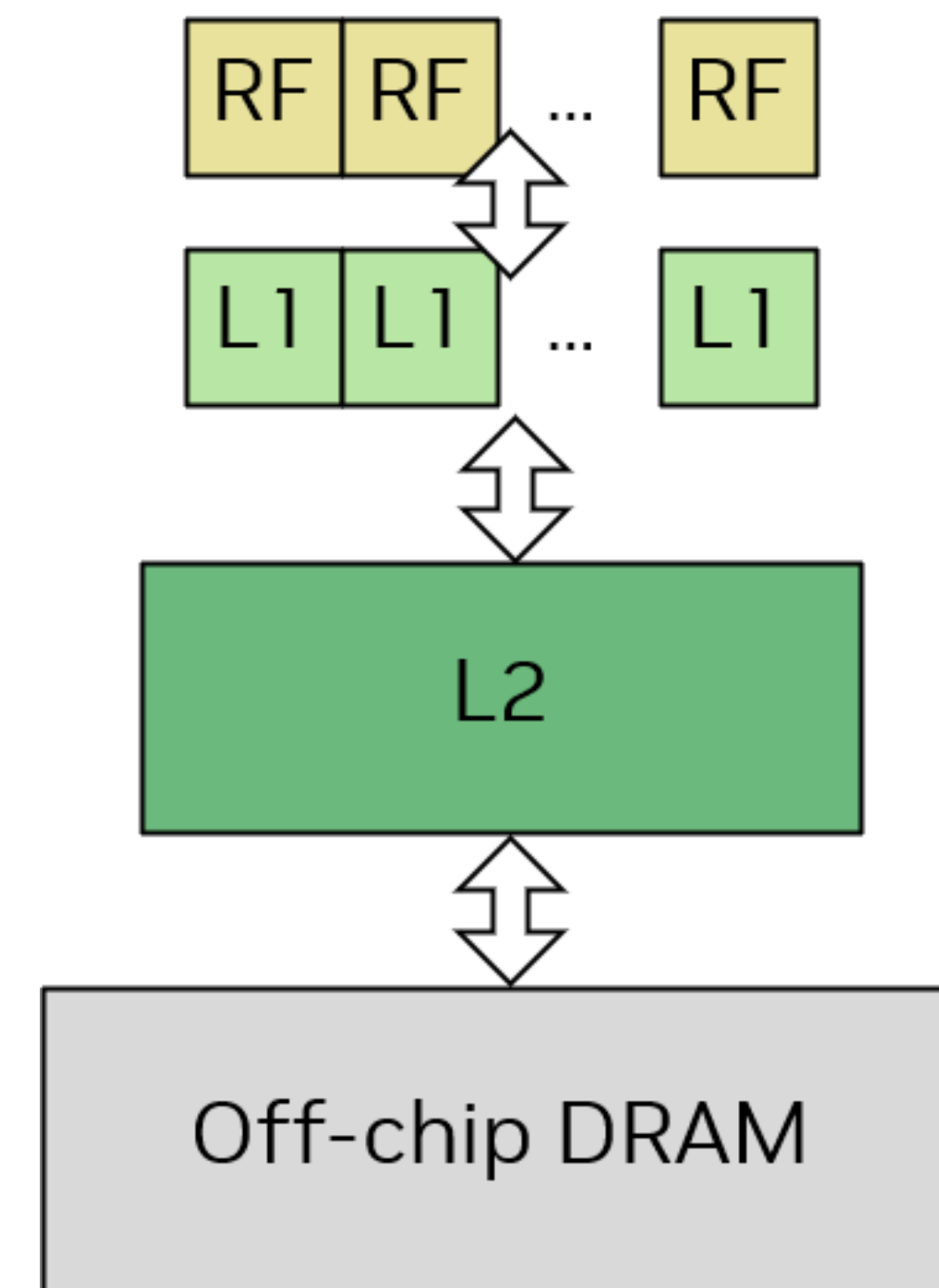
# Approach 1: Design space exploration

Algorithm

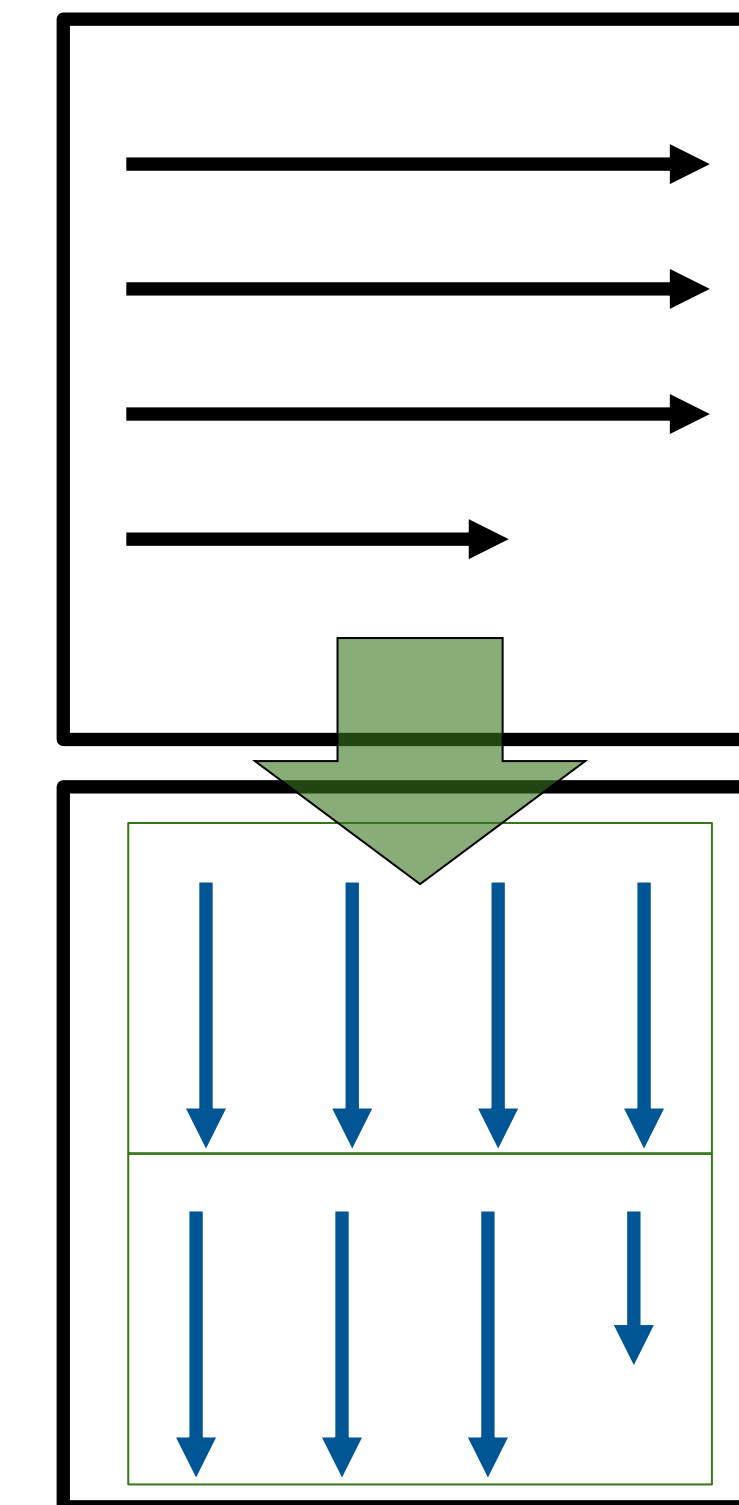


**GPTx**

Architecture Design Space



Mapping Space



Evaluation Time

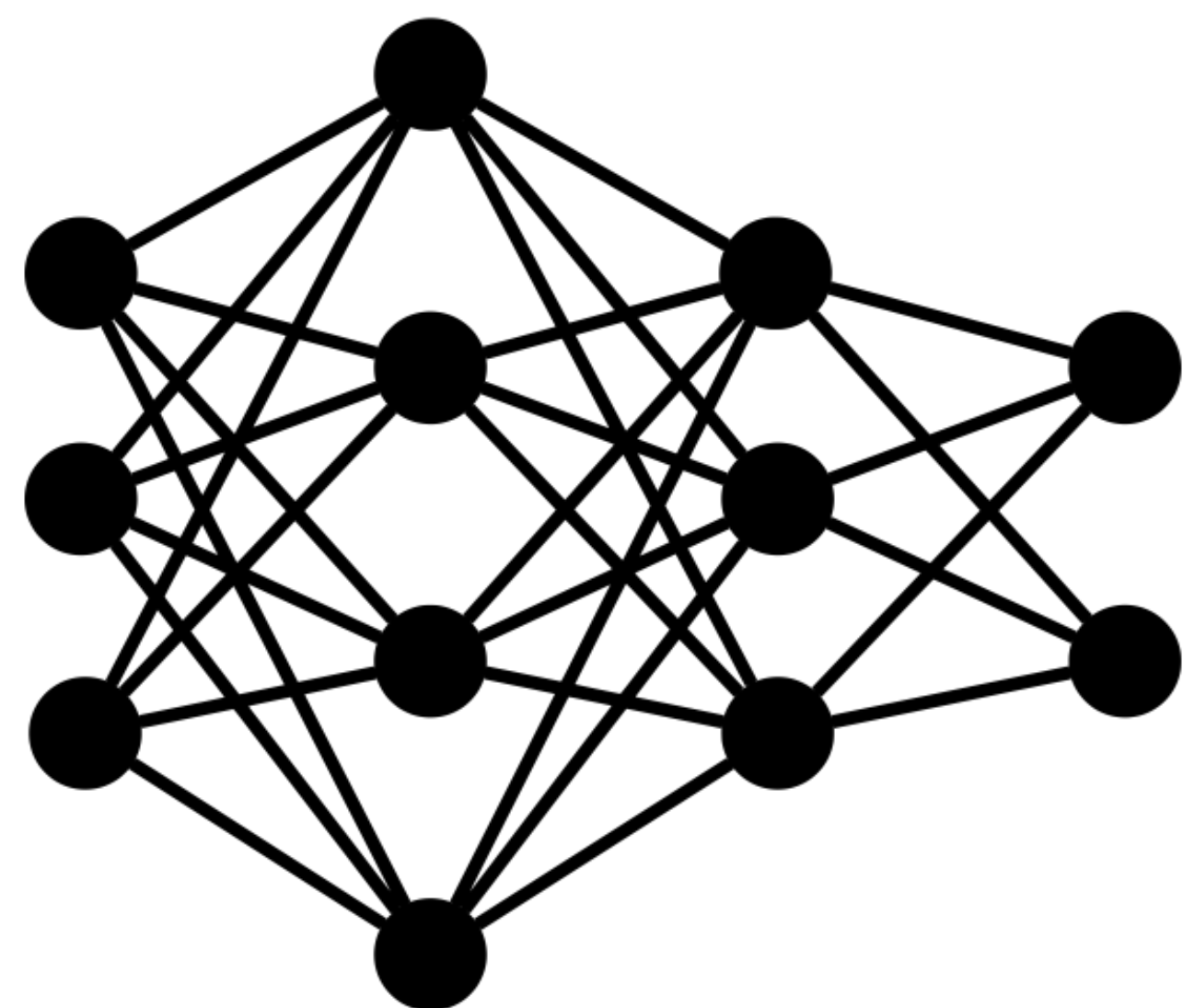
Tool	Eval Time
Timeloop	0.01s
FPGA	2 mins
VCS	10 mins
Power	6 hrs

- Time-consuming and costly
- No optimality guarantee
- Lack of design insight

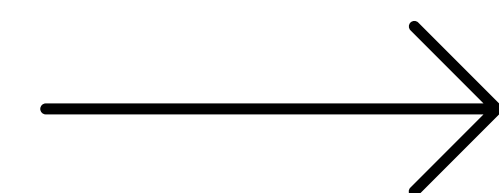
# Approach 2: Roofline model analysis

“Speeds and feeds”

Algorithm



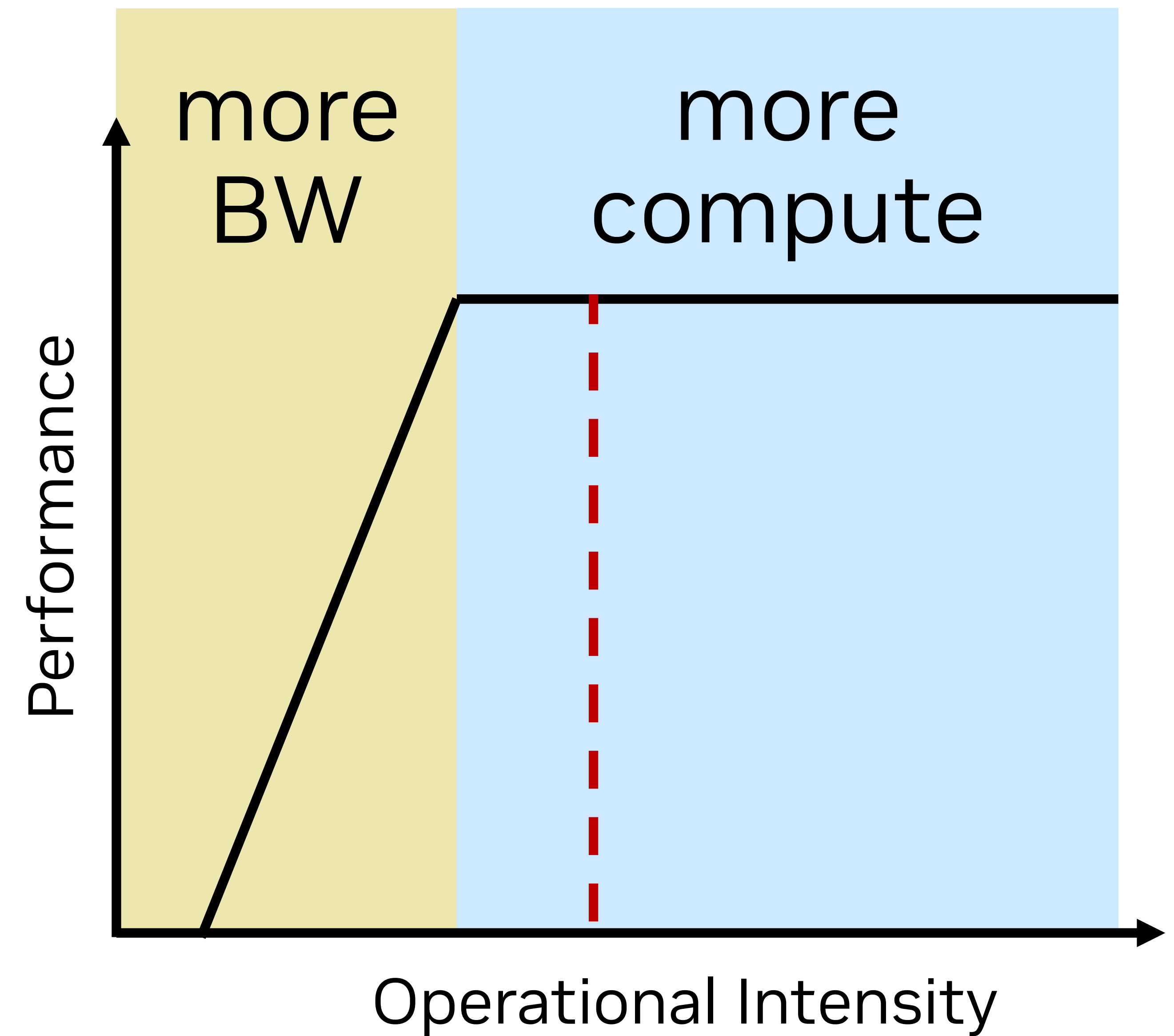
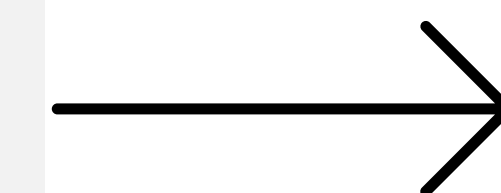
**GPTx**



**Algo max  
operational  
intensity (OI)**

=

$$\frac{\text{total compute}}{\text{algo min accesses}}$$



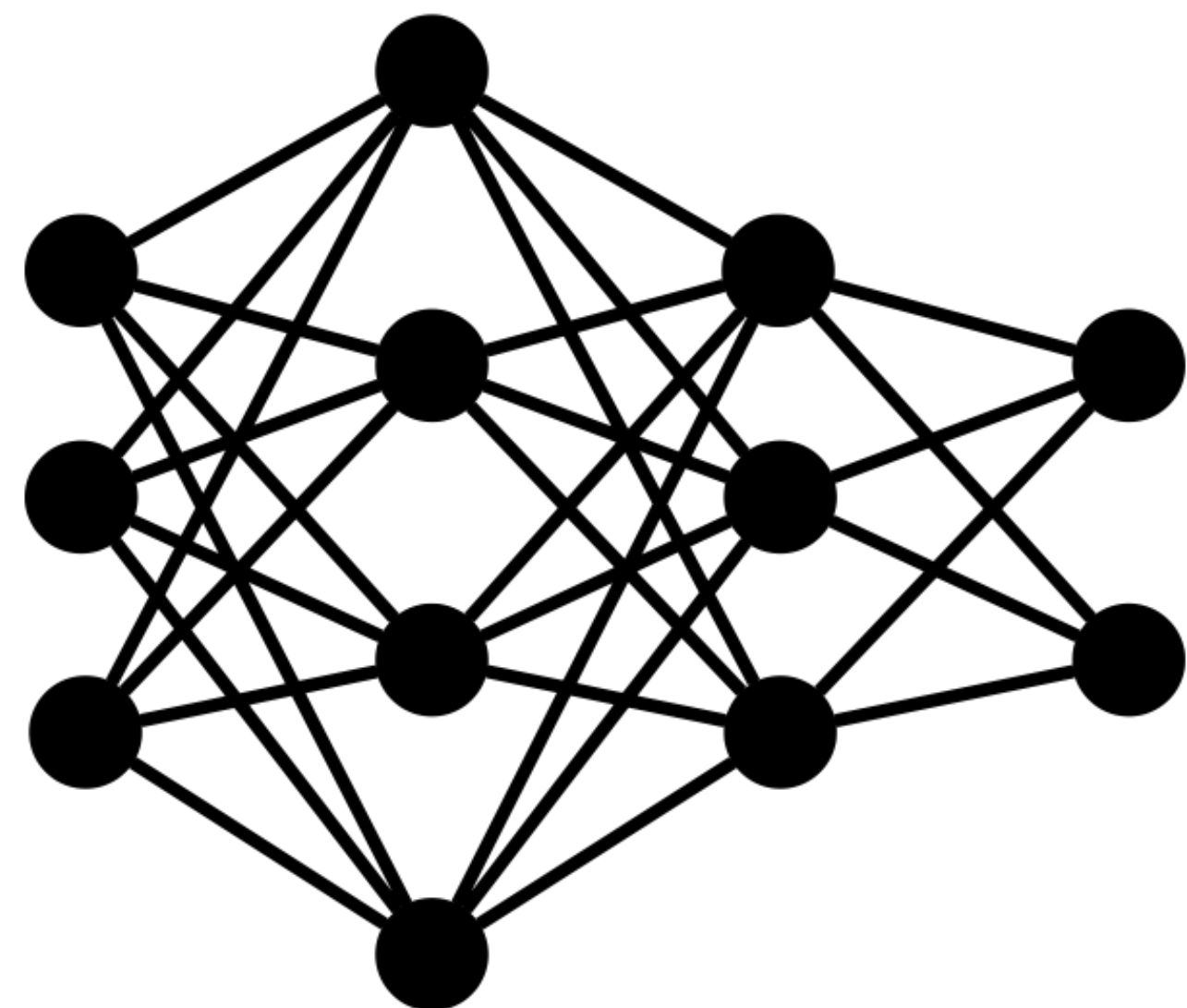
Operational Intensity

**Roofline Model**

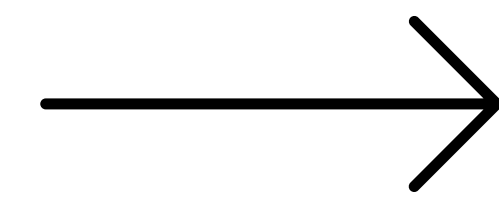
# Approach 2: Roofline model analysis

“Speeds and feeds”

Algorithm



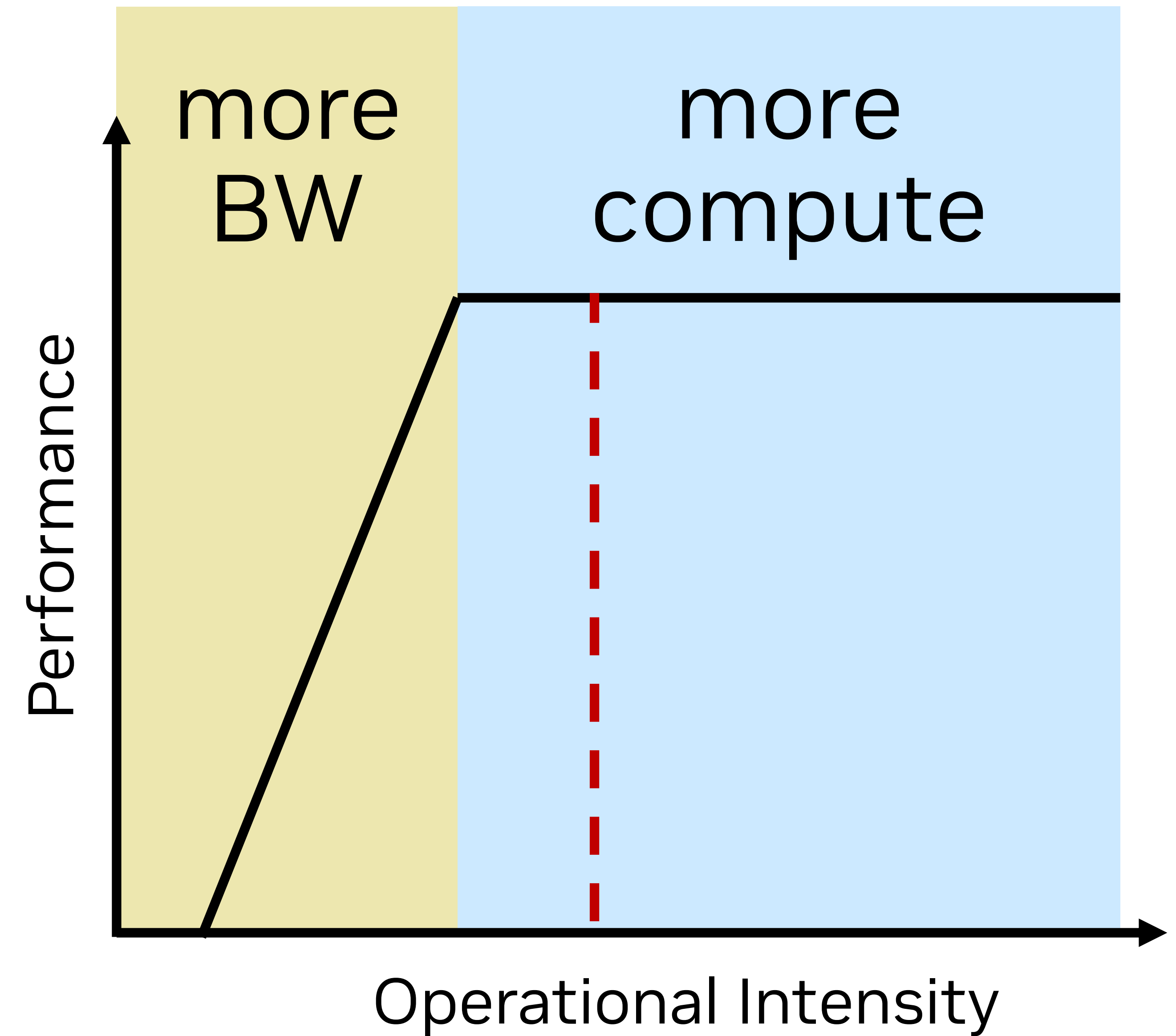
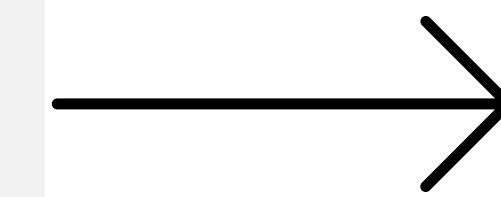
**GPTx**



**Algo max  
operational  
intensity (OI)**

=

$$\frac{\text{total compute}}{\text{algo min accesses}}$$



Operational Intensity

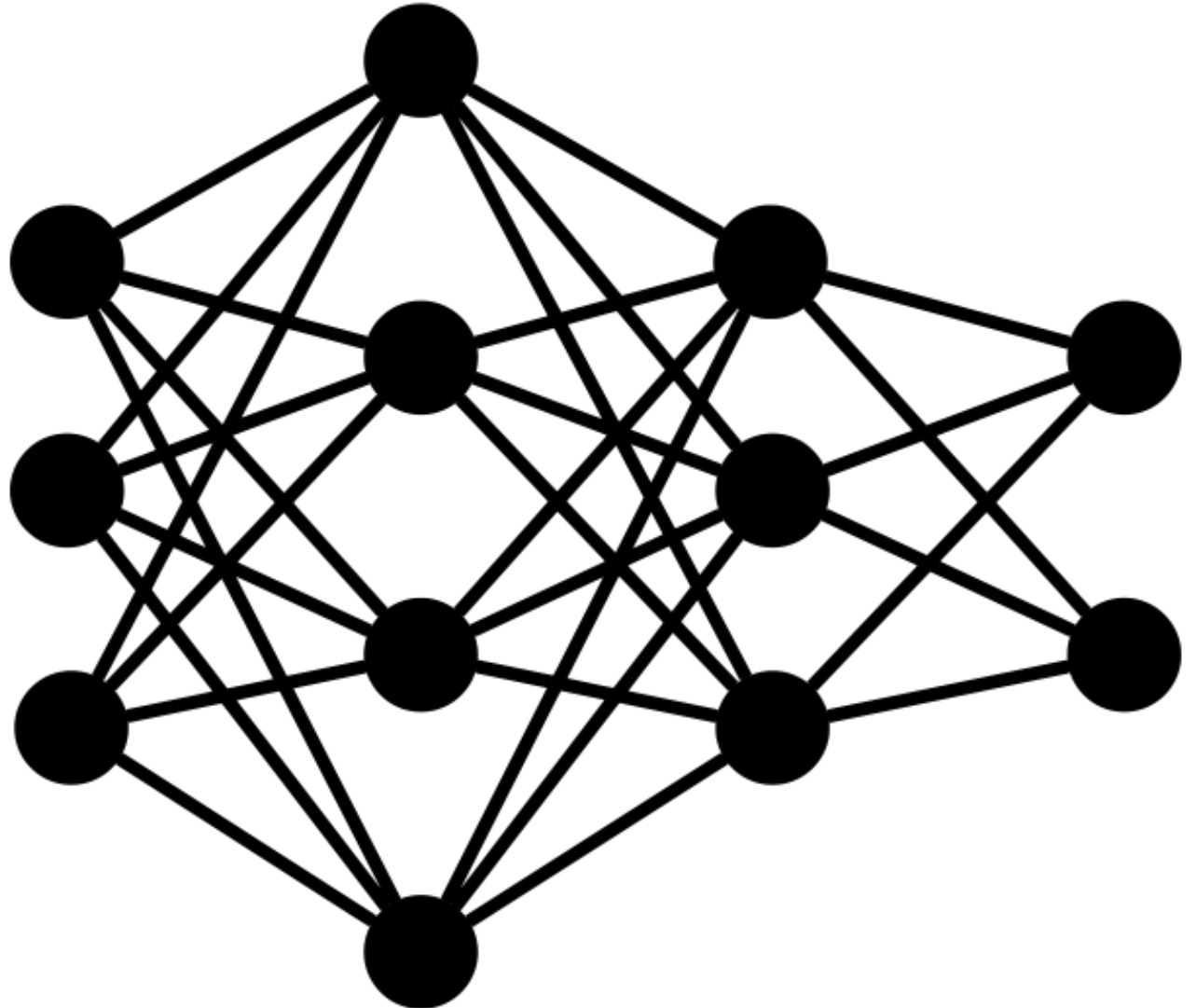
**Roofline Model**

- No buffer storage tradeoffs are present in the analysis

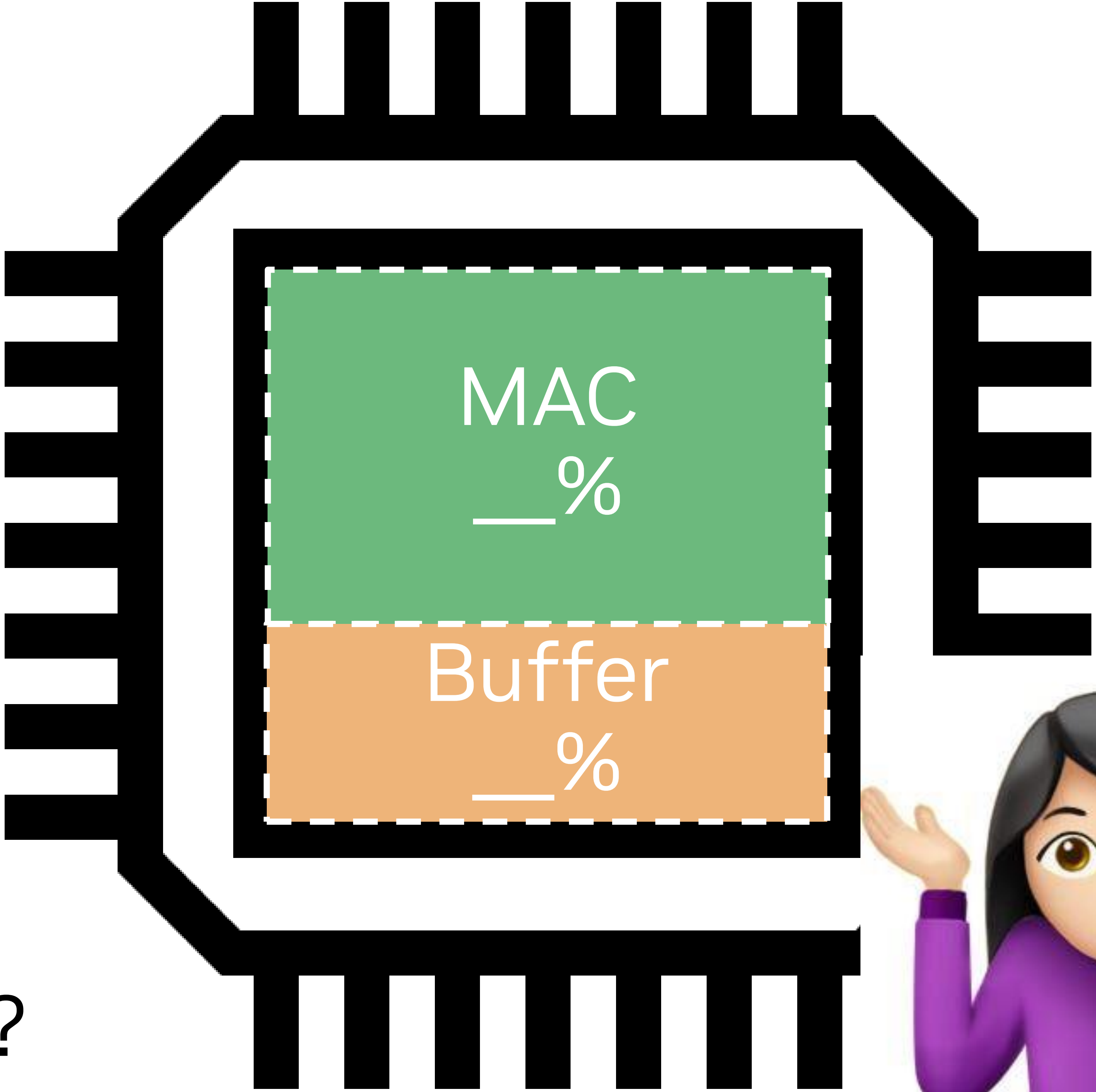


# Motivation: Lack of design tools

Algorithm



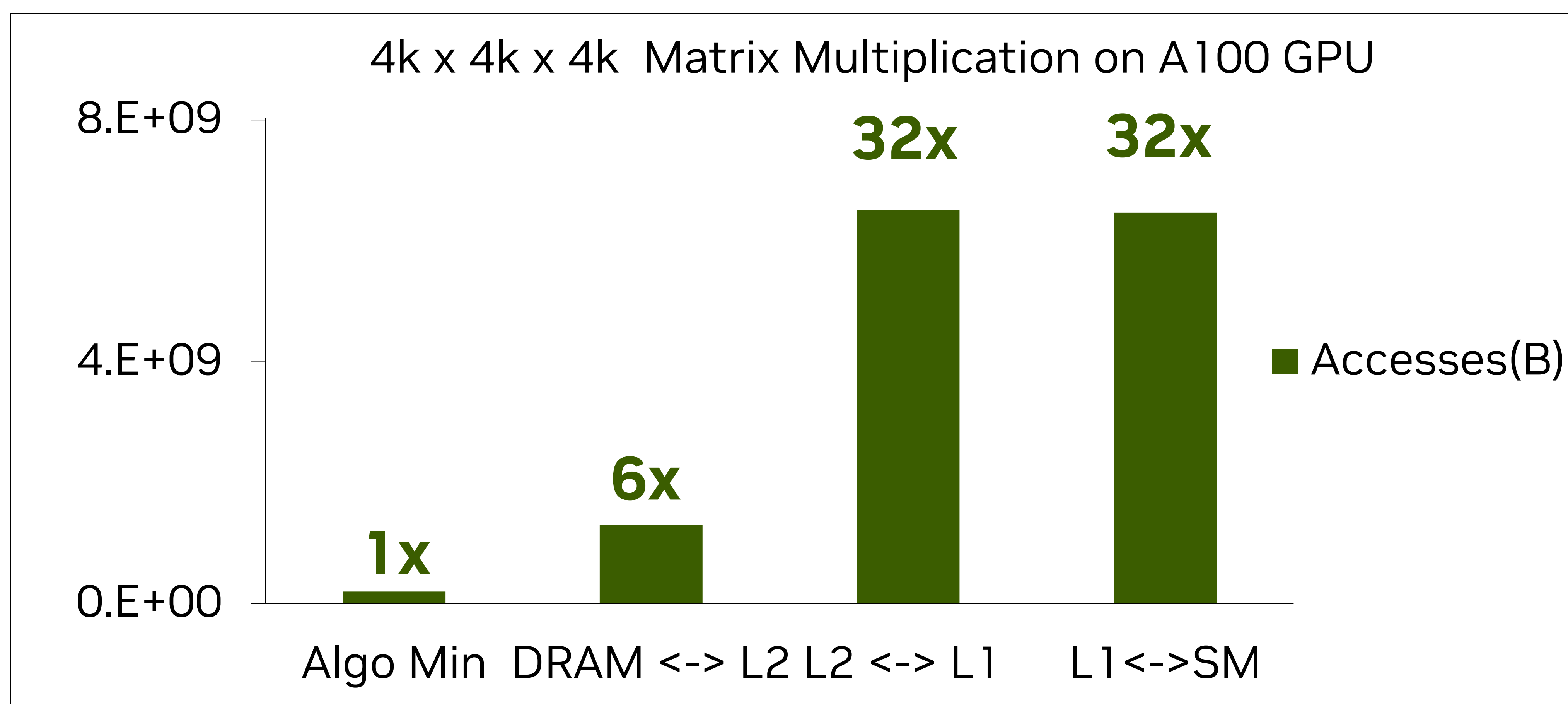
GPTx



How to provision chip area between storage and compute?

# What is missing?

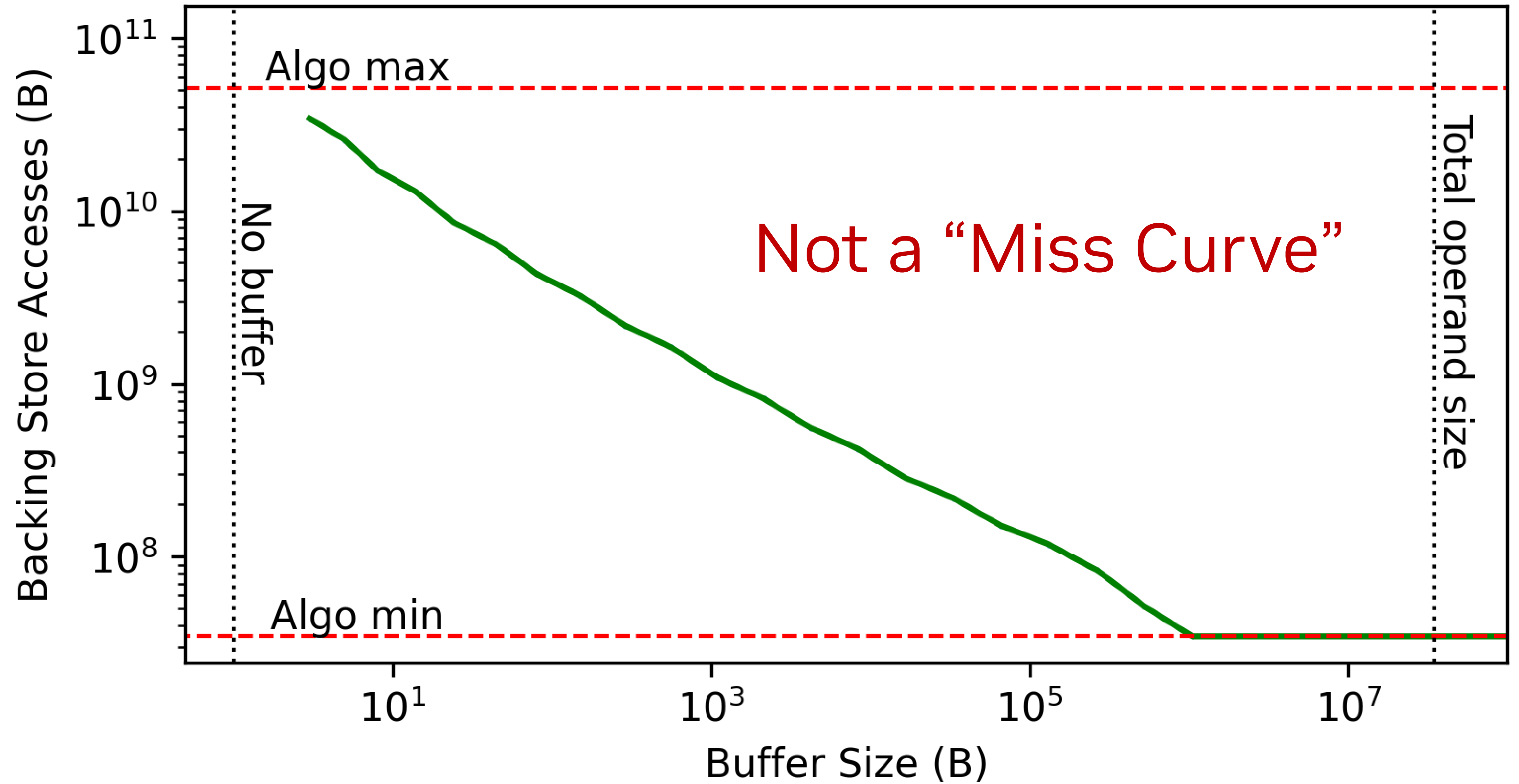
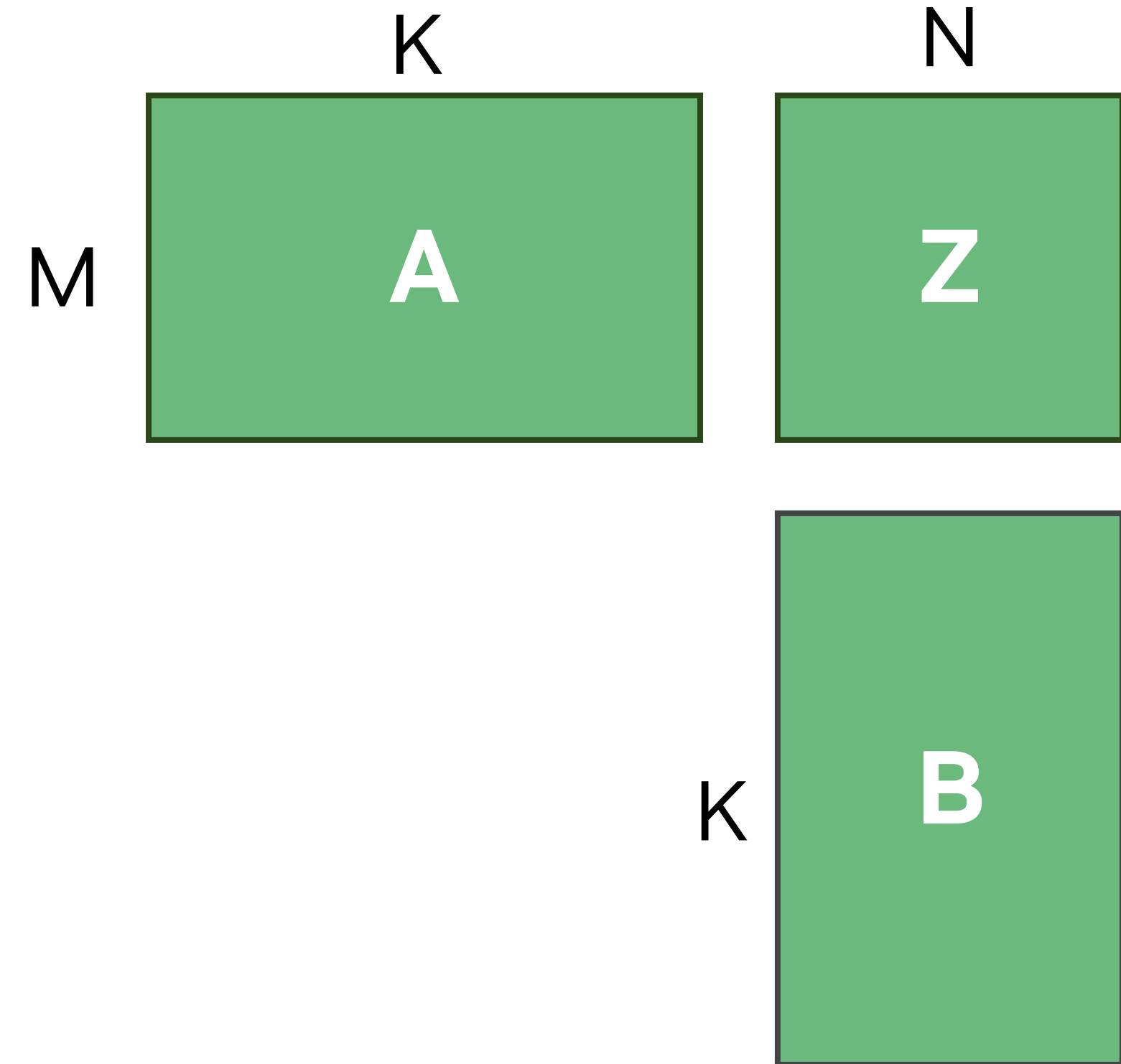
- The workload **does not** always operate with *algorithmic minimal accesses*, or equivalently, *algorithmic maximal OI*.



- Actual *backing-store accesses* and *OI* depend on the **mapping** and **buffer sizes**.

# A desirable data movement bound

Matrix Multiplication  
(GEMM) Einsum:



$$Z_{m,n} = A_{m,k} B_{k,n}$$

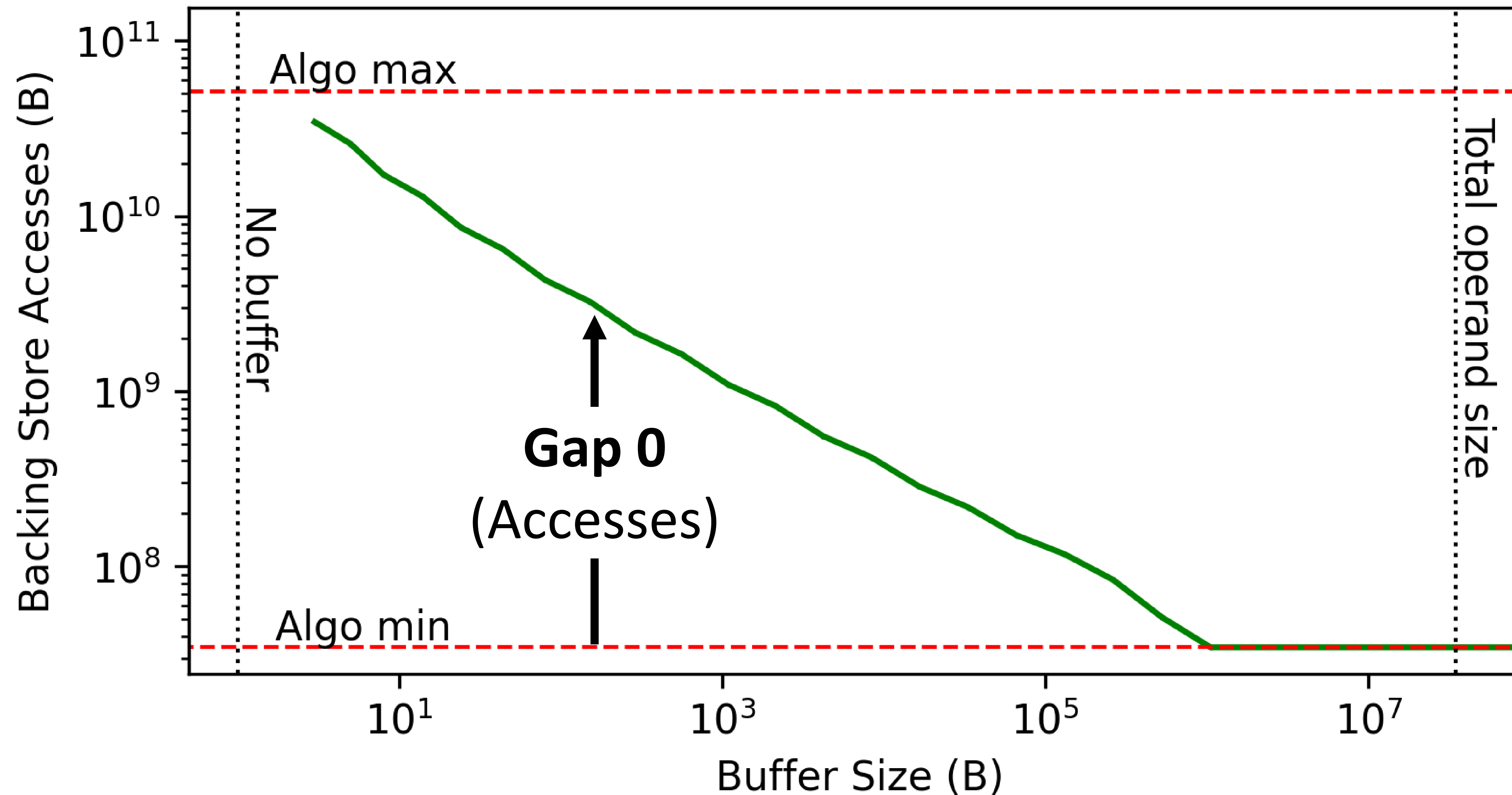
M – output row dim

K – reduction dim

N – output column dim

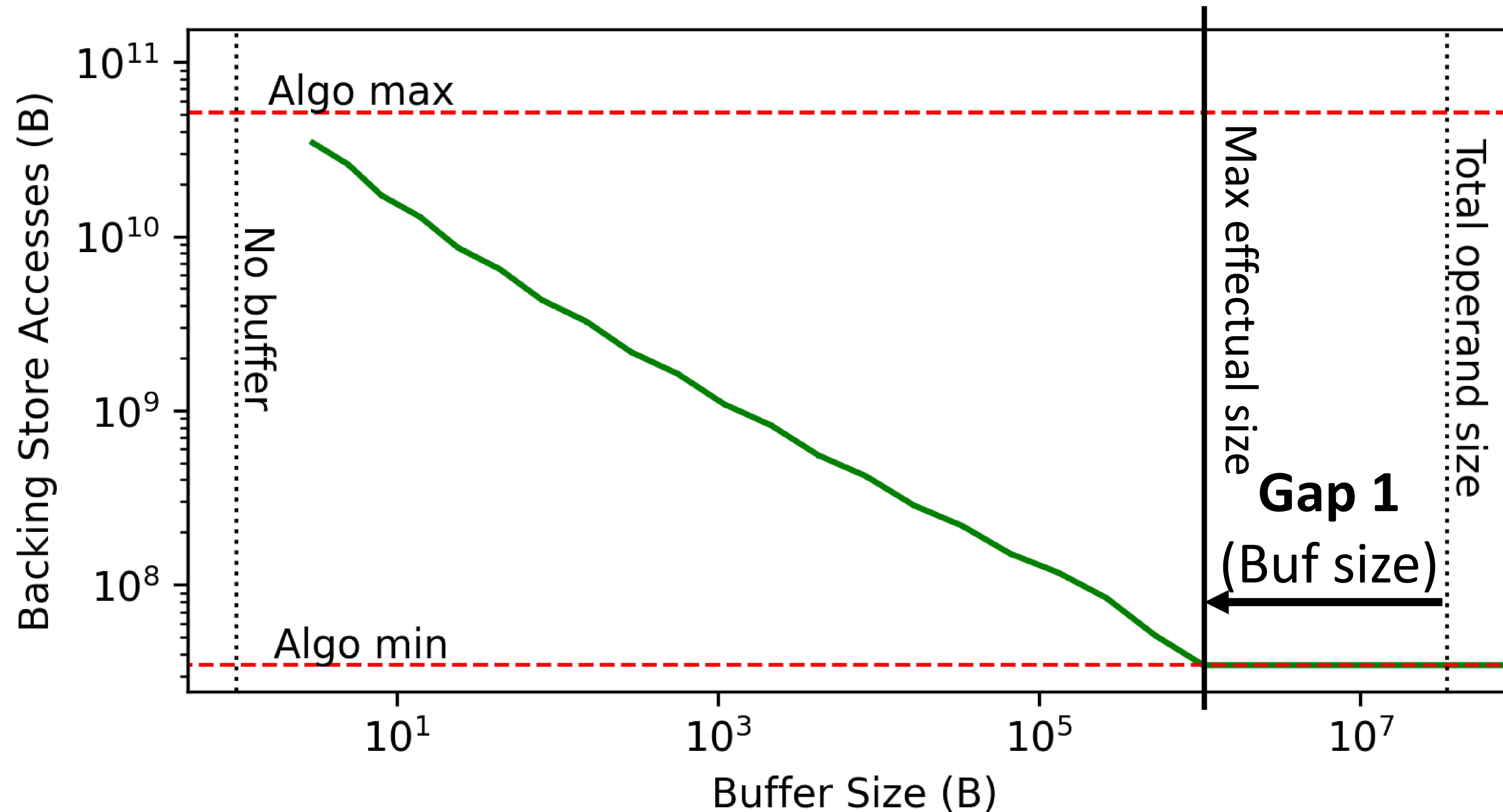
“Ski-slope Diagram”

# Mind the gap: key design questions



**[Gap 0]** Given a buffer capacity, what is the minimal attainable *data access count*?

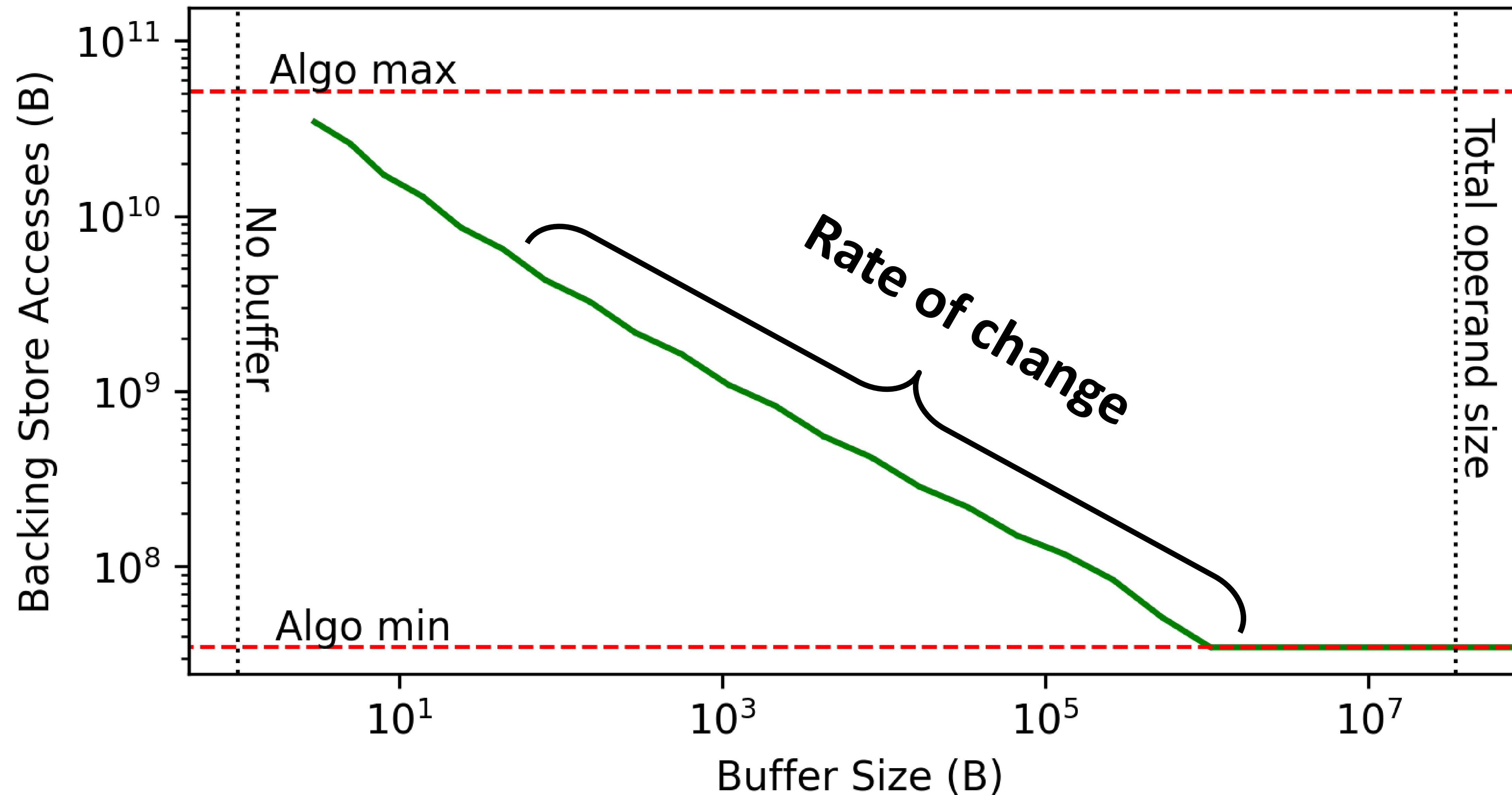
# Mind the gap: key design questions



**[Gap 0]** Given a buffer capacity, what is the minimal attainable *data access count*?

**[Gap 1]** How much buffer capacity is required to achieve min data movement?

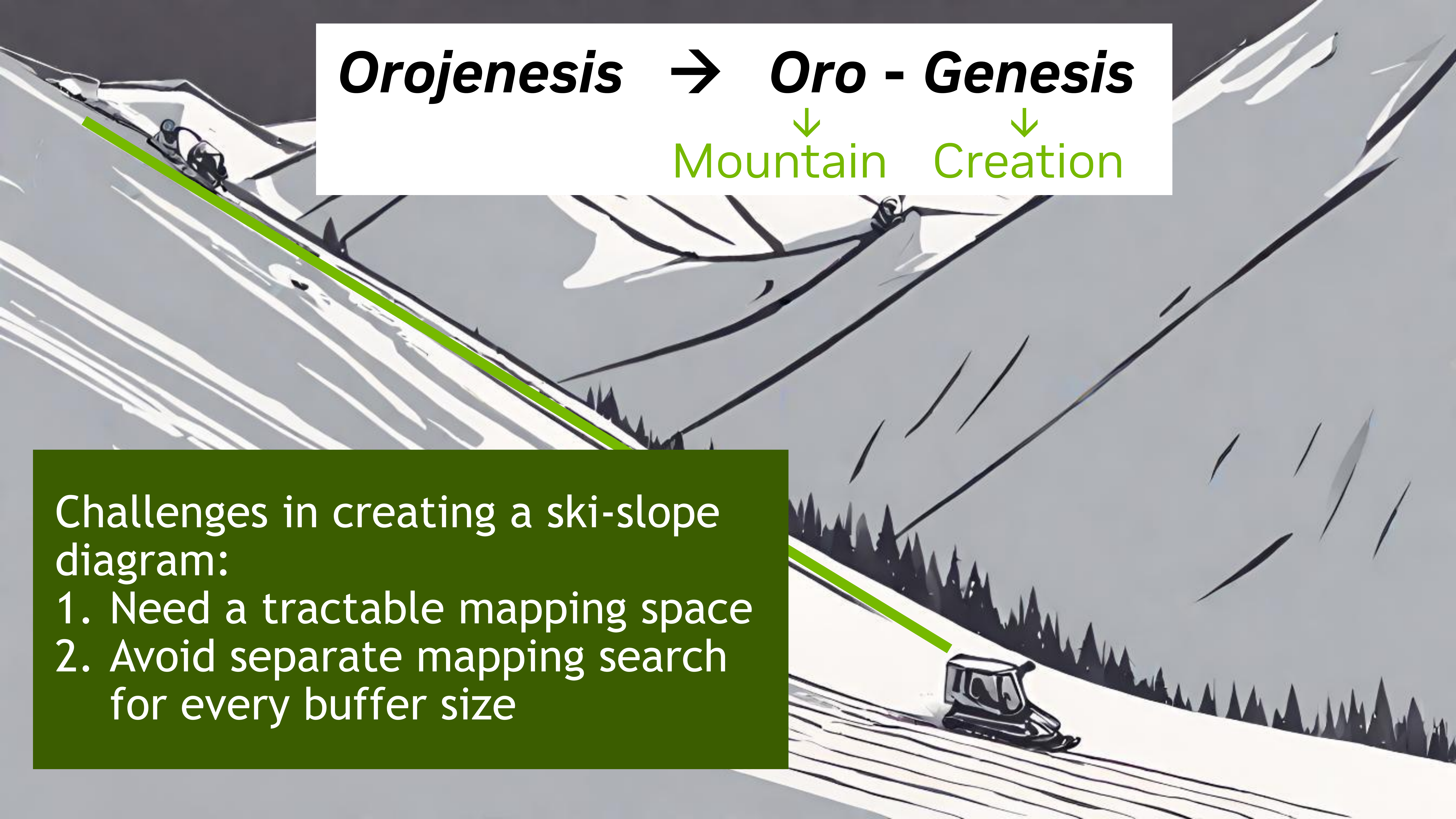
# Mind the gap: key design questions



**[Gap 0]** Given a buffer capacity, what is the minimal attainable *data access count*?

**[Gap 1]** How much buffer capacity is required to achieve min data movement?

**[rate of change of Gap 0]** How does a workload benefit from incremental increase in buffer capacity?



***Orojenesis*** → ***Oro - Genesis***

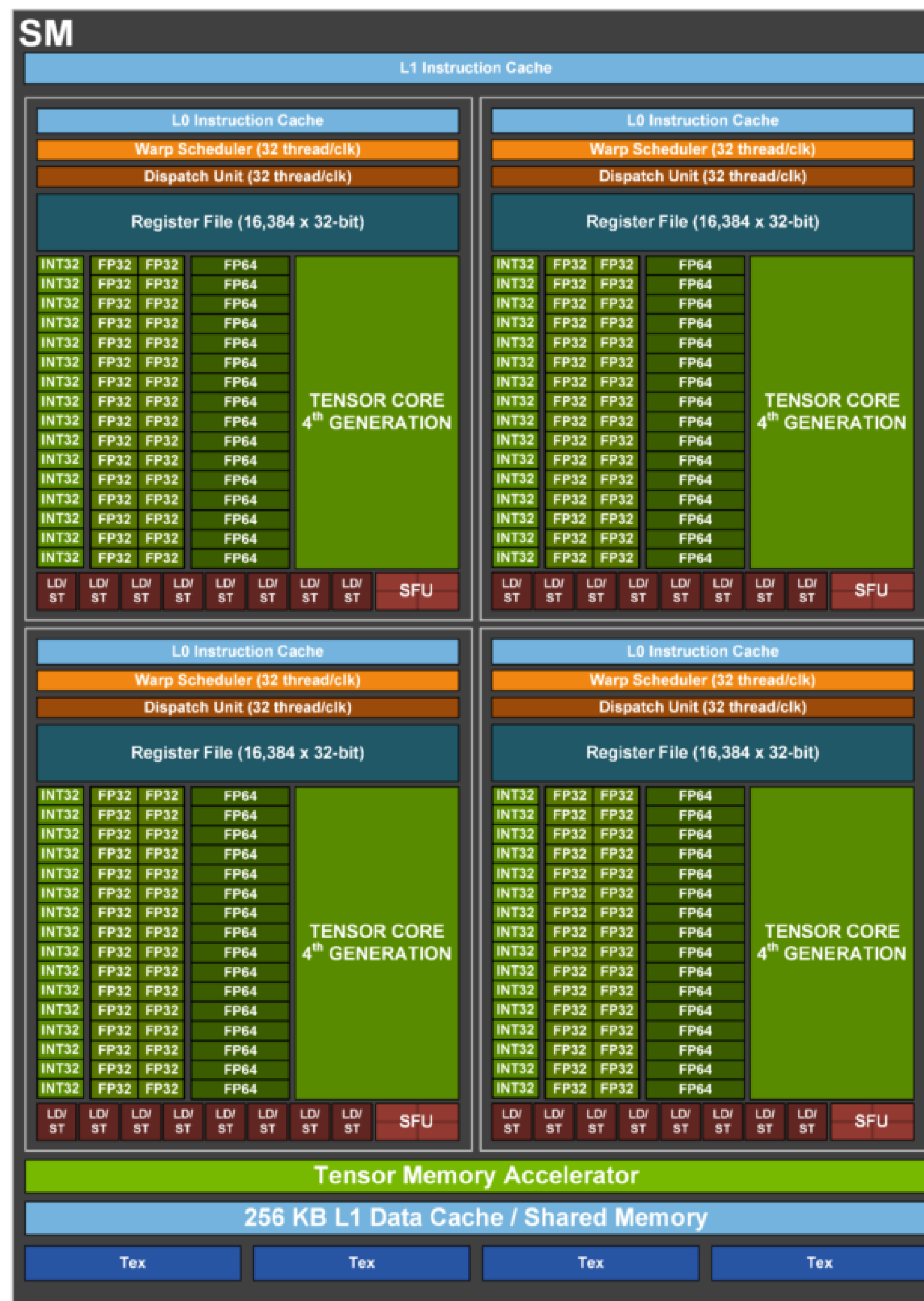
↓  
Mountain    Creation

Challenges in creating a ski-slope diagram:

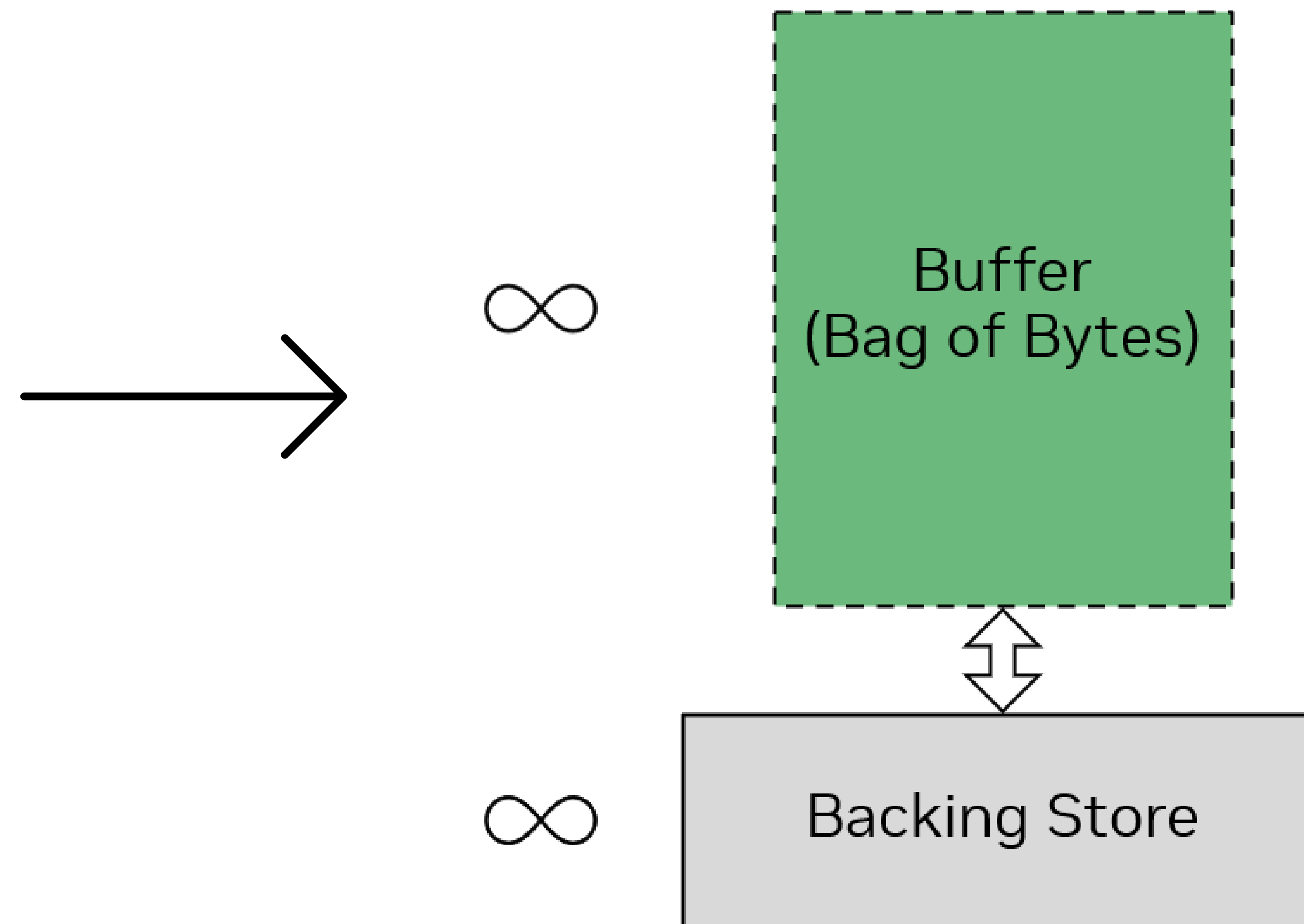
1. Need a tractable mapping space
2. Avoid separate mapping search for every buffer size

# The *Snowcat* Architecture

Enables exhaustive mapping search



Real Design



*Snowcat*  
Architecture



# The *Orojenes* Methodology

A single mapping search per workload

## Inputs

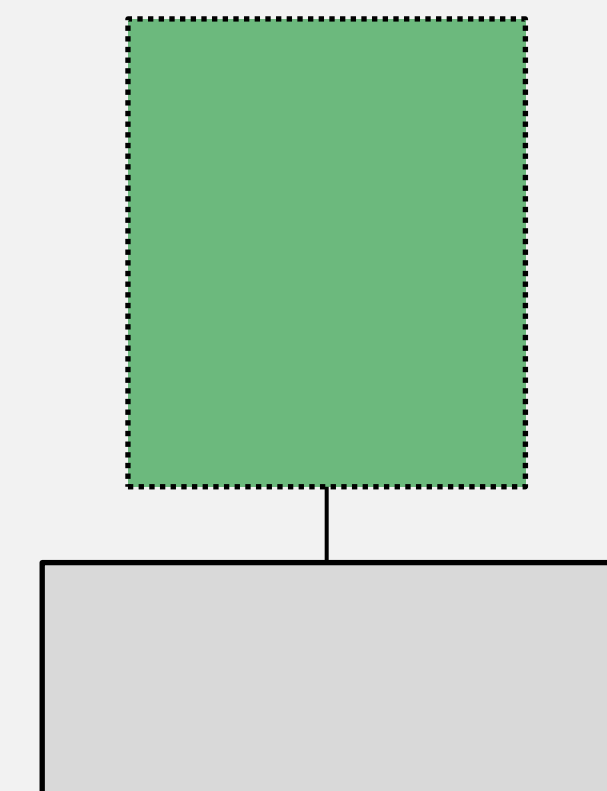
Single Einsum ■

Chain of Einsums ■ ■ ■ ■

## Exhaustive Search

### Snowcat Arch

∞



Mapping 0

Buffer Util

Accesses

Mapping 1

Buffer Util

Accesses

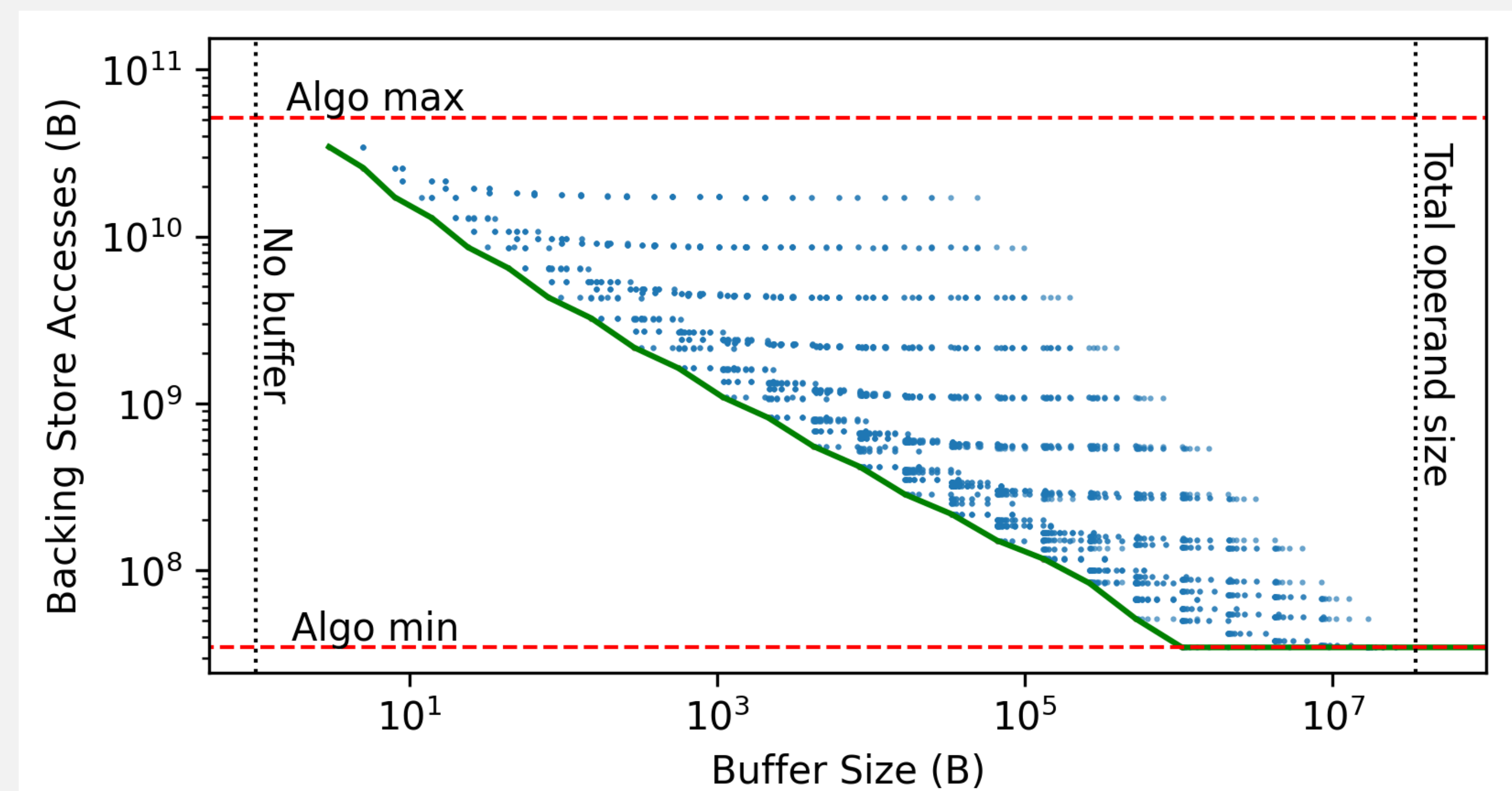
...

Mapping N

Buffer Util

Accesses

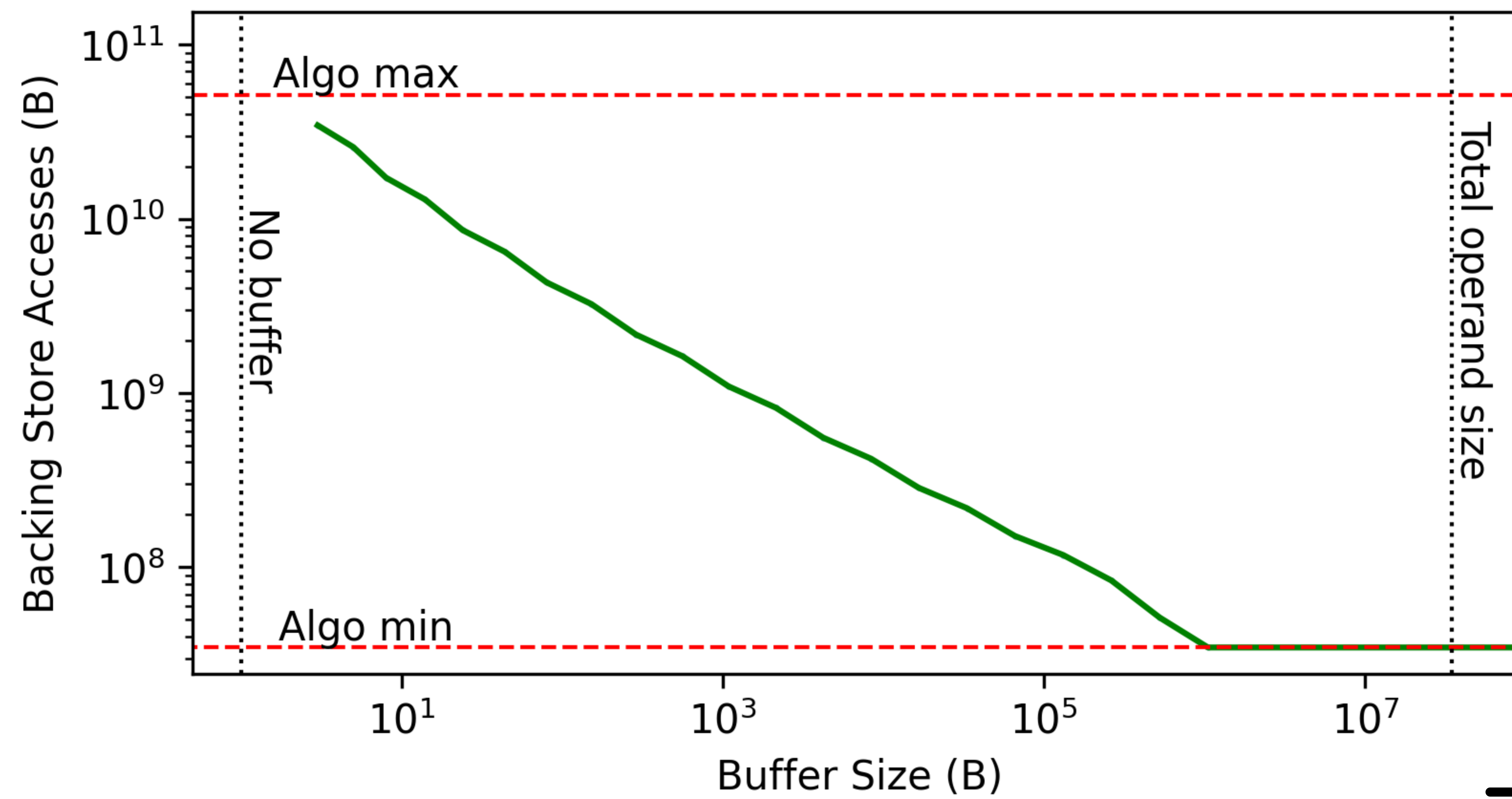
## Ski-slope Diagram



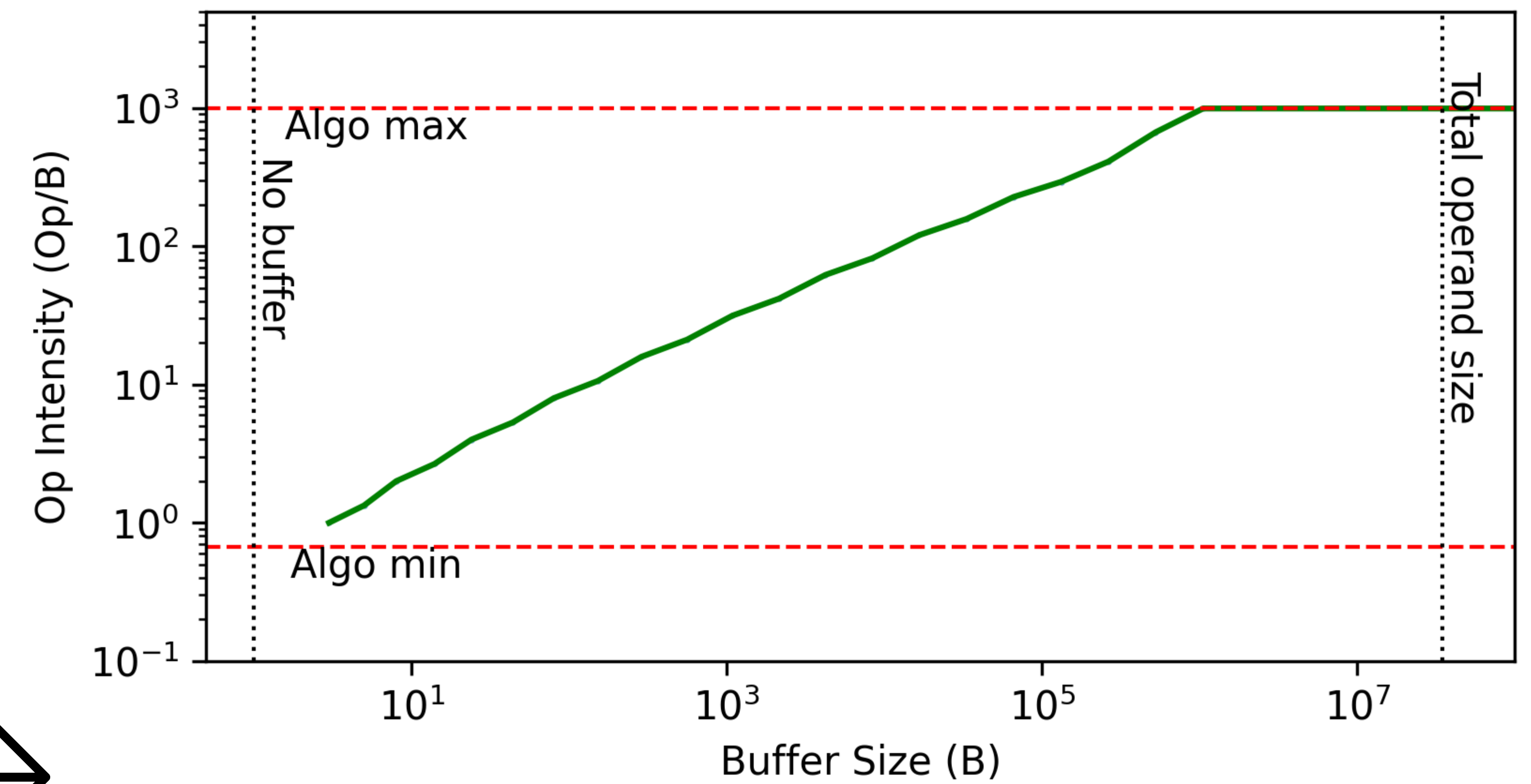
$\min(\text{buffer util, accesses})$

# OI Bound Derivation

## Ski-slope Diagram



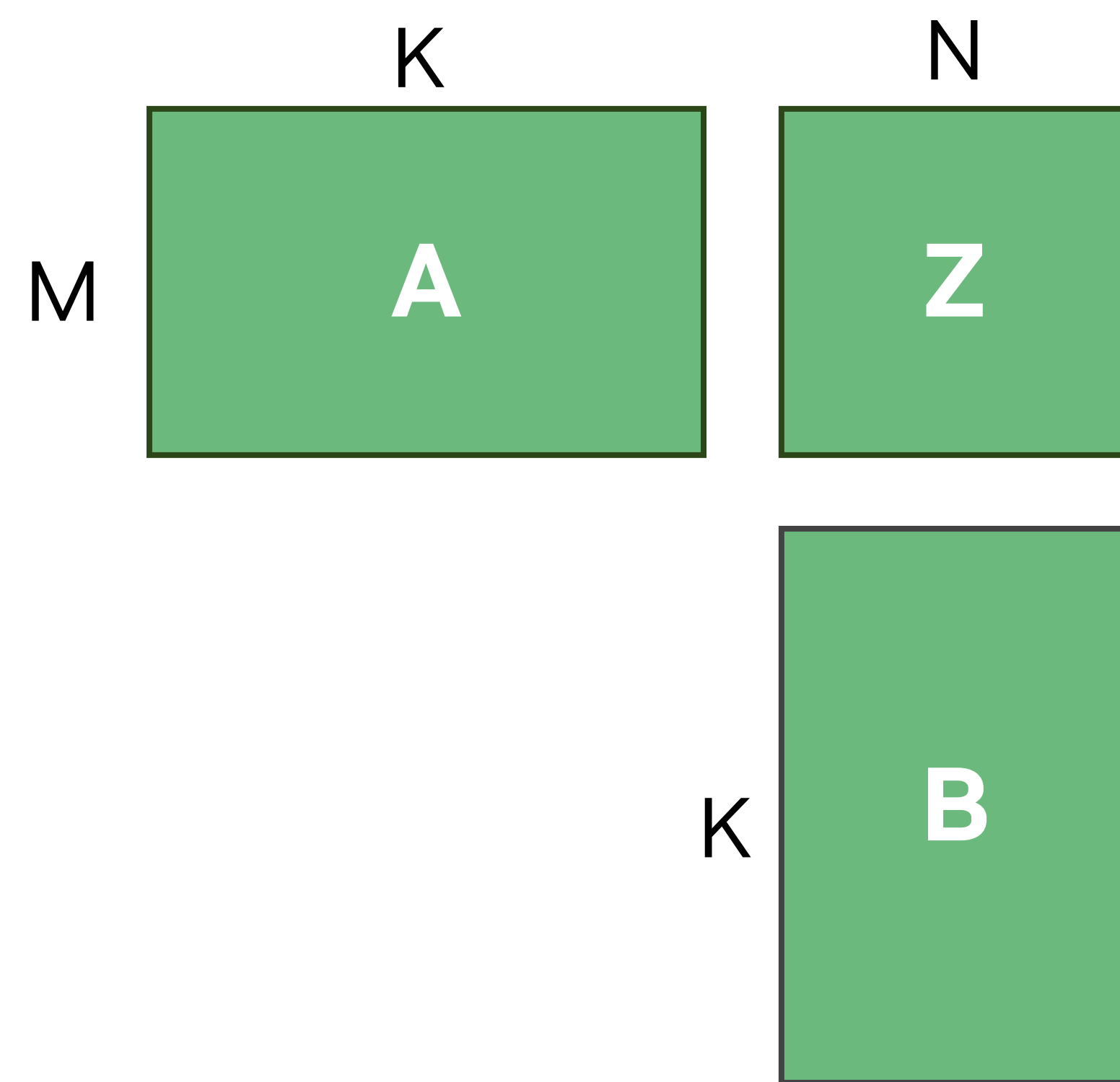
## OI Bound



→  
Inverse  
&  
Multiply by  
total operations

# Example *Orogenesis* Analysis

GEMM Einsum:



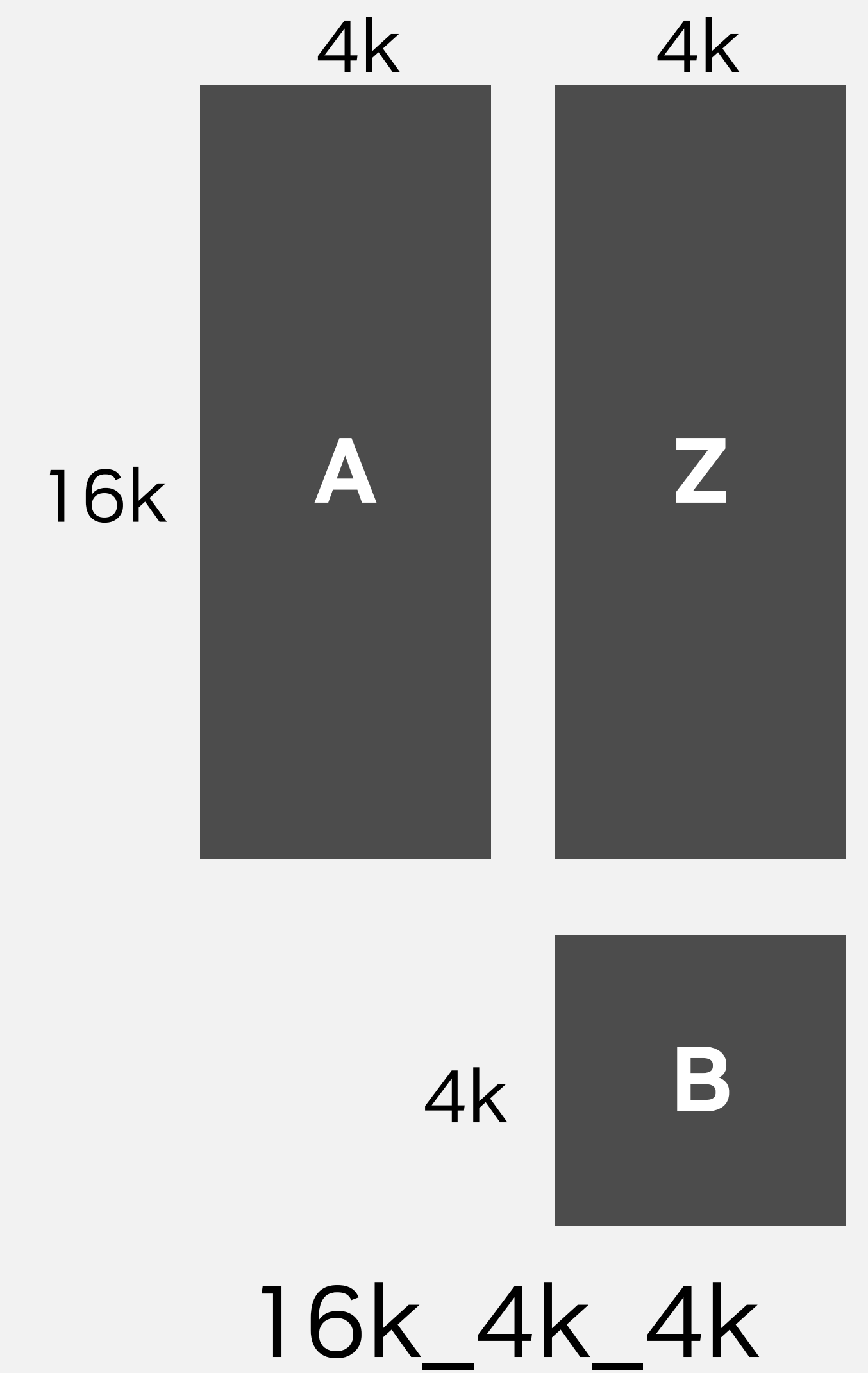
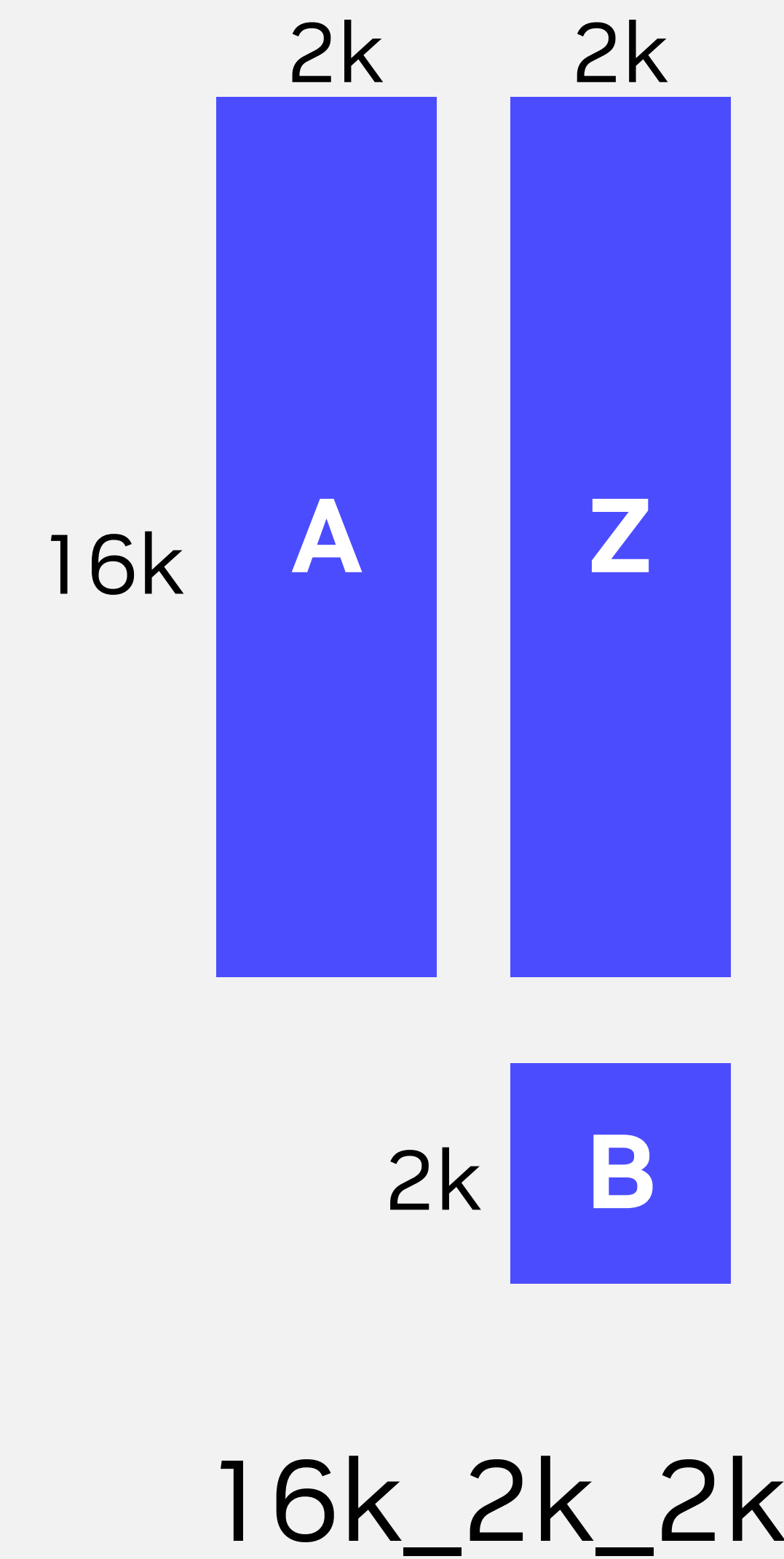
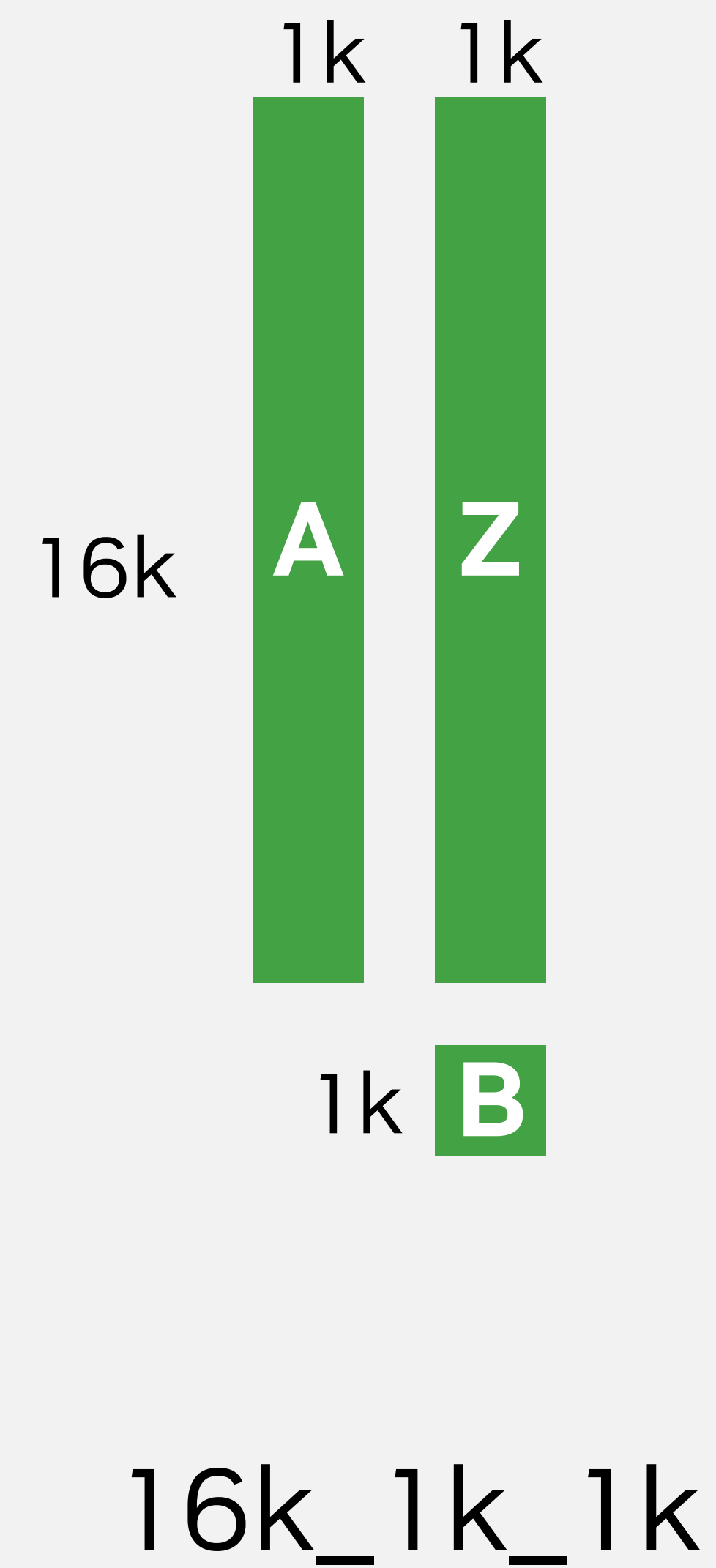
$$Z_{m,n} = A_{m,k} B_{k,n}$$

M – output row dim

K – reduction dim

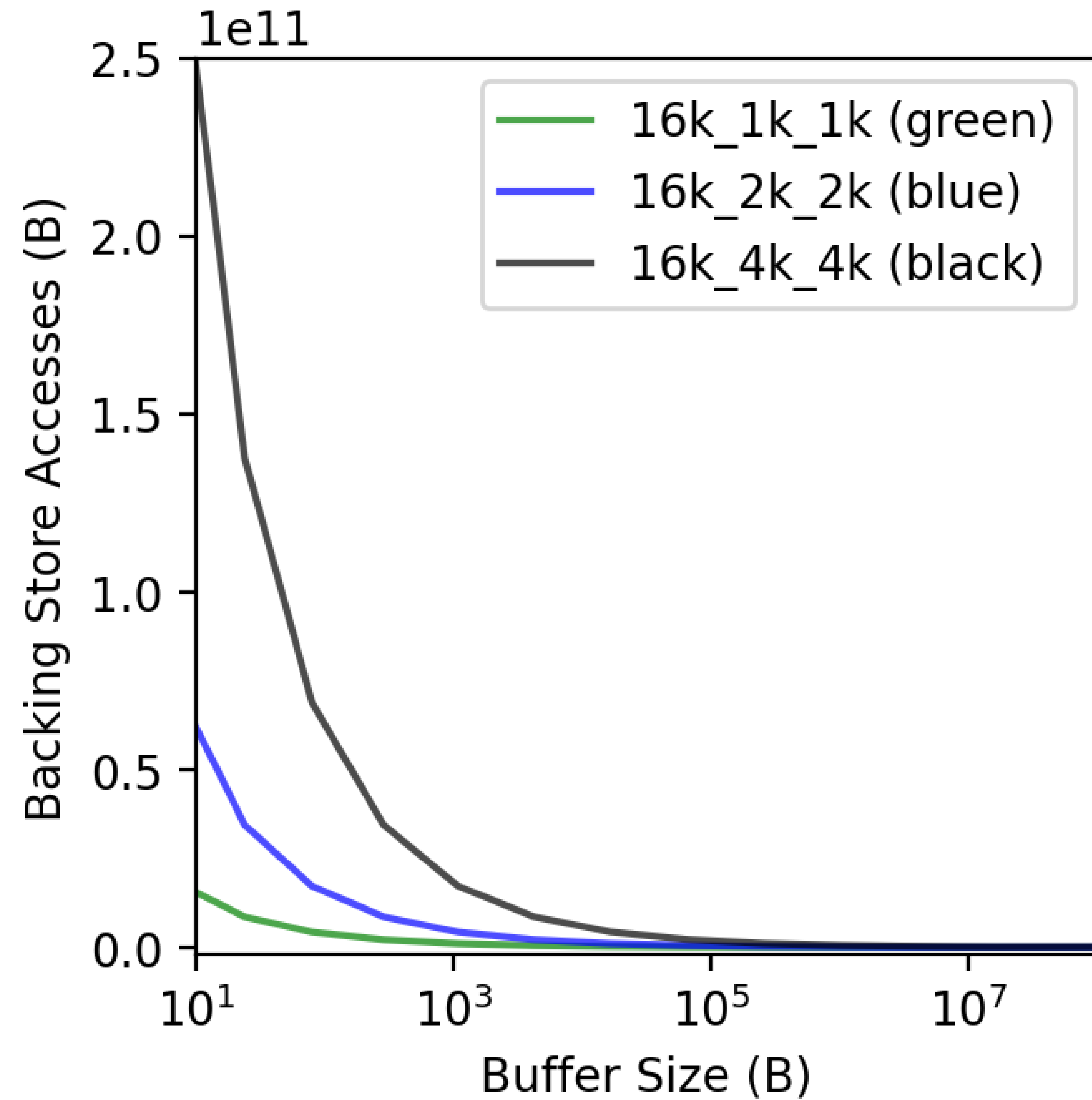
N – output column dim

Input Examples

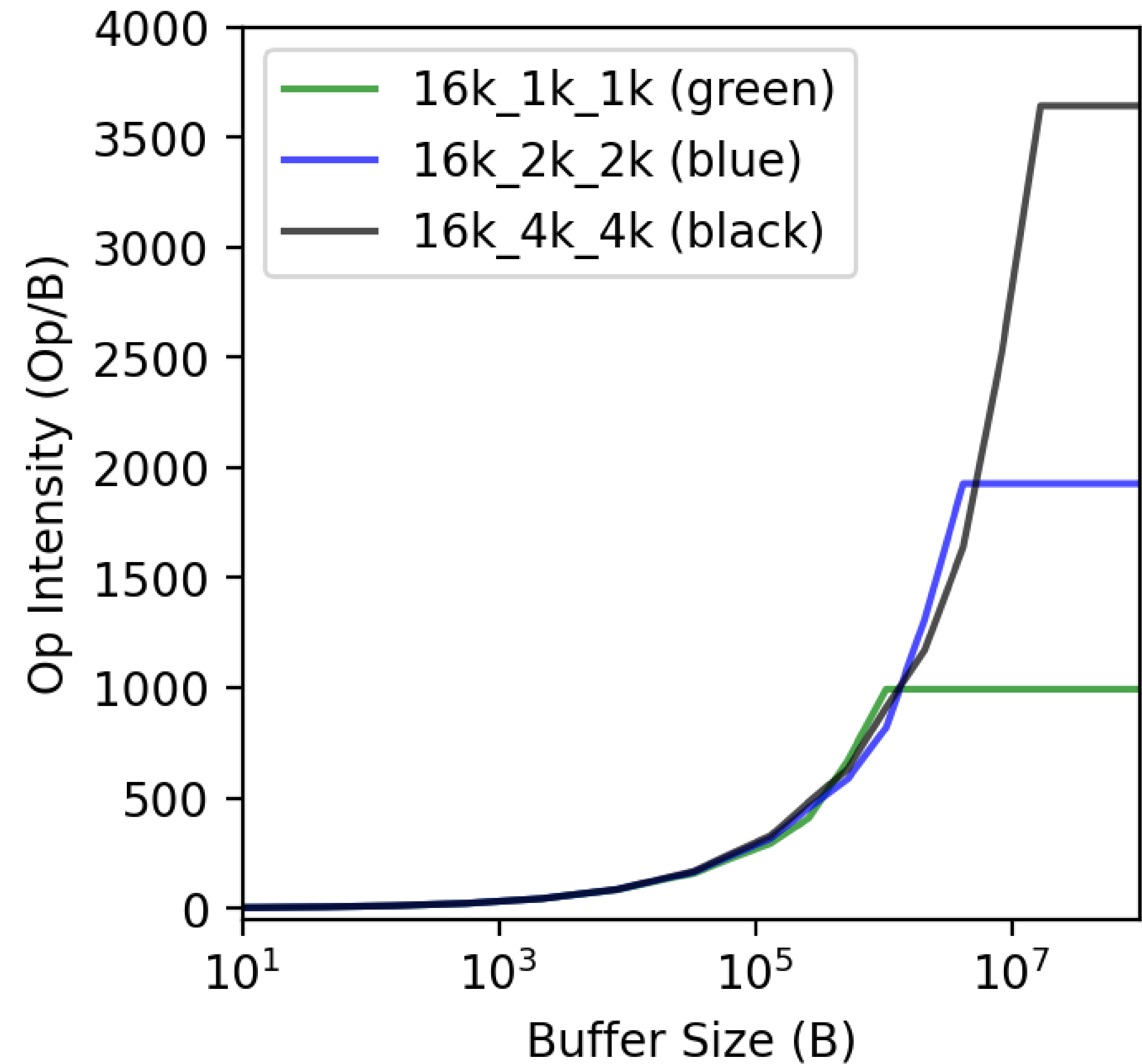


# Example *Orojenes* Analysis

## Ski-slope Diagram

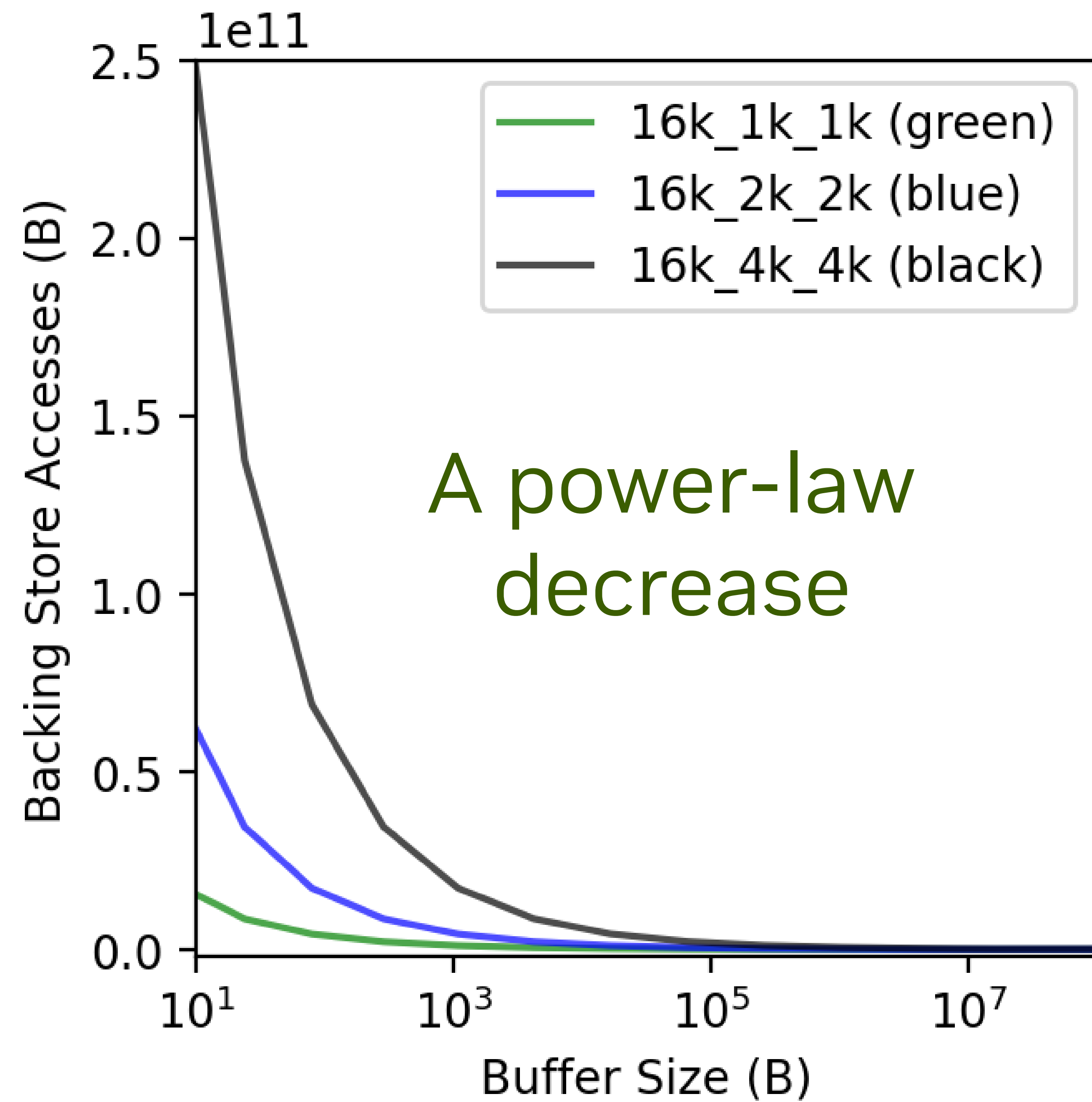


## OI Bound

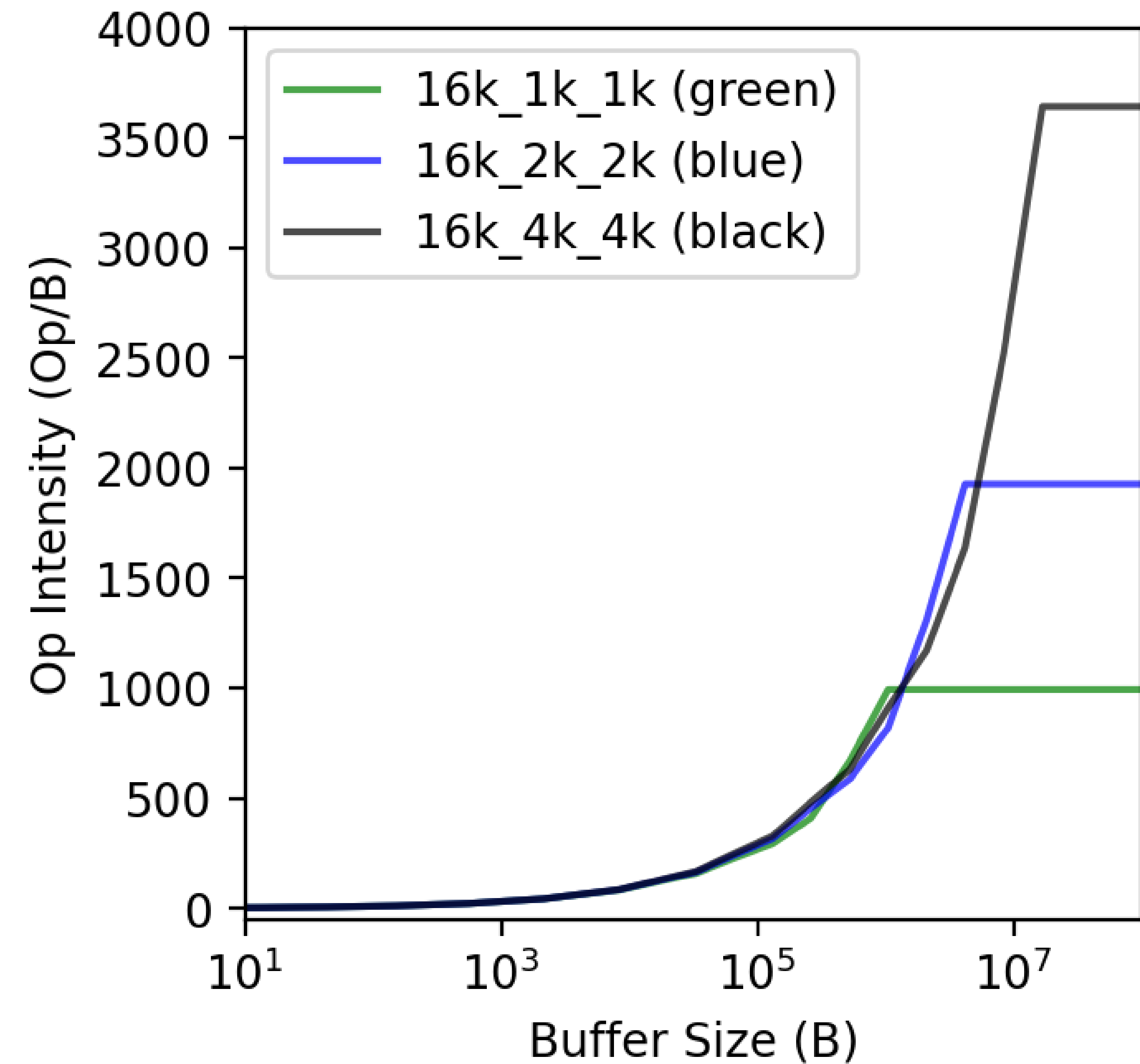


# Example *Orojenes* Analysis

## Ski-slope Diagram

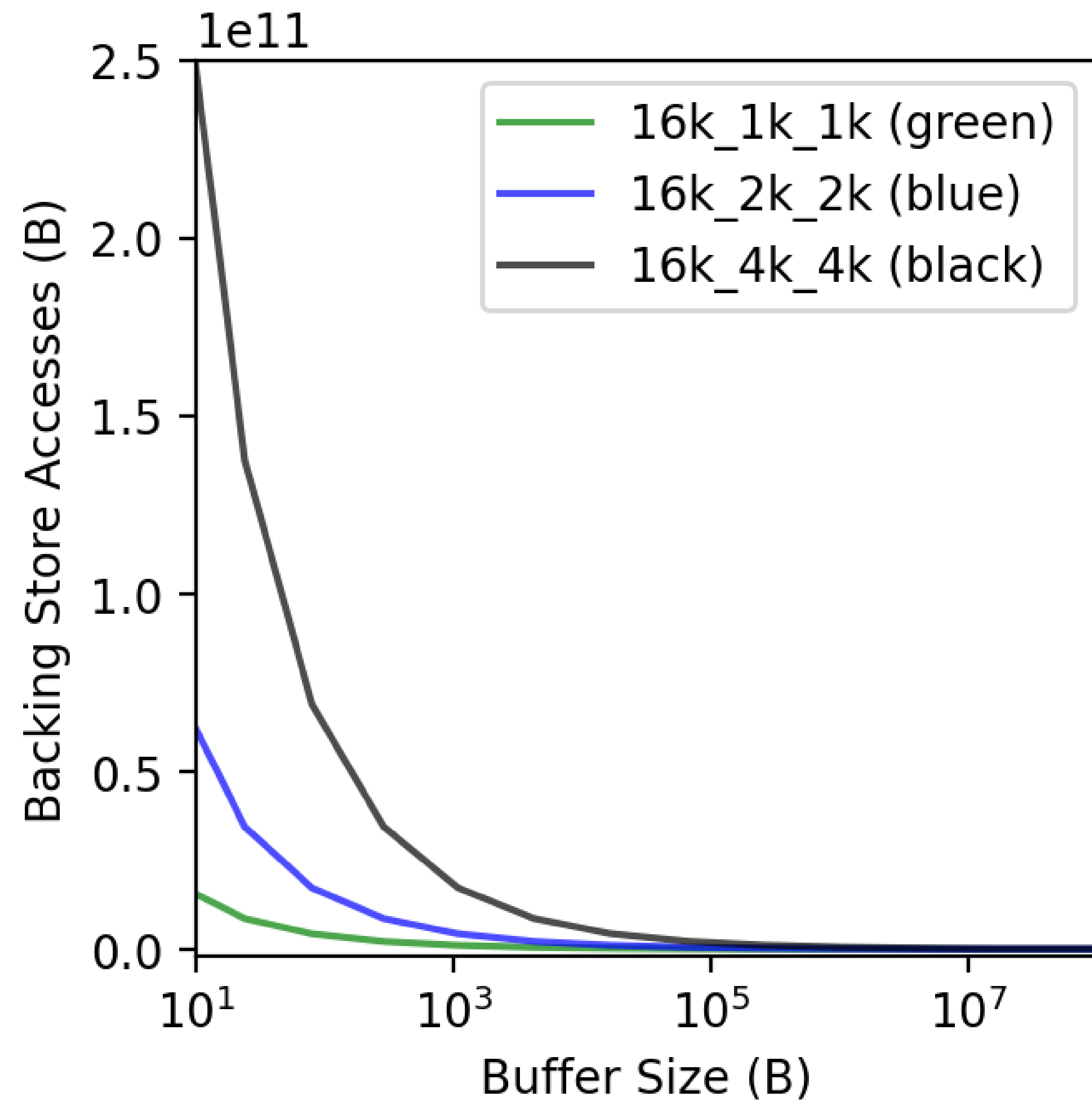


## OI Bound

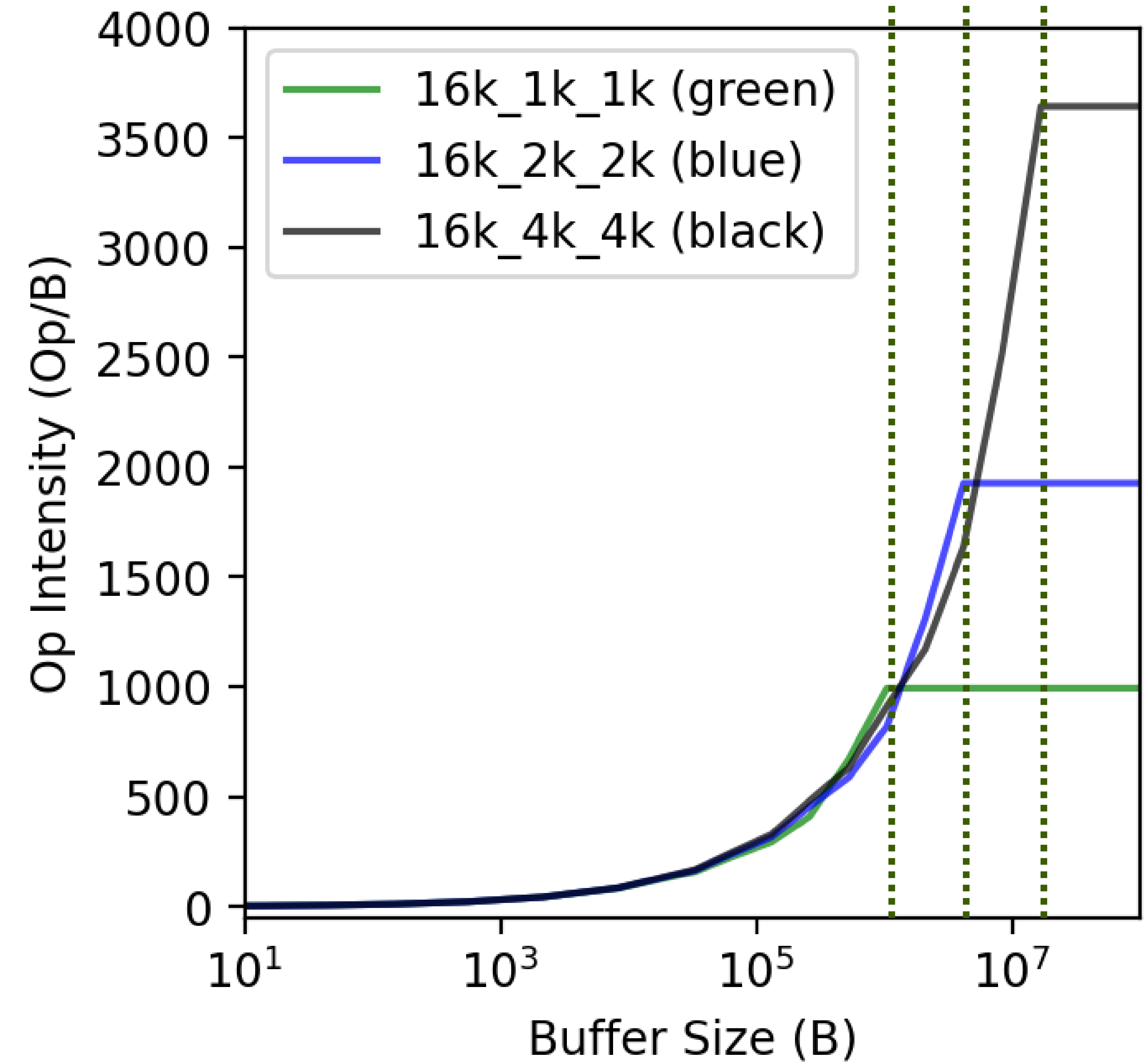


# Example *Orojenes* Analysis

## Ski-slope Diagram



## OI Bound 1 4 16 MB



The maximal effectual buffer size to achieve min data movement for a GEMM is approximately its **smallest operand size**

**#1: Orojenesi produces bounds that reveal powerful design insights**

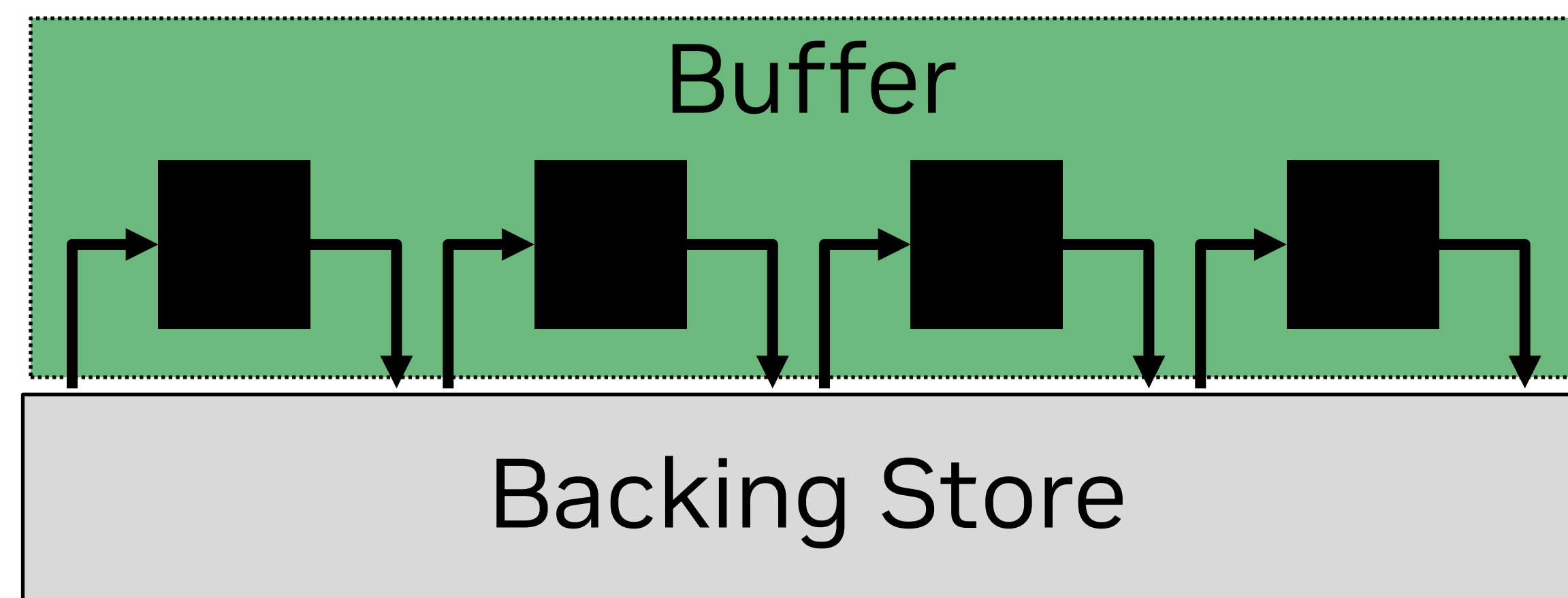
# The *Orogenesis* Fusion Flow

## Inputs

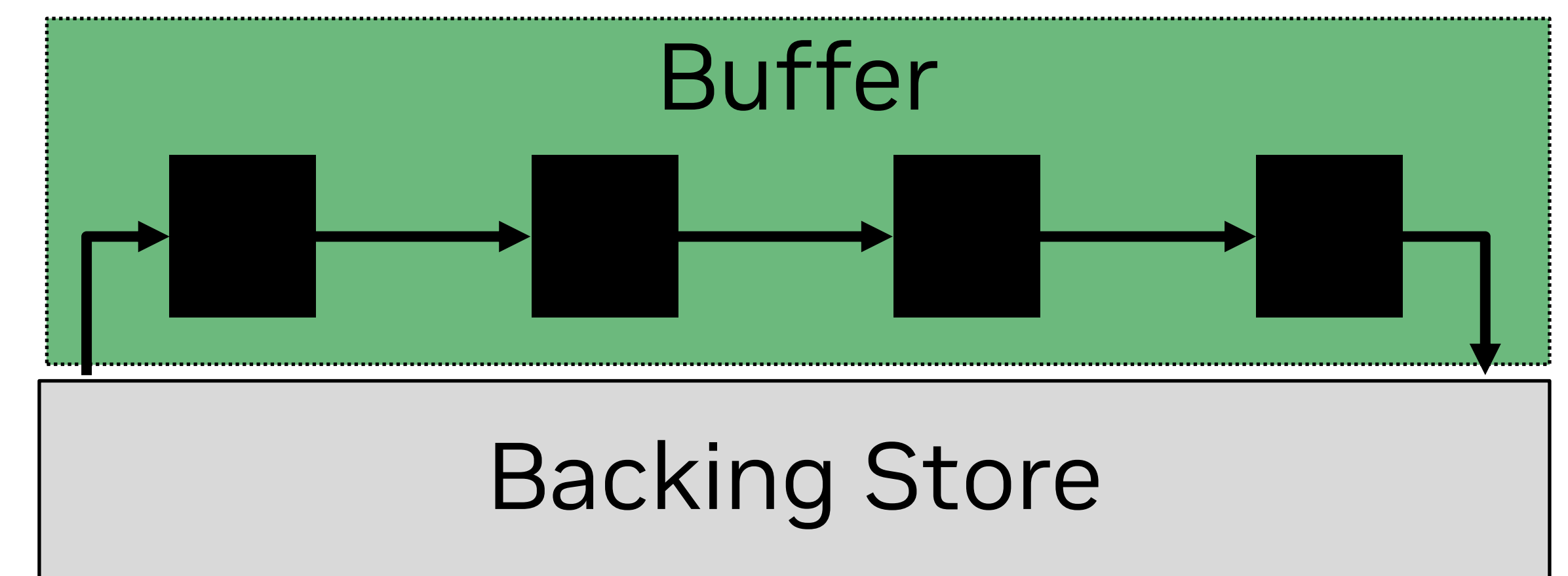
Chain of Einsums



## Unfused



## Fused



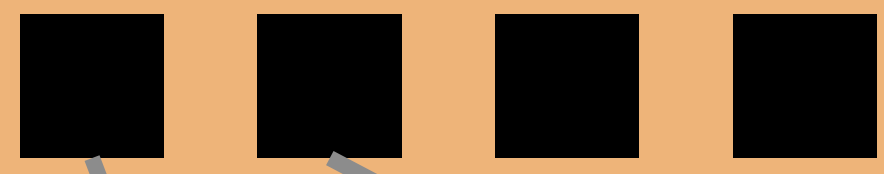
**Fusion** is an effective technique to minimize data movement for a chain of operations



# The *Orojenes* Fusion Flow

## Inputs

Chain of Einsums



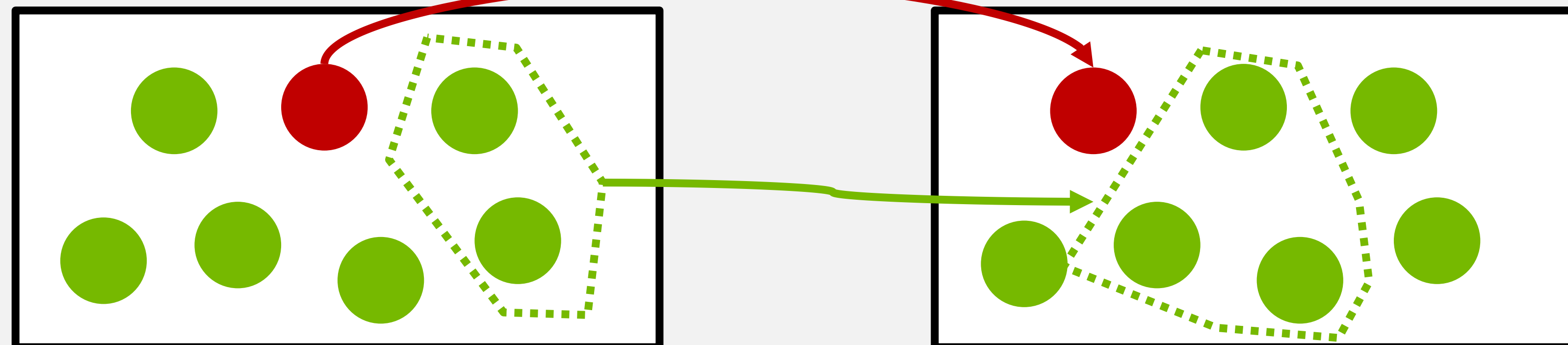
**Fusion** imposes extra intra-layer mapping constraints

## Mapspace

Producer

Incompatible

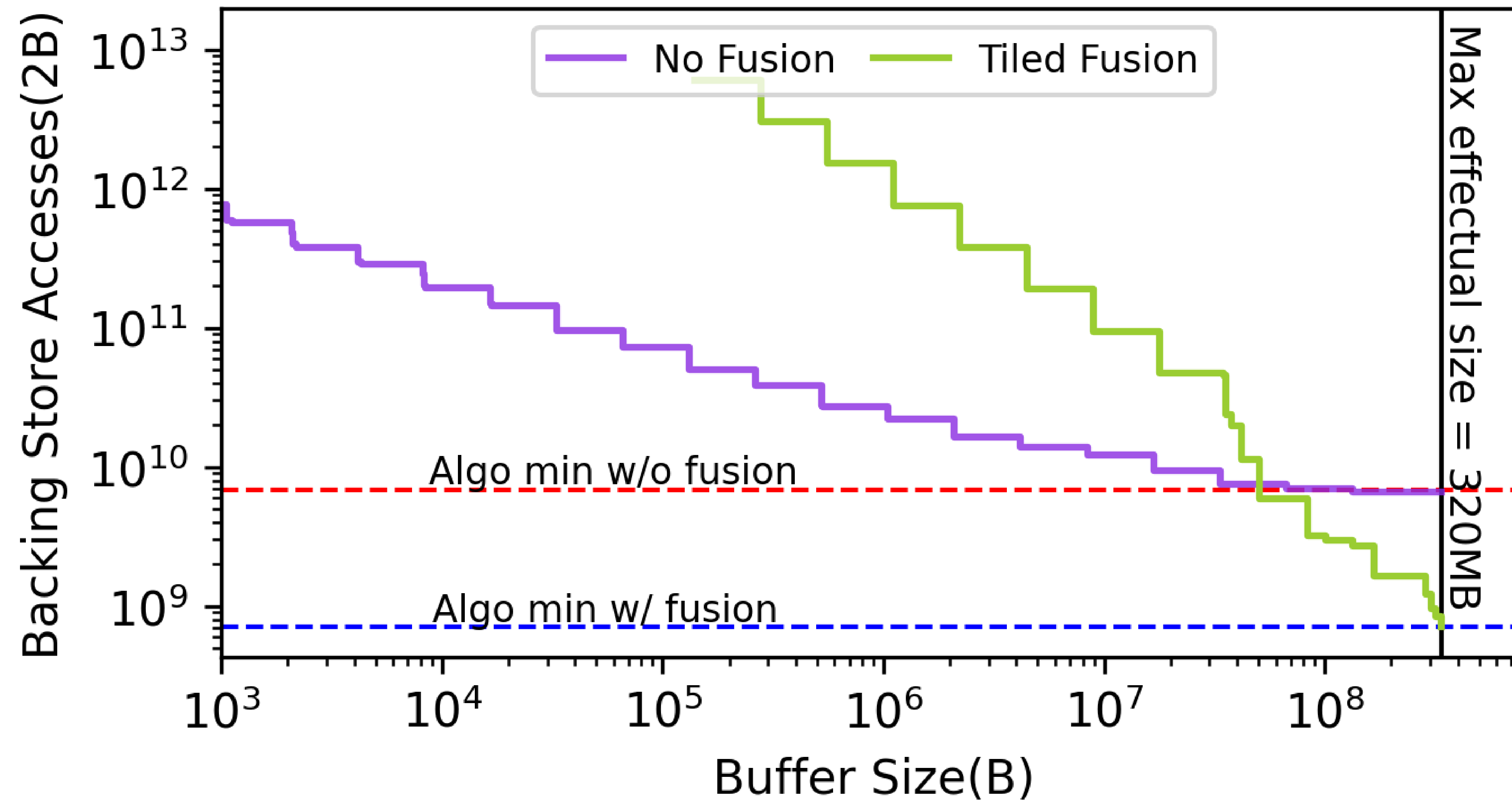
Consumer



**w/ Fusion**

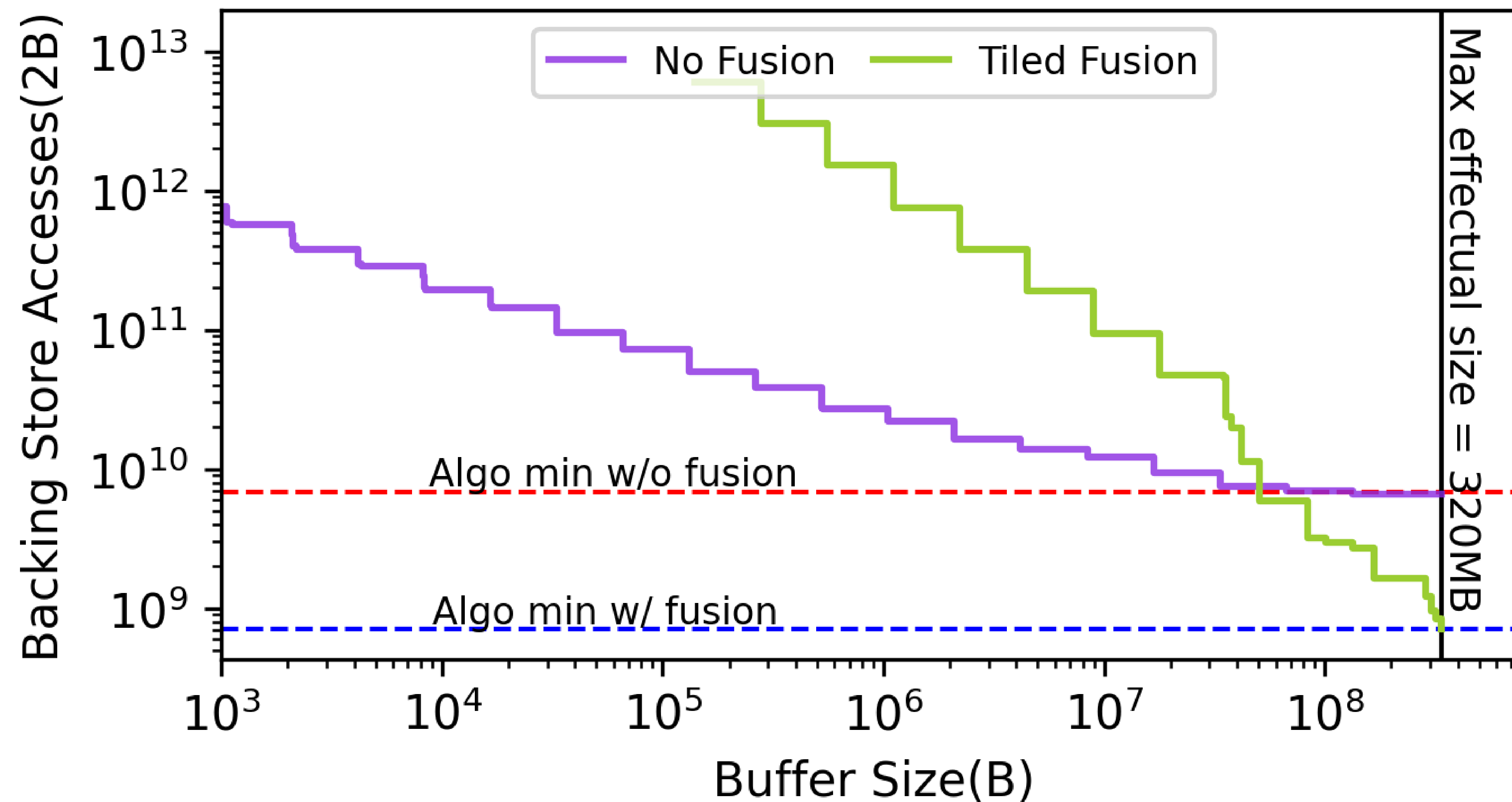
# Fusion Analysis

A chain of 6 operations in GPT-6.7b block



# Fusion Analysis

A chain of 6 operations in GPT-6.7b block

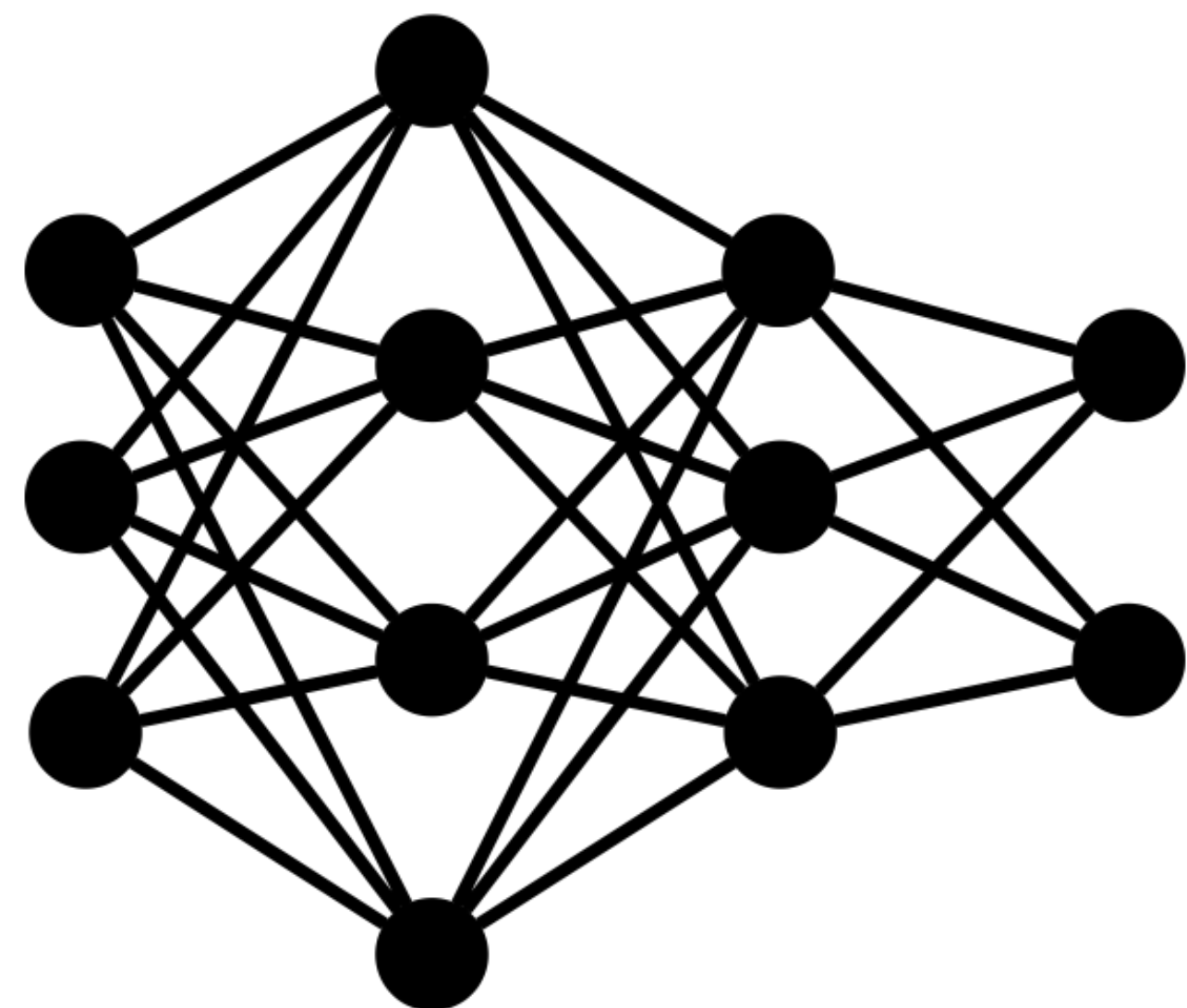


Fusion is effective when buffer size is large

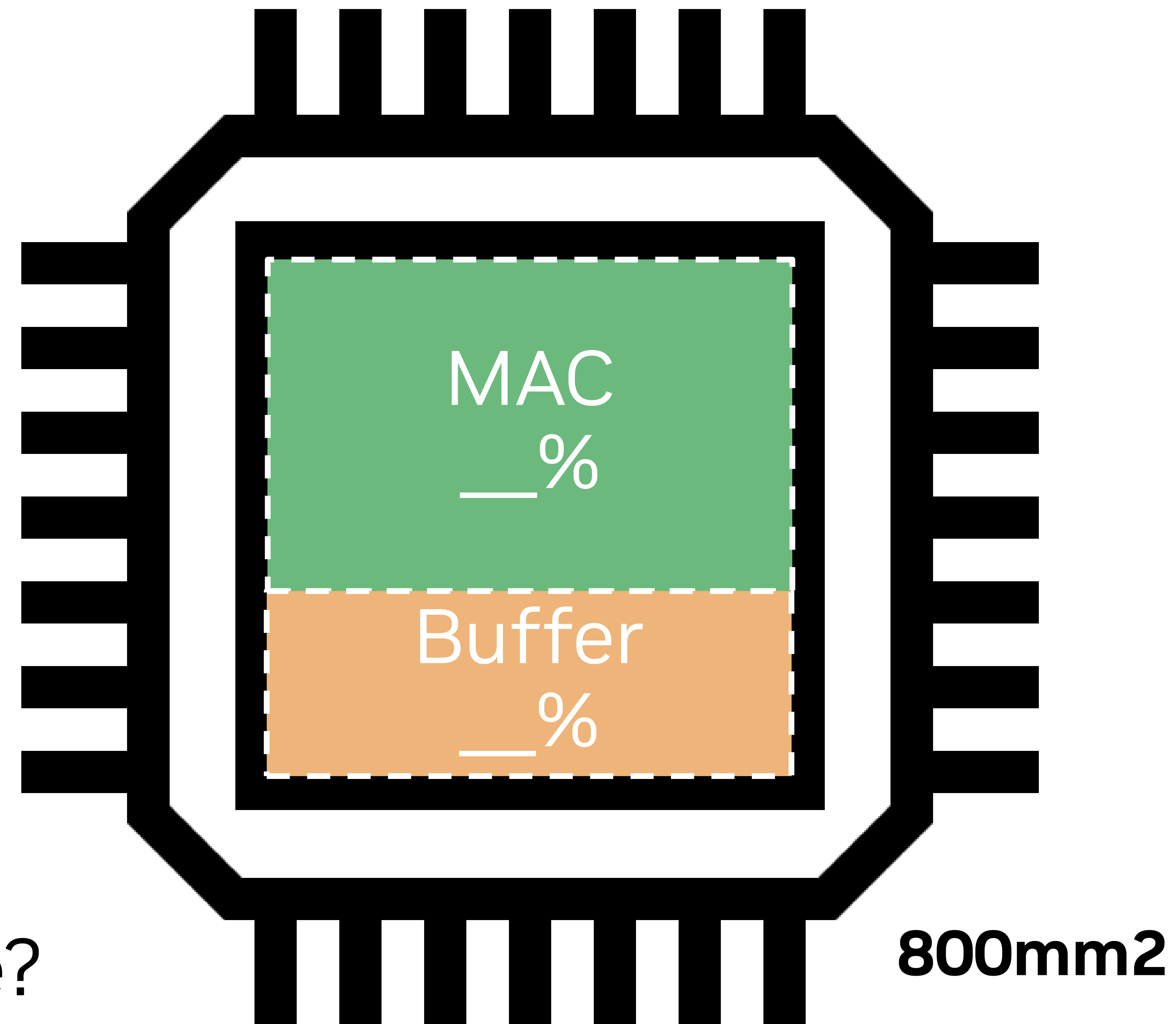
# **#2: Orojenesis comprehends complex workload optimizations**

# Motivation: A design challenge

Algorithm

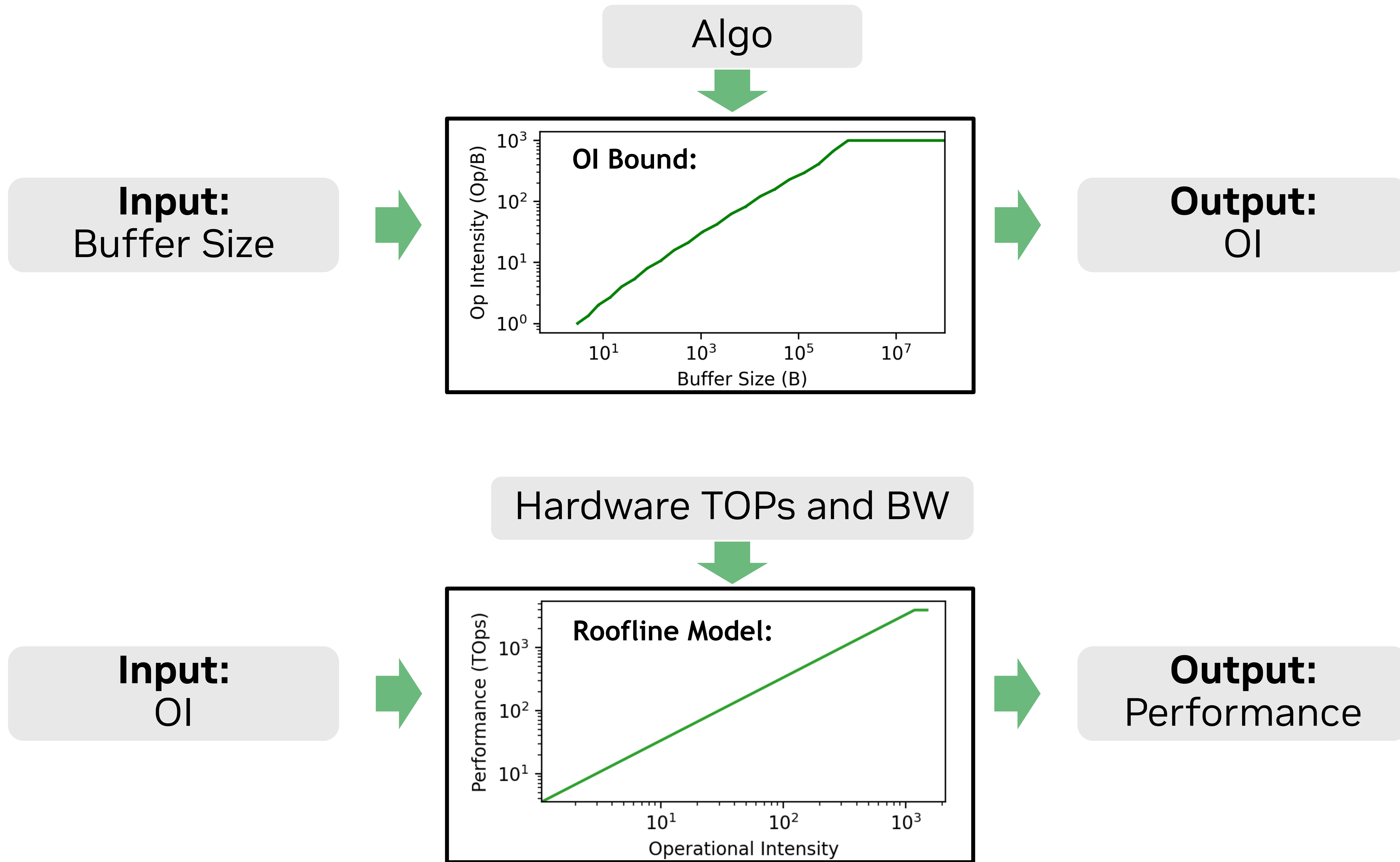


**GPTx**

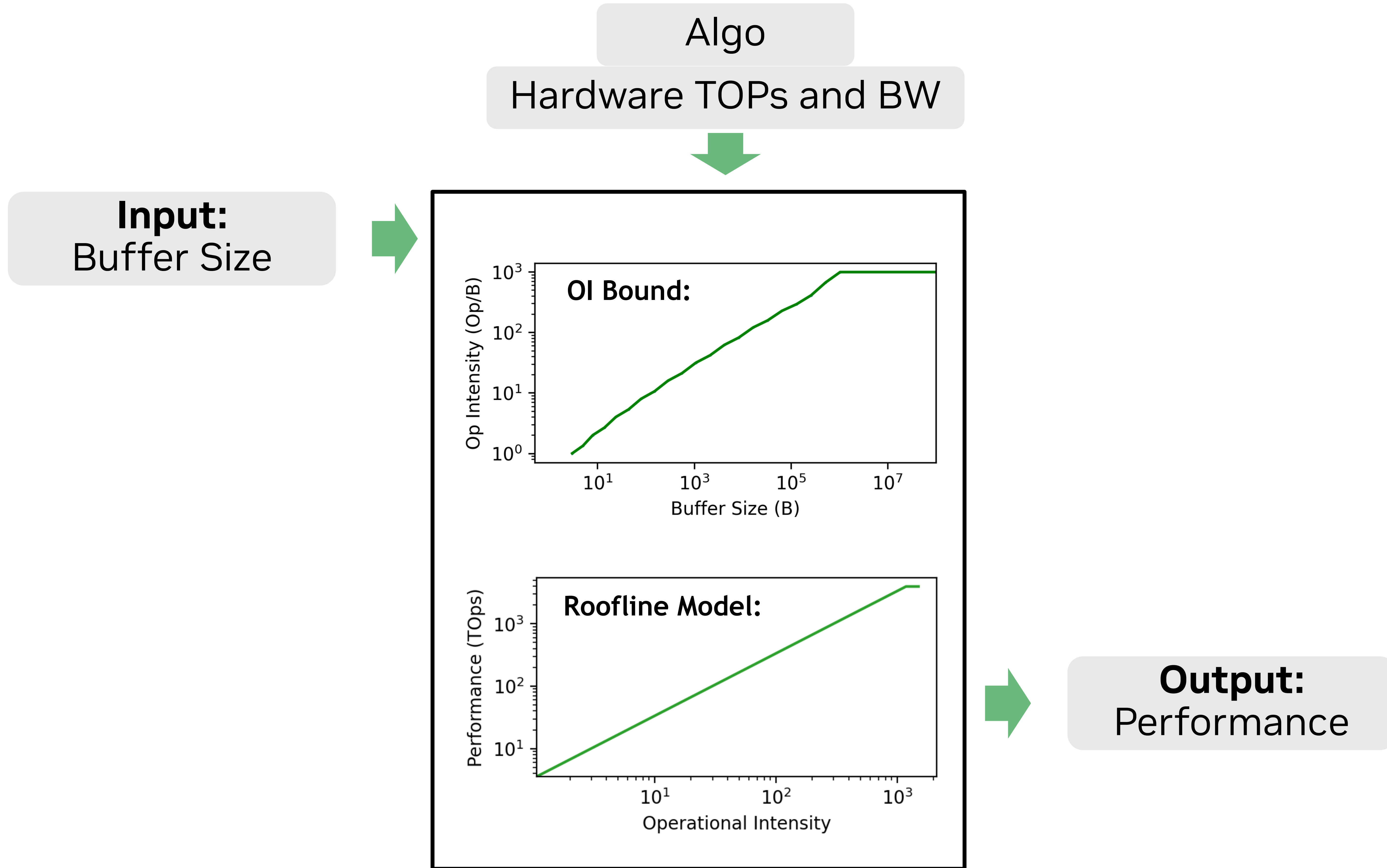


How to provision chip area  
between storage and compute?

# Orojenes Performance Model

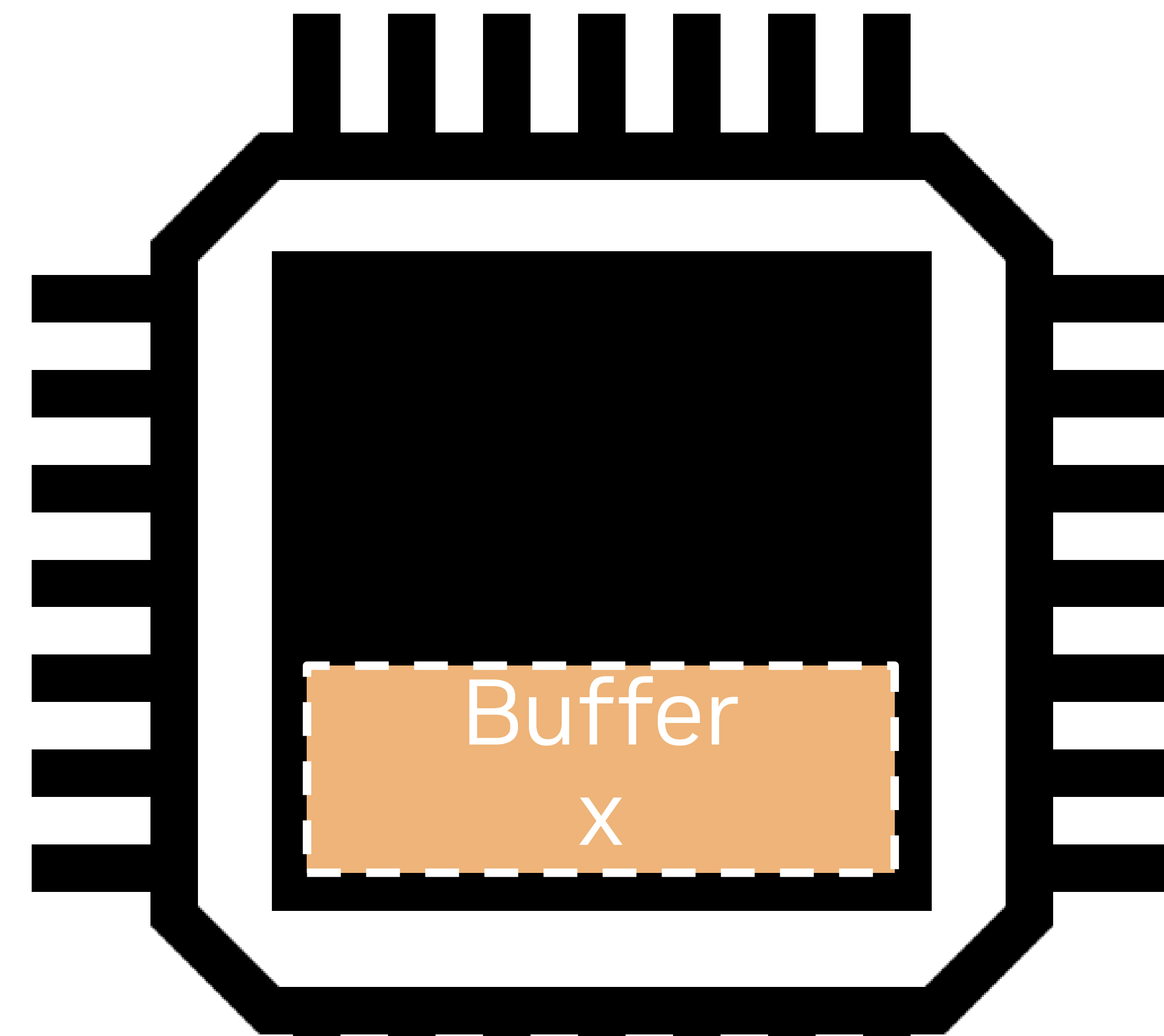
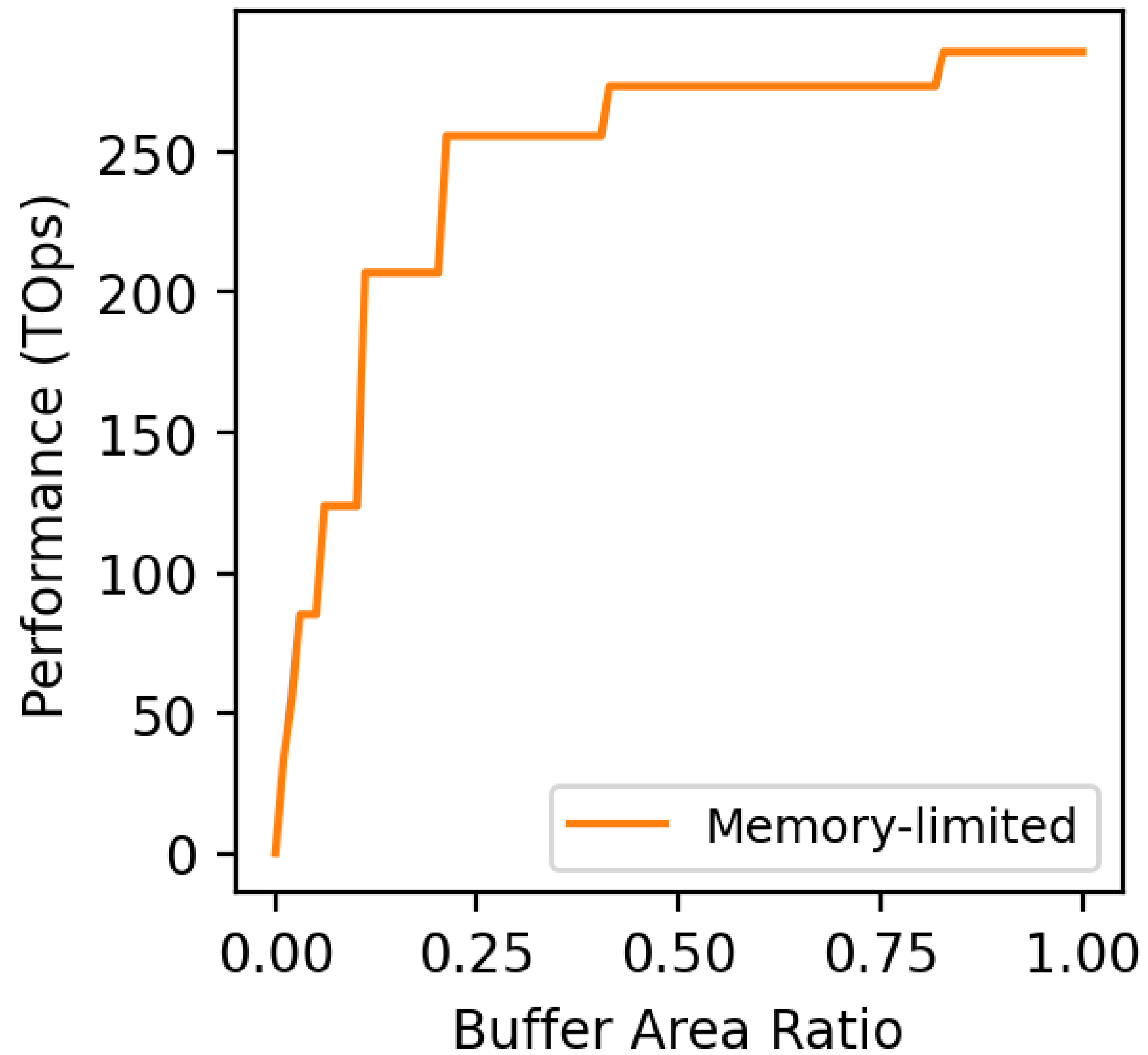


# Orojenesis Performance Model



# Orojenesis for DSE

GPT3-6.7b



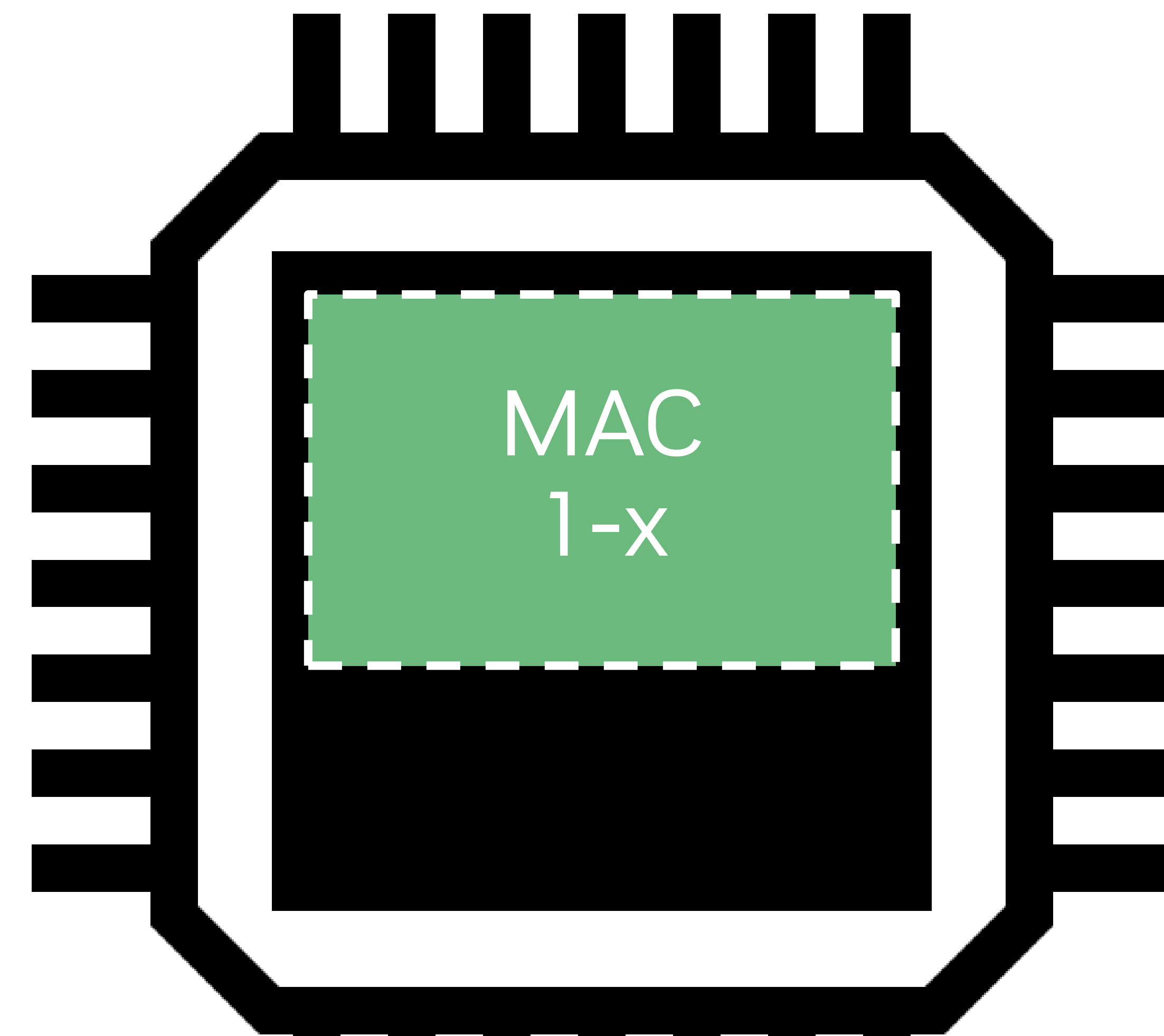
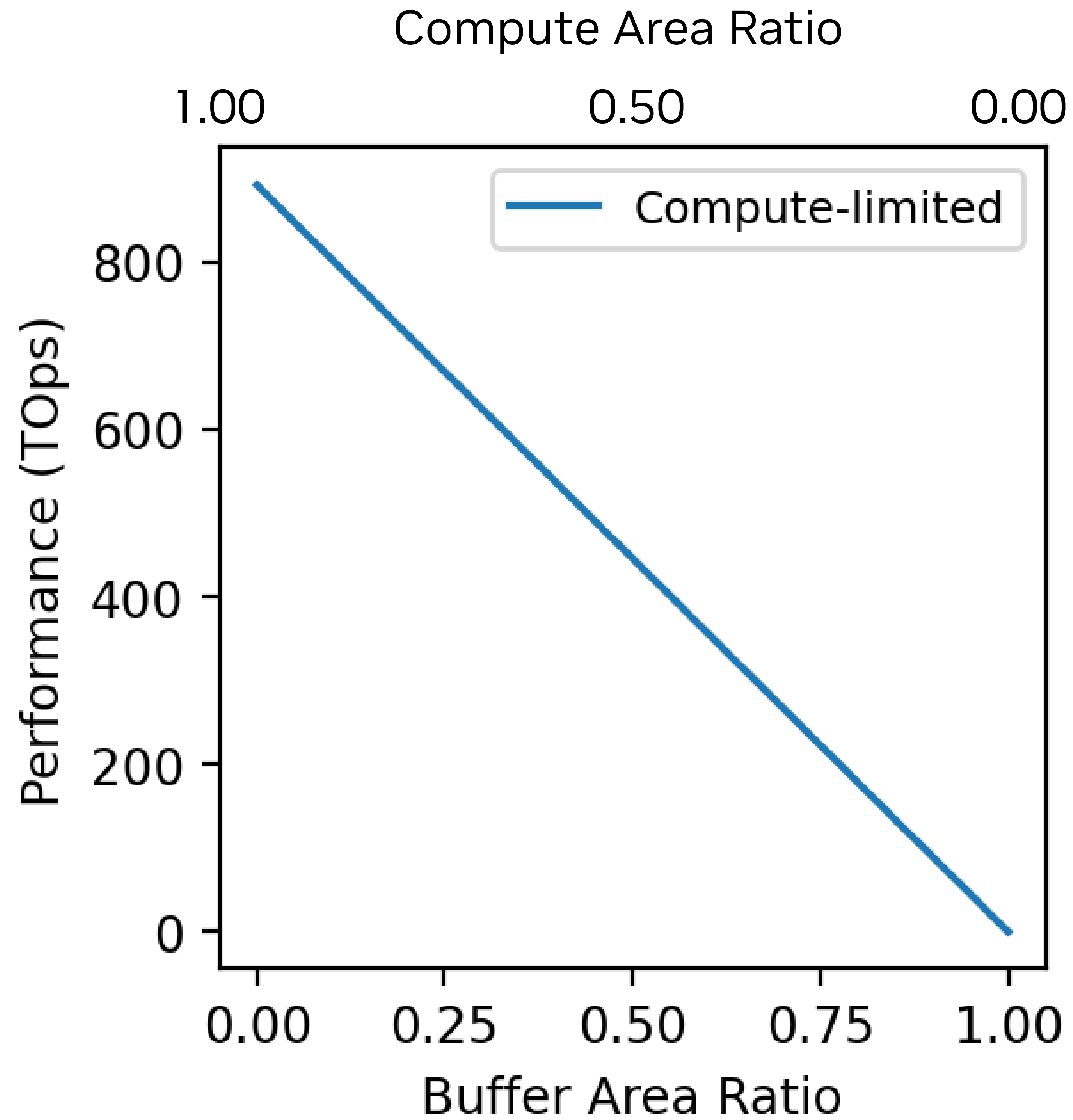
## HW Specs:

- Total chip area
- Area per Byte
- Area per MAC
- Backing-store BW
- Frequency



# Orojenesi for DSE

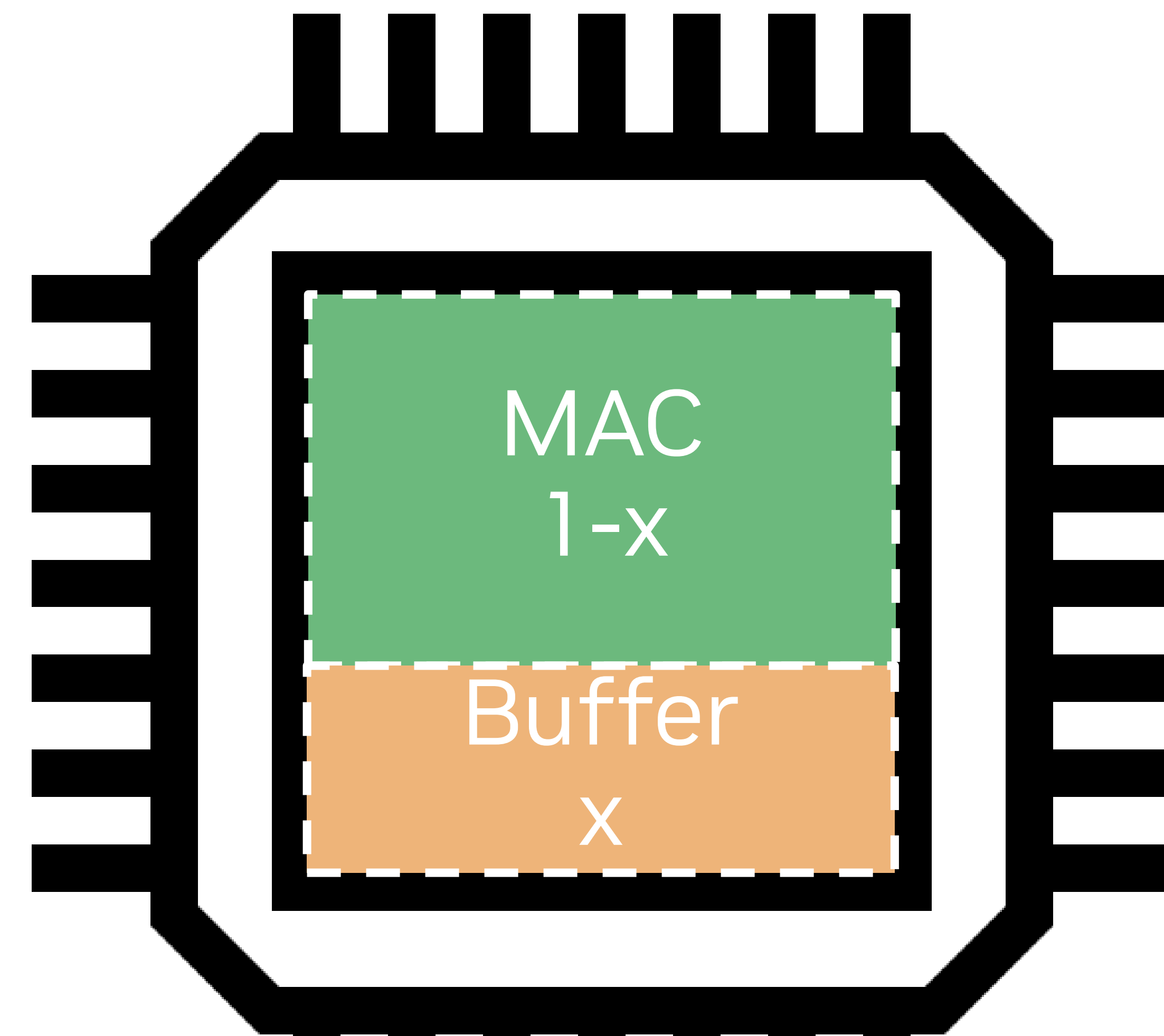
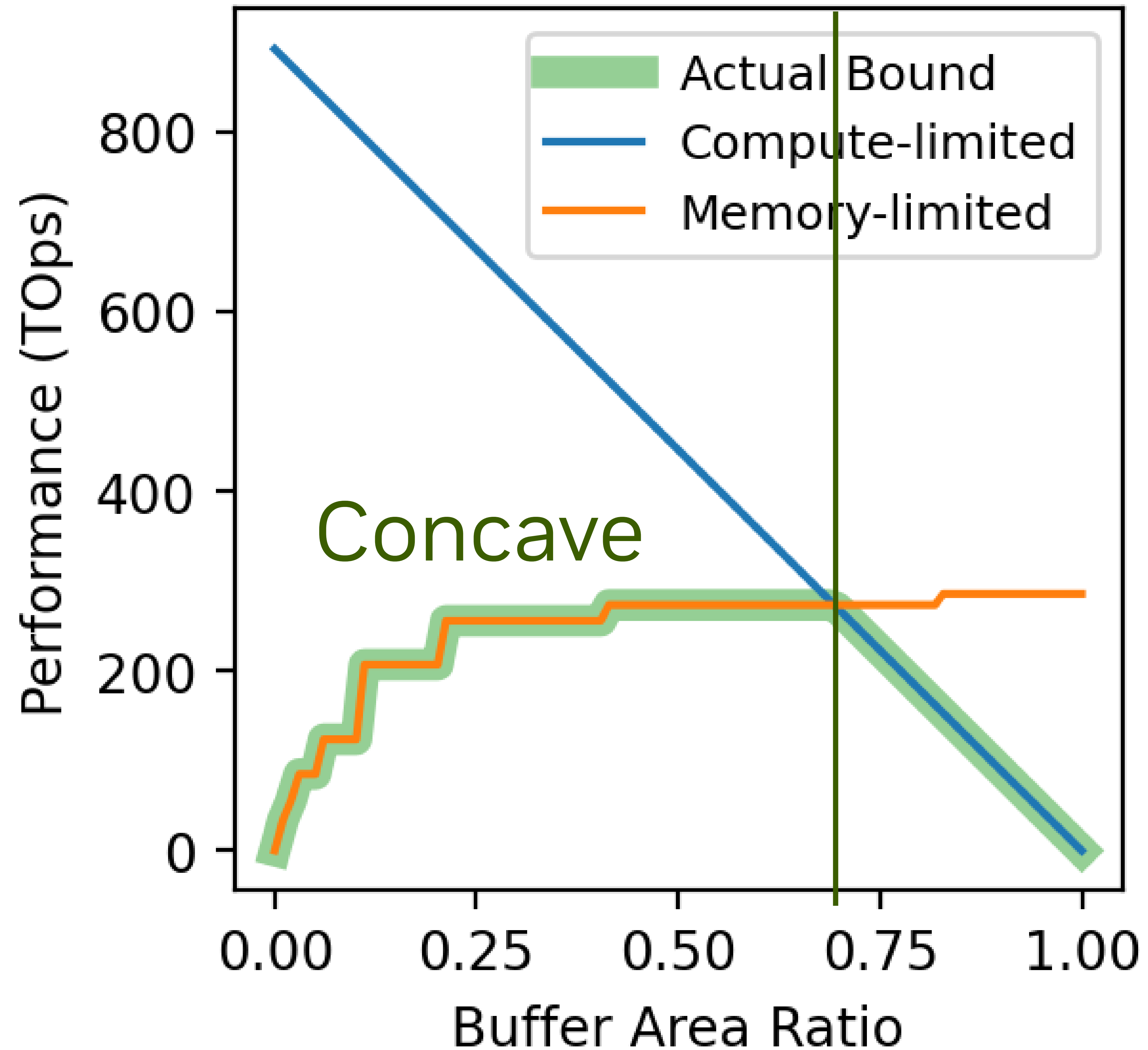
GPT3-6.7b



**HW Specs:**  
Total chip area  
Area per Byte  
Area per MAC  
Backing-store BW  
Frequency

# Orojenesis for DSE

GPT3-6.7b

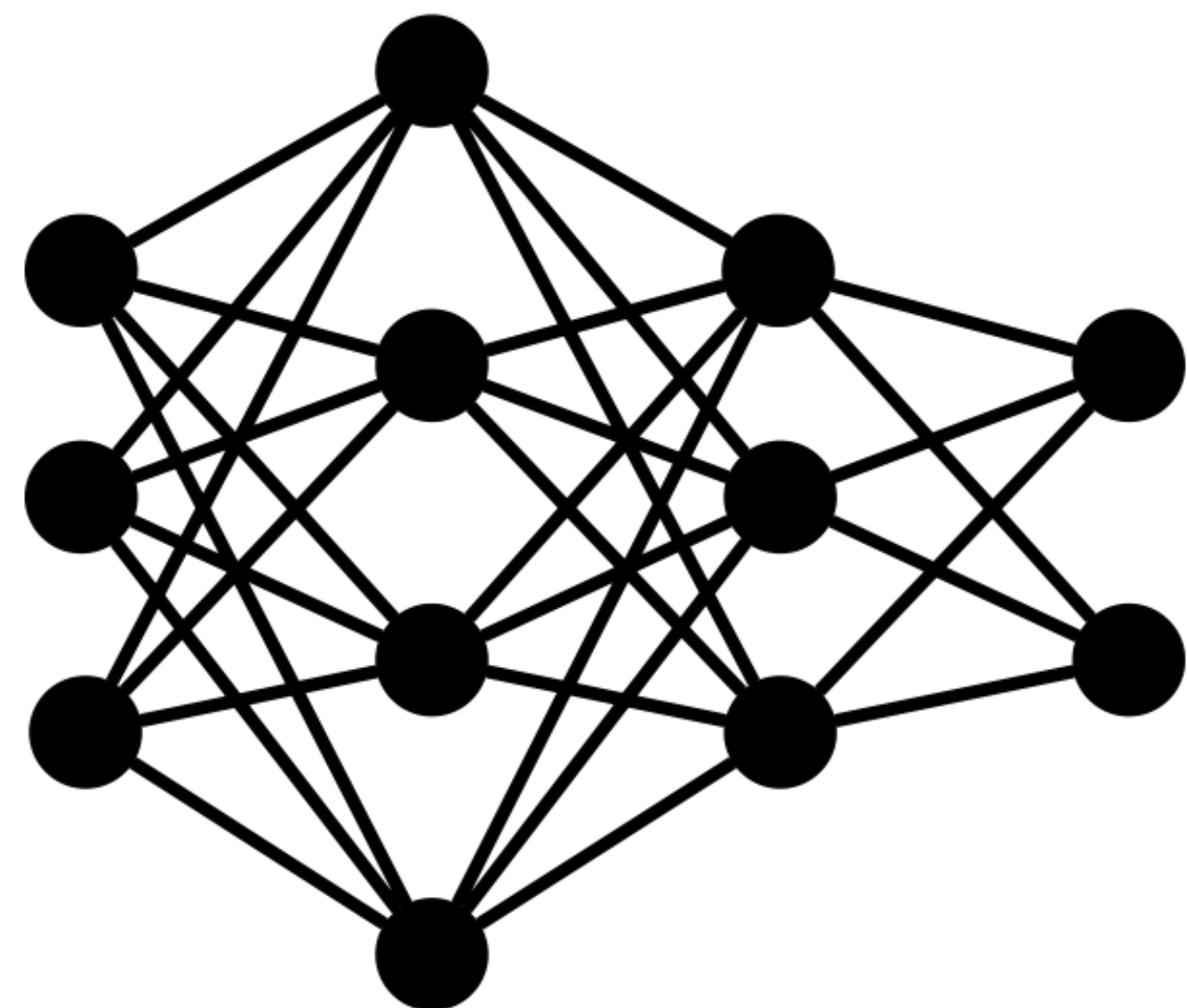


## HW Specs:

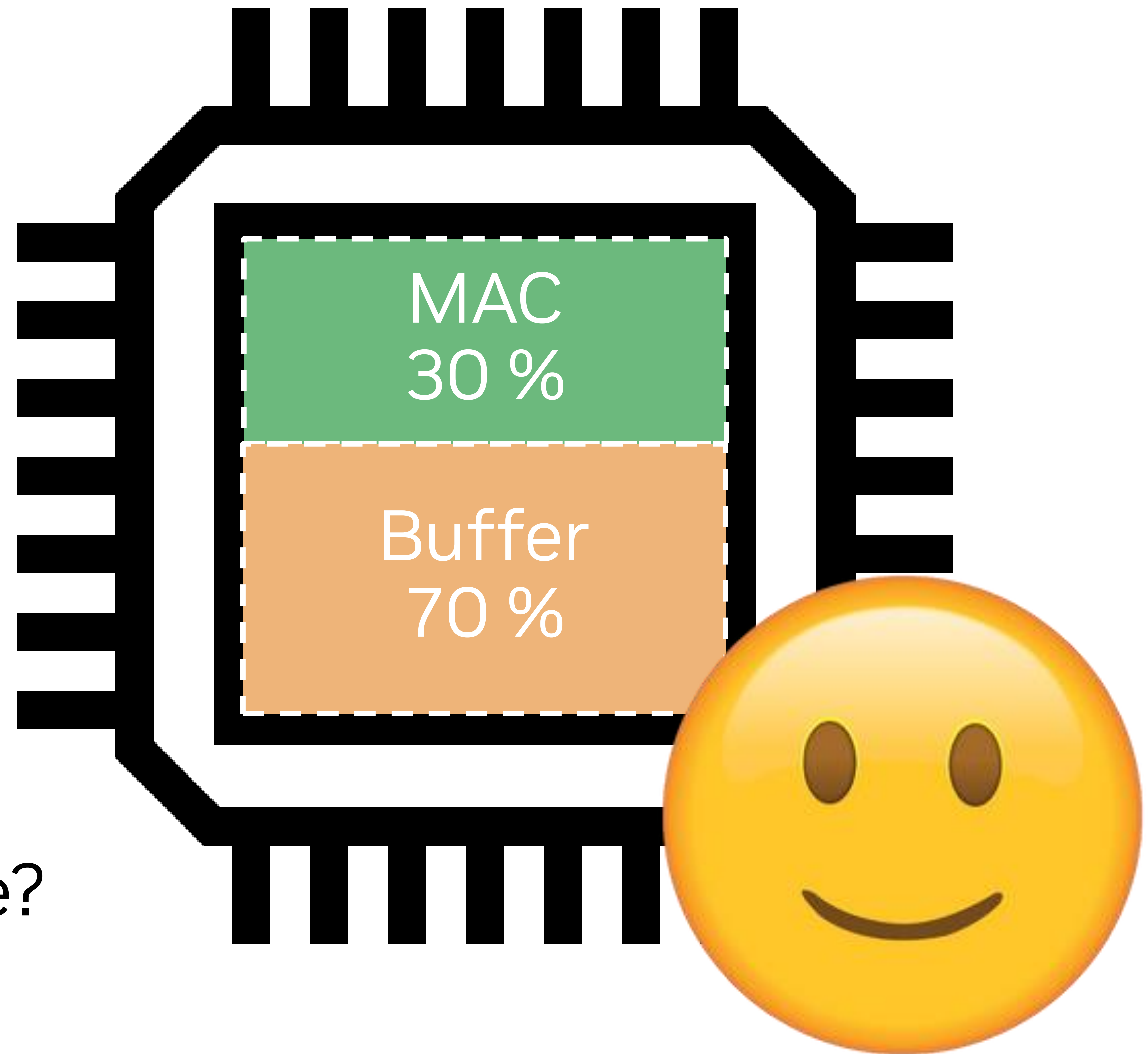
Total chip area  
Area per Byte  
Area per MAC  
Backing-store BW  
Frequency

# Motivation: A design challenge

Algorithm



**GPTx**

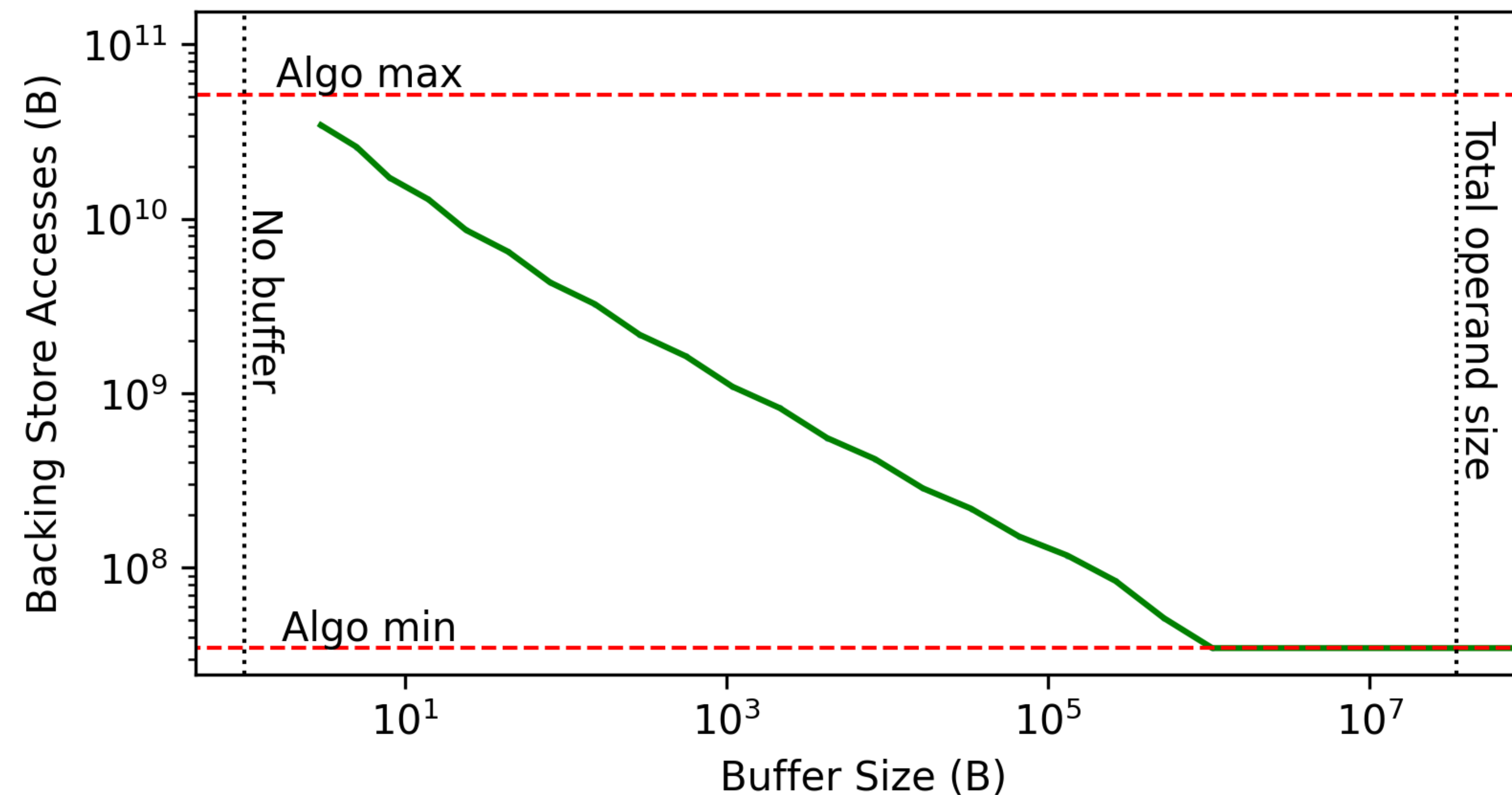


How to provision chip area between storage and compute?

**#3: Orojenesis complements the roofline model to provide buffer size suggestions**

# Orojenesis

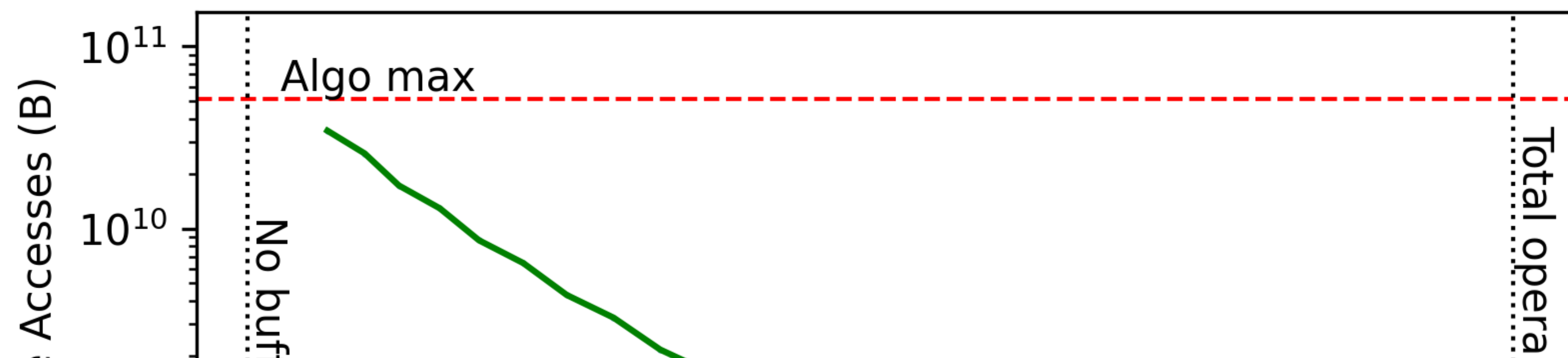
- A radically new design approach for early-stage architectural DSE
- Offers **visualization** and **insights** for design tradeoffs
- Can be **a powerful addon** to the roofline performance model



Ski-slope Diagram

# Orojenesis

- A radically new design approach for early-stage architectural DSE
- Offers **visualization** and **insights** for design tradeoffs
- Can be **a powerful addon** to the roofline performance model



Website: <https://timeloop.csail.mit.edu/orojenesis>

Artifact: [zenodo.org/doi/10.5281/zenodo.10850531](https://zenodo.org/doi/10.5281/zenodo.10850531)

