Blue Brain Project

Building Brain Circuits

Experiences with shuffling terabytes of data over MPI

MUG'20 : August 26, 2020

EPFL

Matthias Wolf, Nicolas Cornu, Pramod Kumbhar, James King
High Performance Computing Team, Blue Brain Project

Overview

Background & Motivation

- Blue Brain Project, use cases

Brain Circuits

- Reconstruction from biological data

TouchDetector

- Algorithm and implementation

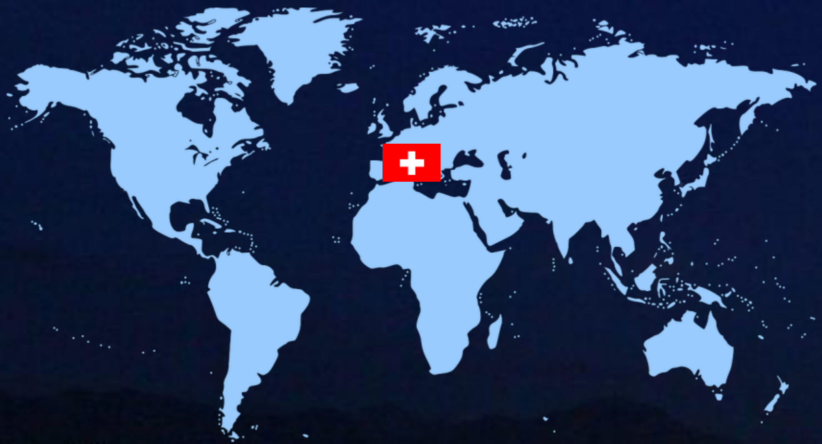
MPI Benchmarks

- Comparison of HPE-MPT, MVAPICH2

Summary and Future Work

- Next steps, improvements

Ecole polytechnique fédérale de Lausanne



- University in Lausanne, Switzerland
- One of the two Swiss federal institutes
- ~11k students, ~5k academic staff

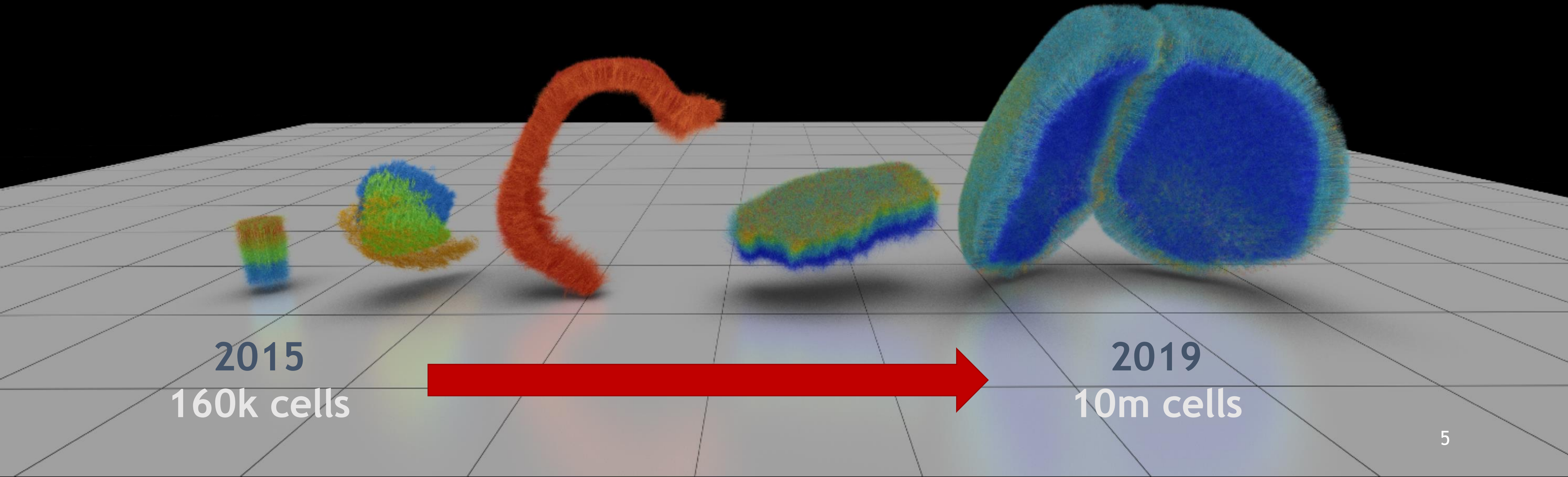
Blue Brain Project

- BBP started in 2005 @ BMI
- Moved to Campus Biotech ~4 years ago
- CB : Hub for Life Sciences

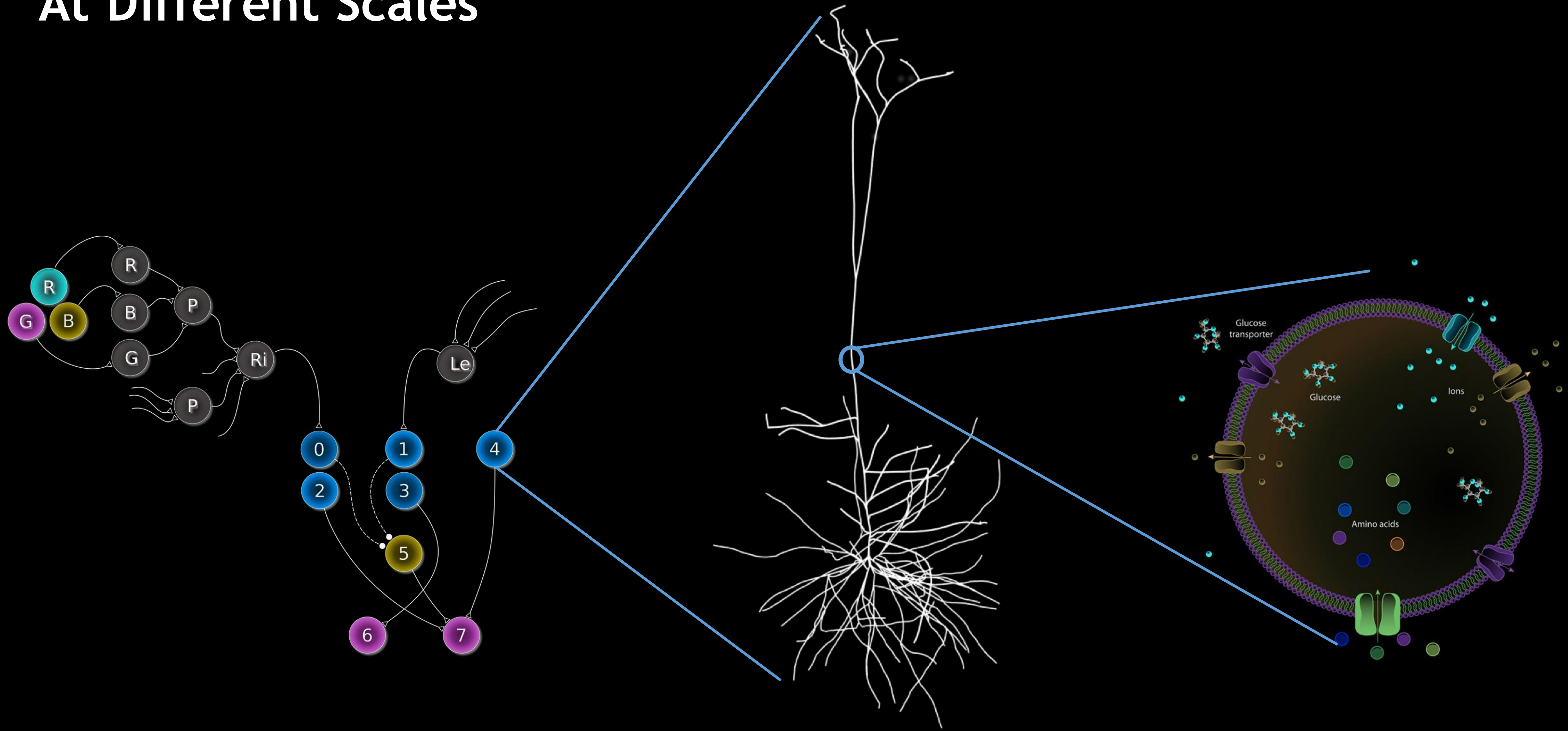
Project Timeline and Goals

Simulate the brain

- Focus on the rat, as structural close to human
- Initial groundwork before 2010 with 10k cells



At Different Scales



Point Neurons

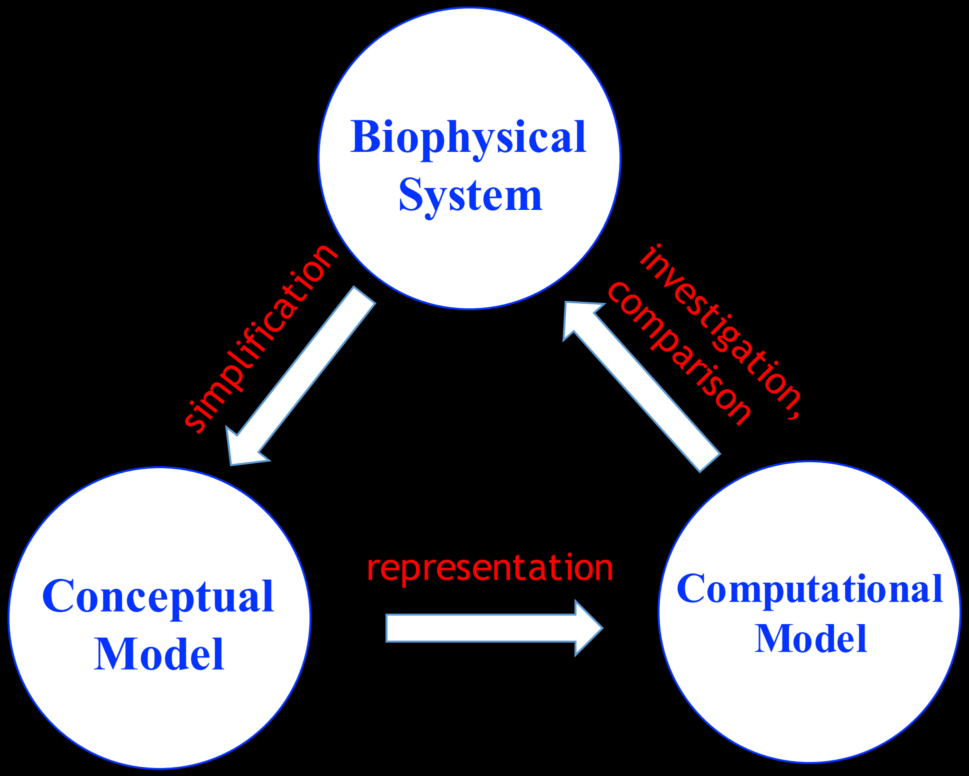
Morphologically Detailed Neurons

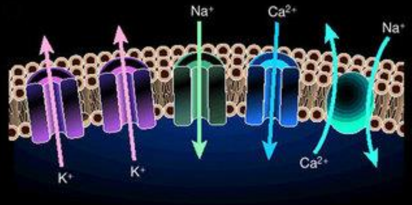
Molecular Level

Blue Brain Project

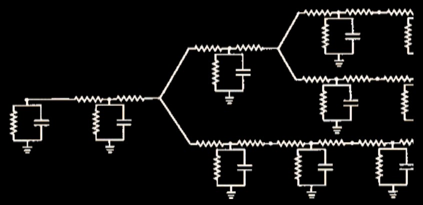
Comprehensive approach to systematically create unifying models of brain circuits by

- reverse engineering biological components
- construction models of the biophysics





See source7



morphologically detailed neurons

Blue Brain 5

Located at CSCS in Lugano

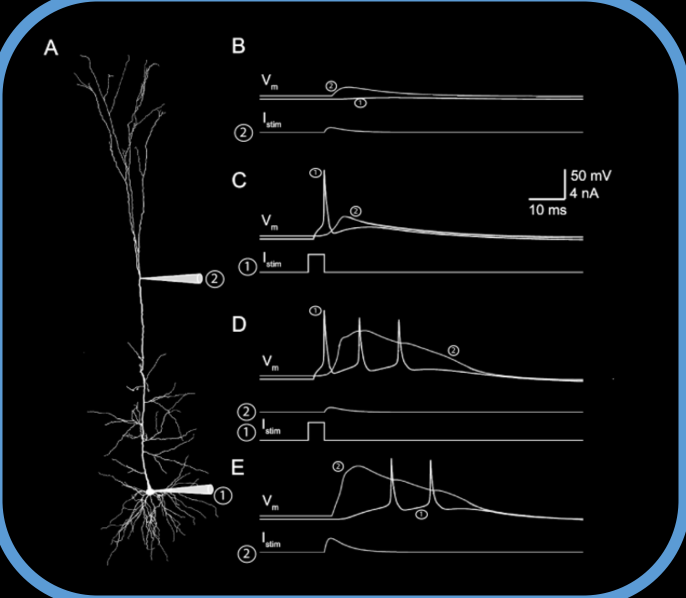
- ~200 Skylake nodes
- ~800 Cascade Lake nodes
- 100Gbit EDR InfiniBand, fat tree
- Auxiliary GPU nodes
- 6 Pb storage in GPFS
- Burst Buffer (IME)

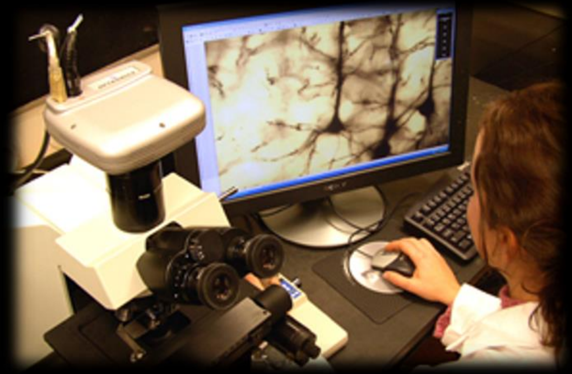
- Vendor provides **HPE-MPI**
- We also provide **MVAPICH2** (faster IME support)
- Want improve performance, avoid lock-in

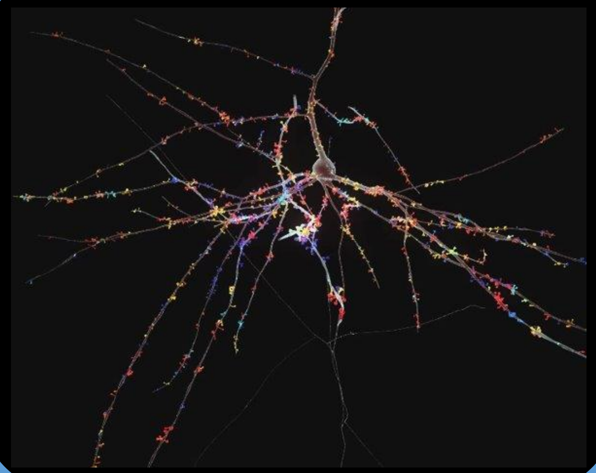
Brain Circuits

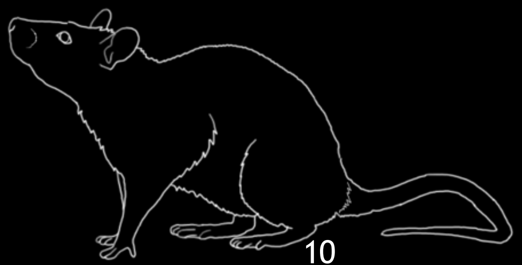
From in-vitro to in-silico

Obtaining input data from lab experiments...

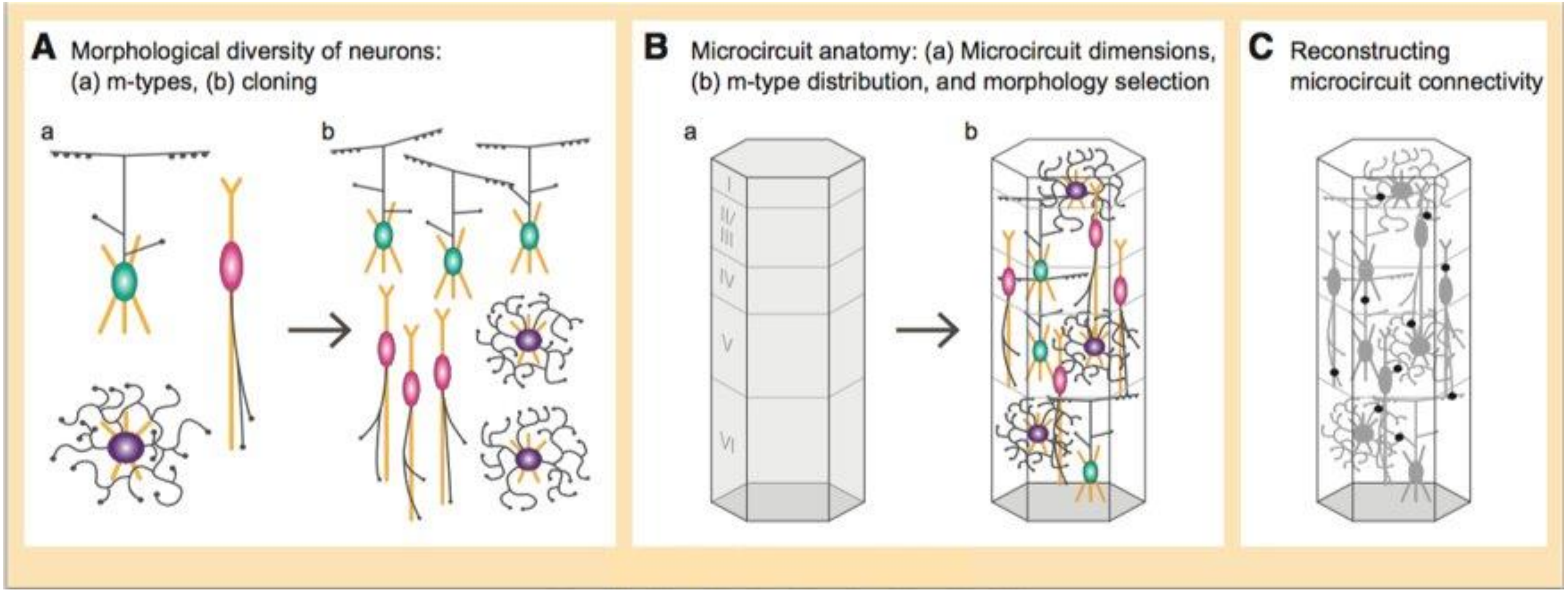








Data Driven Reconstruction Workflow



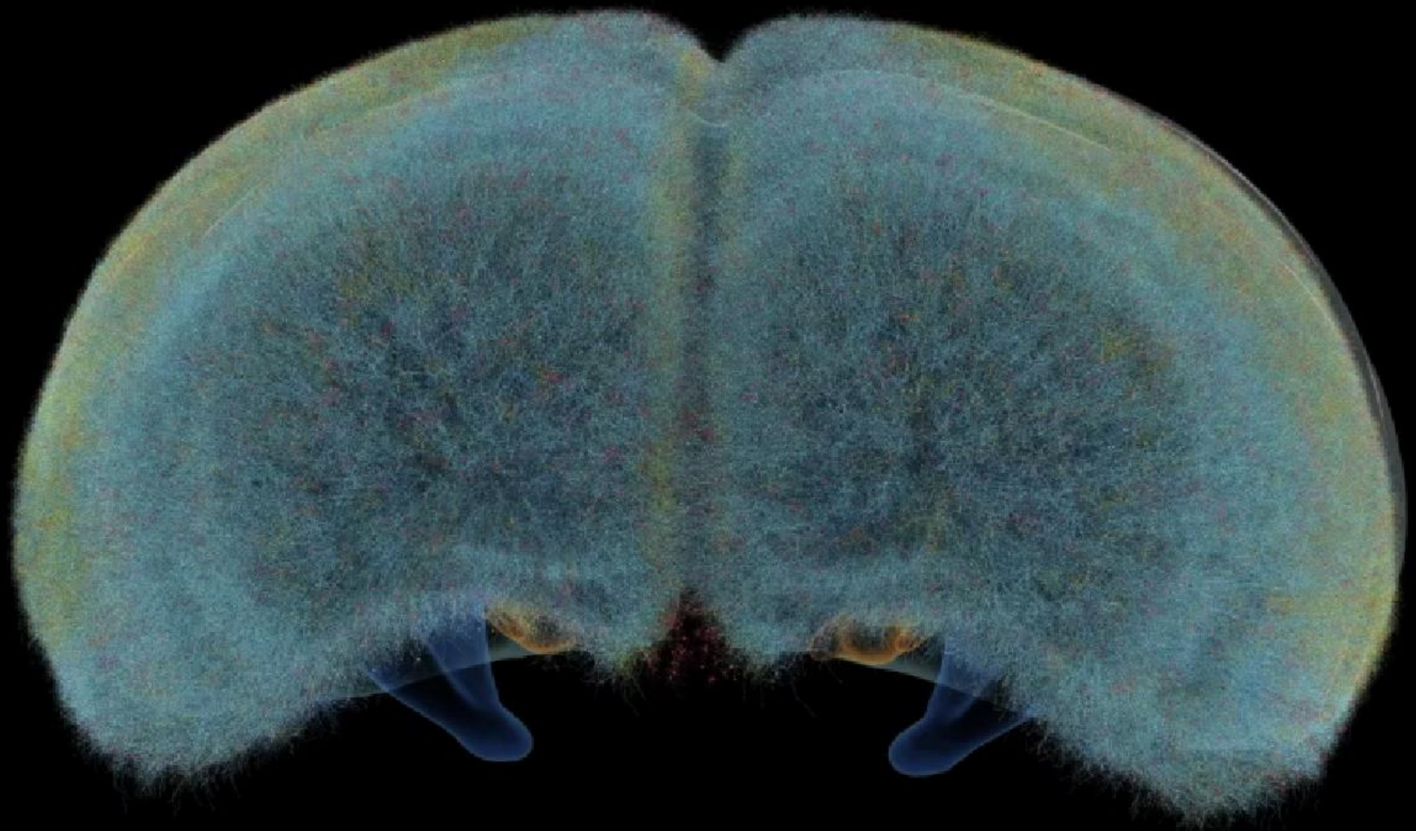
- Create variations of neuron types
- Populate volume according to biological type distribution
- Establish connectivity

An example of morphologically detailed Neocortex circuit

- **#Cells** : ~9.3 million
- **#Compartments** : ~3.5 billion
- **#Synapses** : ~145.8 billion

- **#Channel types** : ~18

- **Total memory requirement for simulation:** ~169 TB

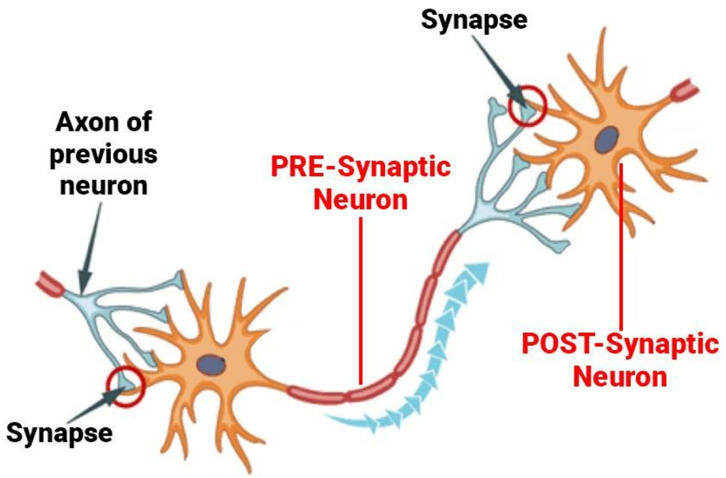


TouchDetector

Assembling circuits using distributed computing

Neurons, Morphologies and Synapses

- **Neuron:** nerve cell
- **Morphology:** physical shape of a neuron
  - Central part (**soma**) can be represented as a sphere
  - Branches (**axon**, **dendrite**) are simplified to sequences of cylinders
- **Touch:** region of physical proximity between neuron
- **Synapse:** punctual chemical or electrical connection between neurons



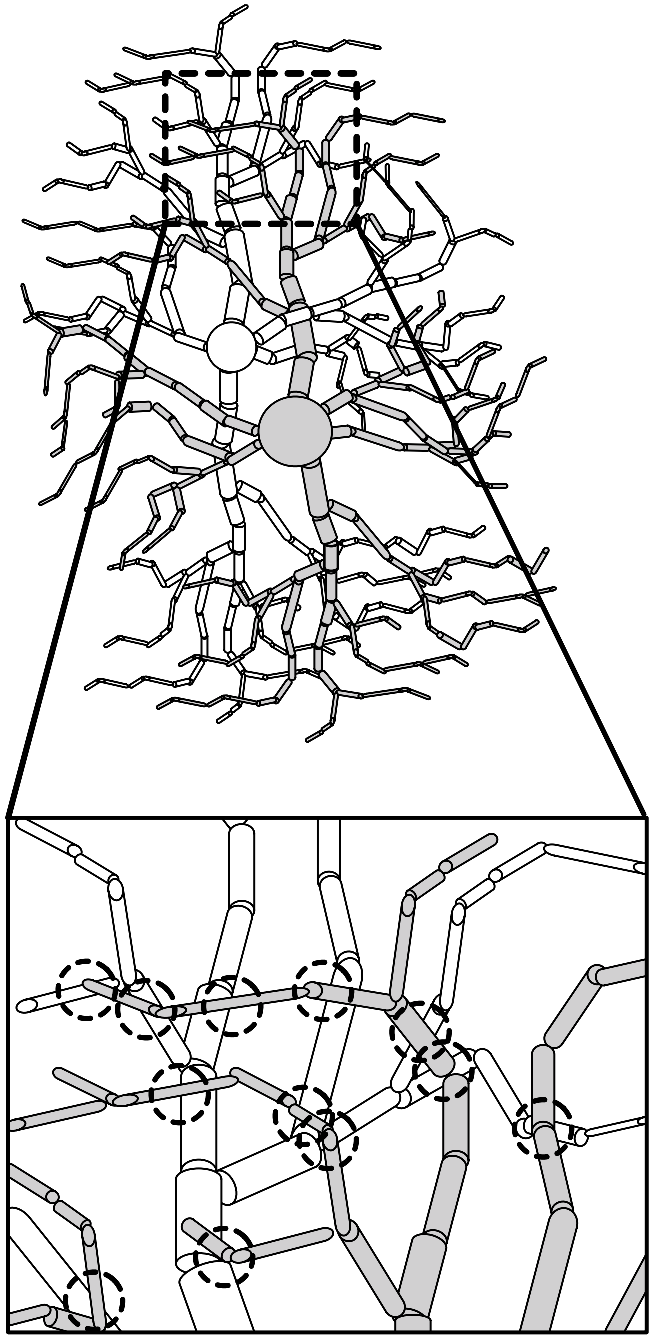
source: https://bit.ly/3gqau6V

Building the Neural Connectome

Connections between the branches of neurons:

- Model the branches of neurons as sequences of cylinders
- Some branches have **only outgoing** connections
- Some branches have **only incoming** connections

Every connection is based on cylinder overlap and saved as the projection on the cylinder axis.



Building the Neural Connectome

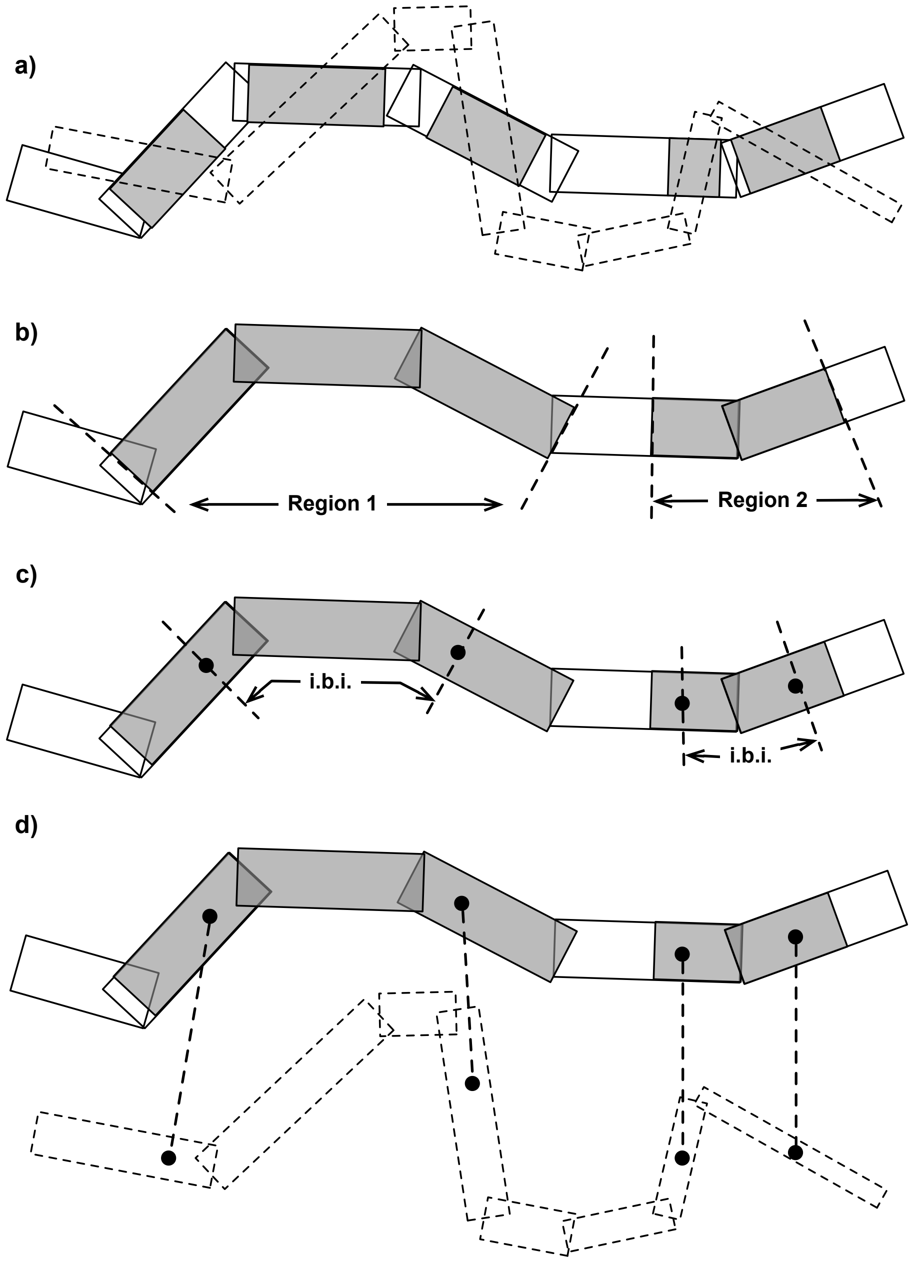
Cylinders can be very uneven in size:

- Can result in dense clusters based only on representation

- Post-process connections between two branches

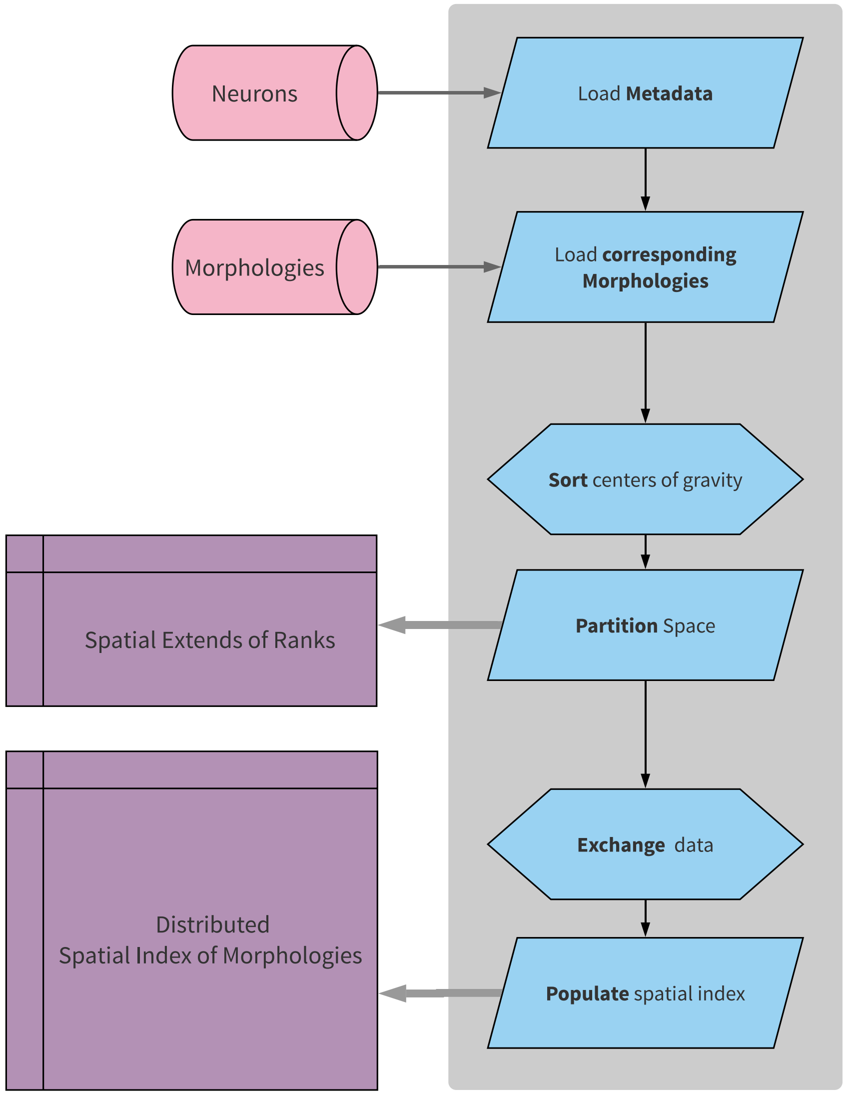
- Re-distribute connections to match biological spacing

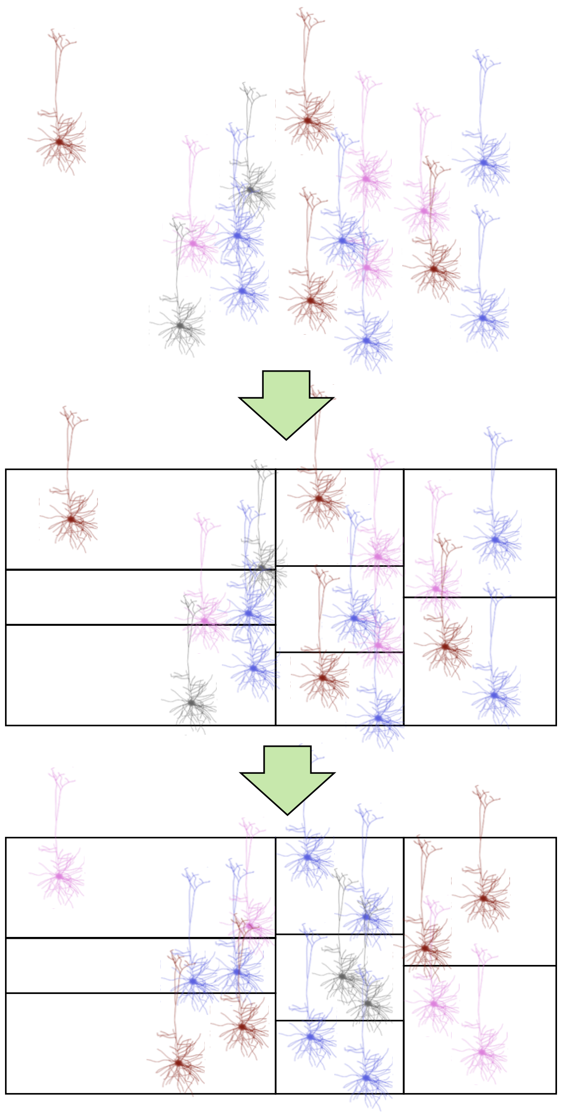
Requires collecting all touches between two neurons



Scaling up: Distributed Spatial Index

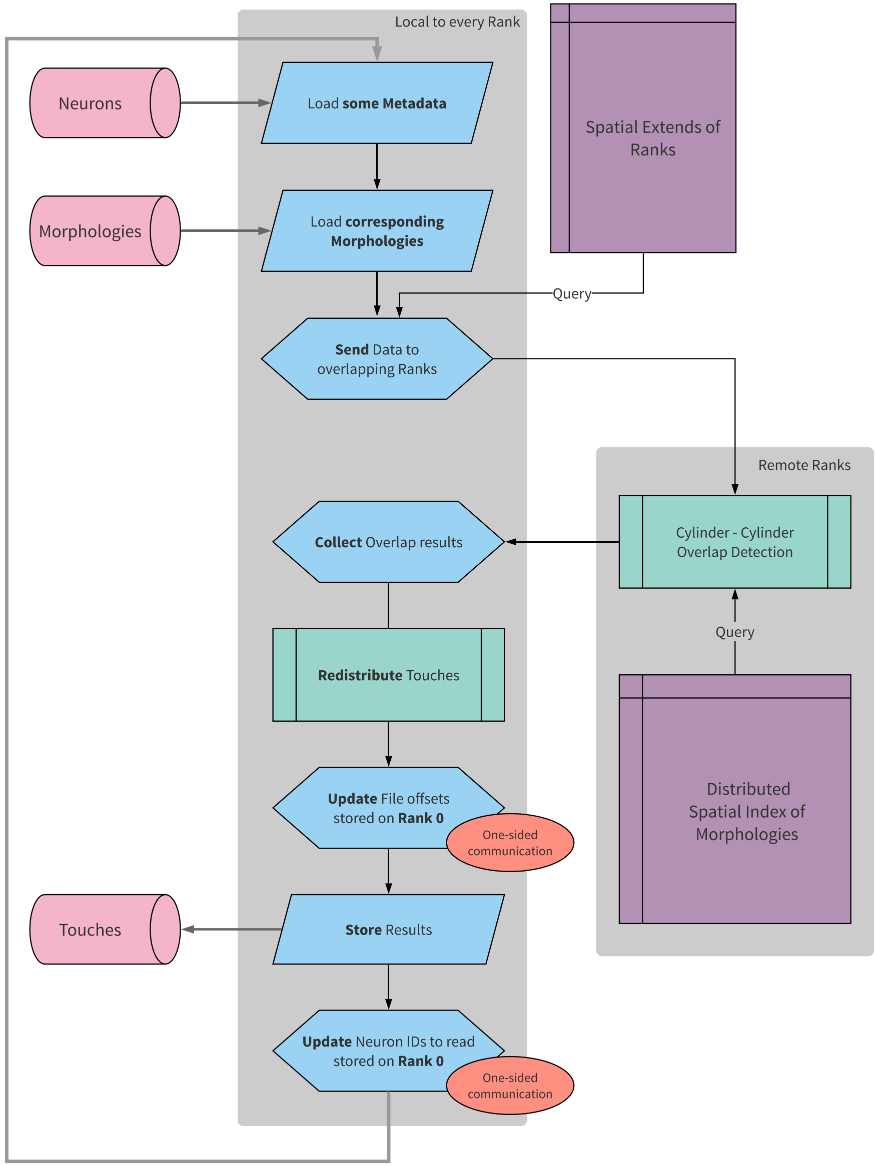
- Load neuron metadata
- Load shape information
- Shift and rotate shapes
- Sort branches, assign to ranks
- Transfer data to ranks

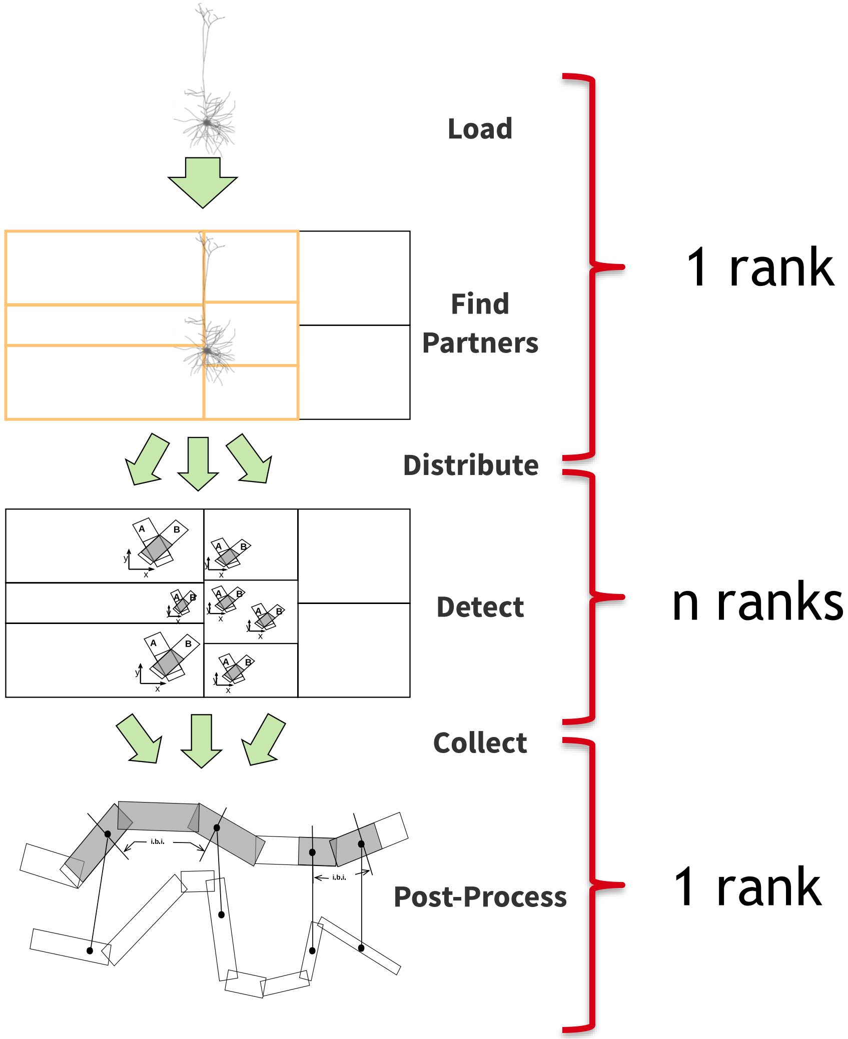




Scaling up: Neuron Overlap

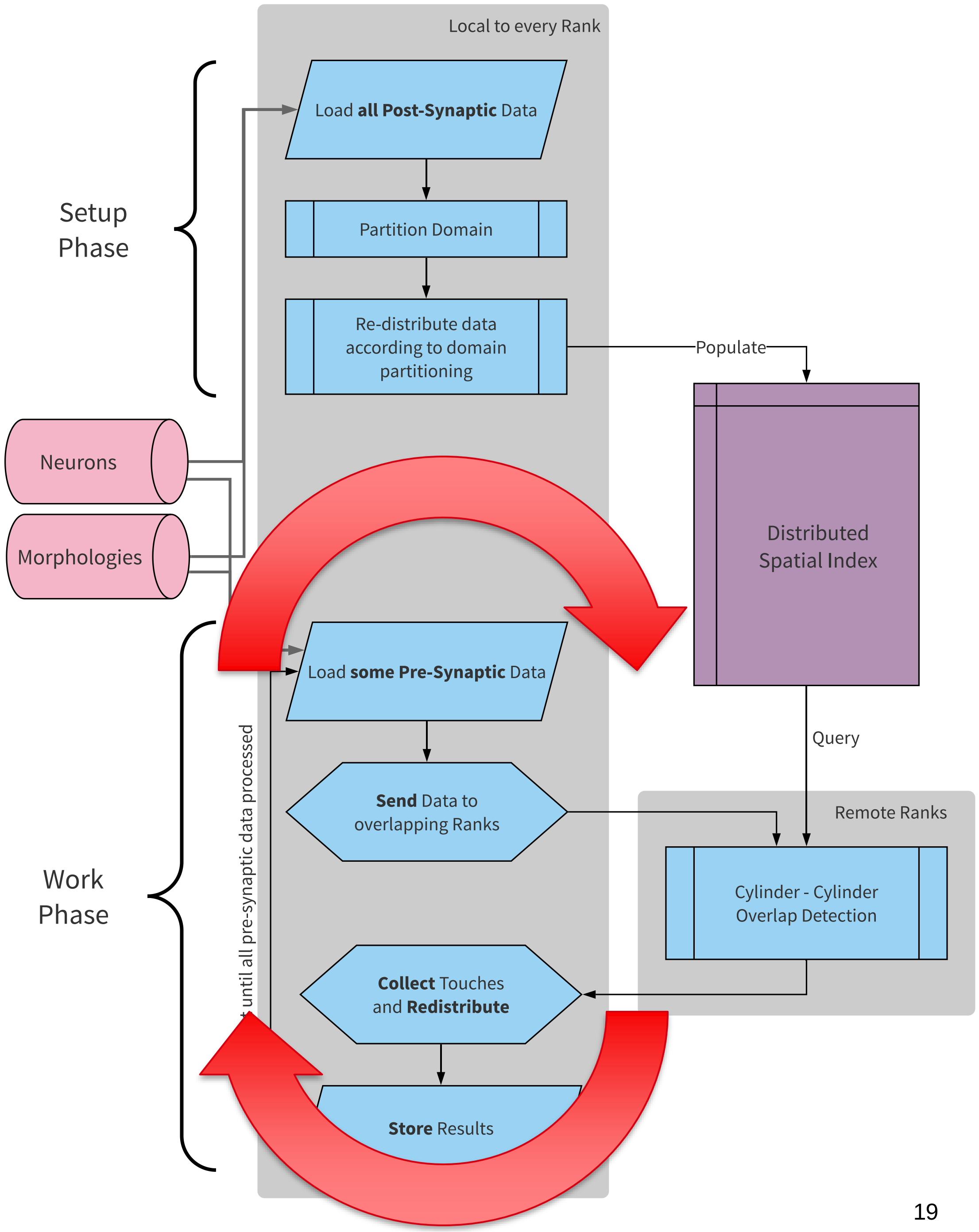
- Process batches
- Every rank does the same
- Overlapping communication





Algorithm Summarized

1. Load one side of all neurons
   - Create a distributed spatial index
2. Load batches of the other side of neurons
   - Remote overlap detection
   - Collect data
   - Redistribute
   - Write to disk



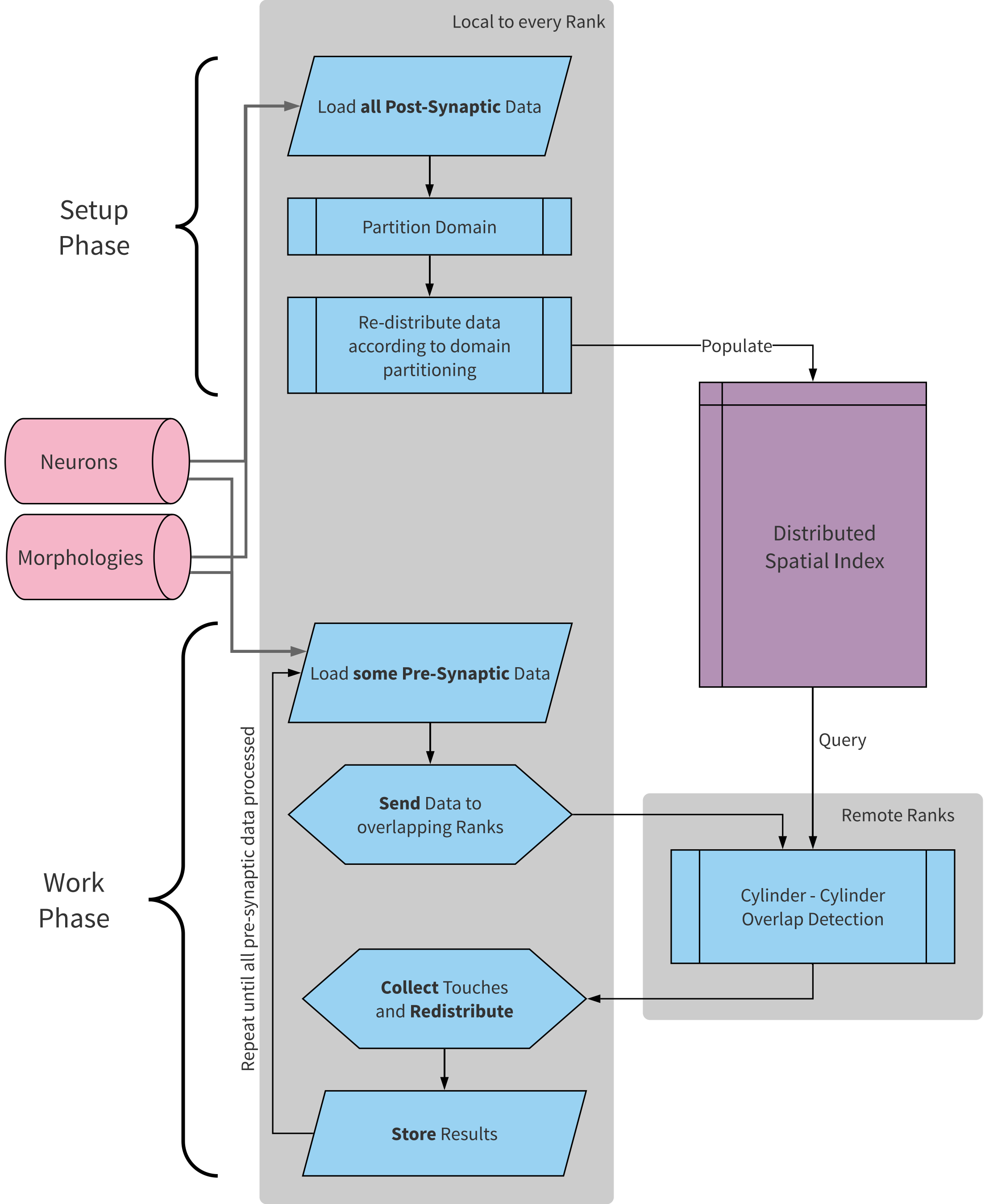
MPI Communication patterns

Collective Operations

- `MPI_Allreduce`
- `MPI_Allgather(v)`
- `MPI_Alltoall(v)`

Individual Operations
(mostly async)

- `MPI_(I)Probe`
- `MPI_Recv`
- `MPI_Isend`
- `MPI_File_write_at`
- `MPI_Win_(un)lock`
- `MPI_Fetch_and_op`



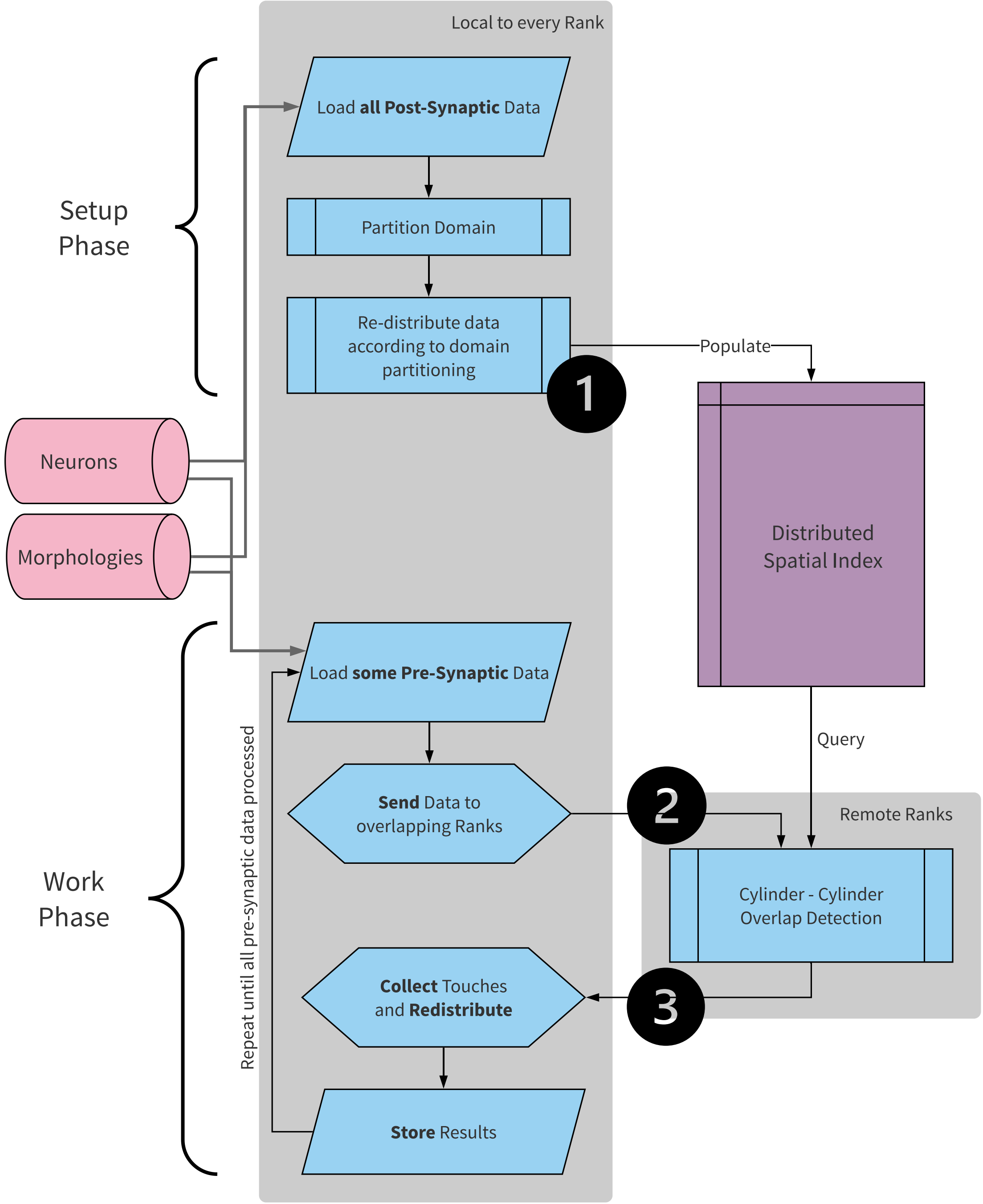
MPI Communication Challenges

Large amounts of data transferred
e.g. for 10 mio neurons:

1. ~2 TB total, ~500 Mb per rank via all-to-all
2. ~60 TB total distributed to other ranks
   Average message size ~0.2 MB
3. ~100 TB total collected from other ranks

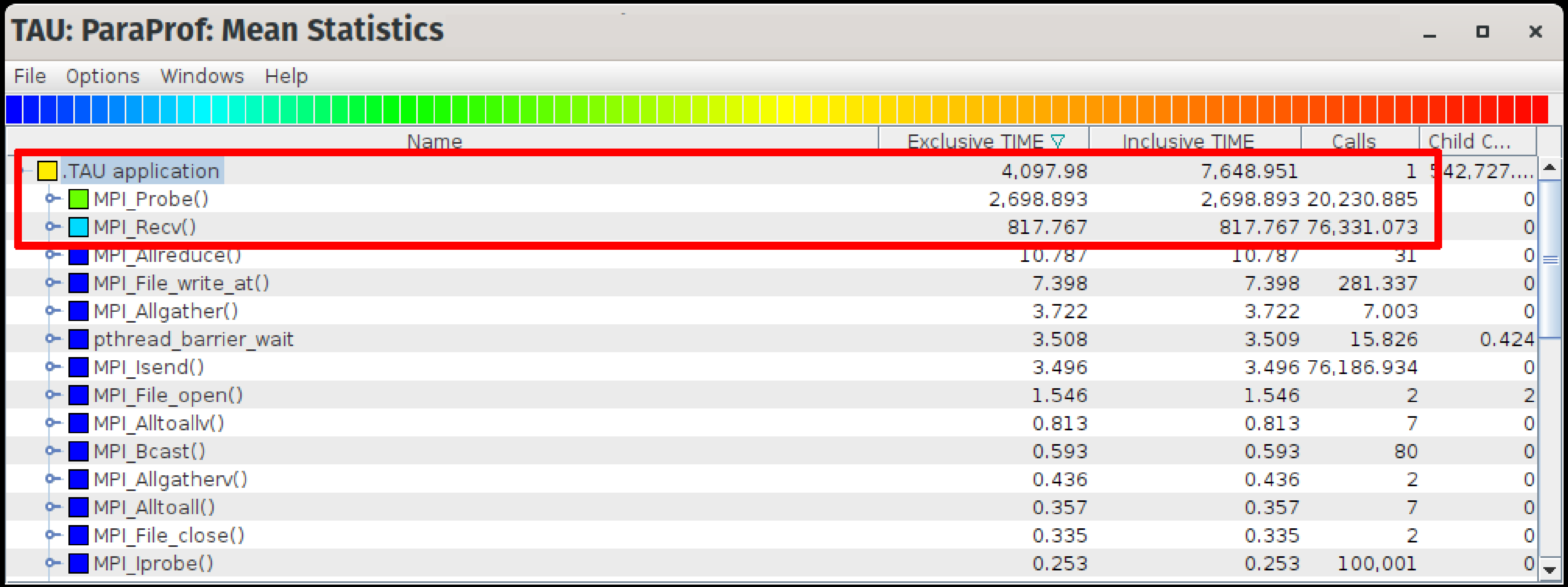
Communication pattern is not fixed after setup

- Need to send data to 10-40% of the ranks
- Partners vary depending on data loaded, spatial index location



Profiling communication

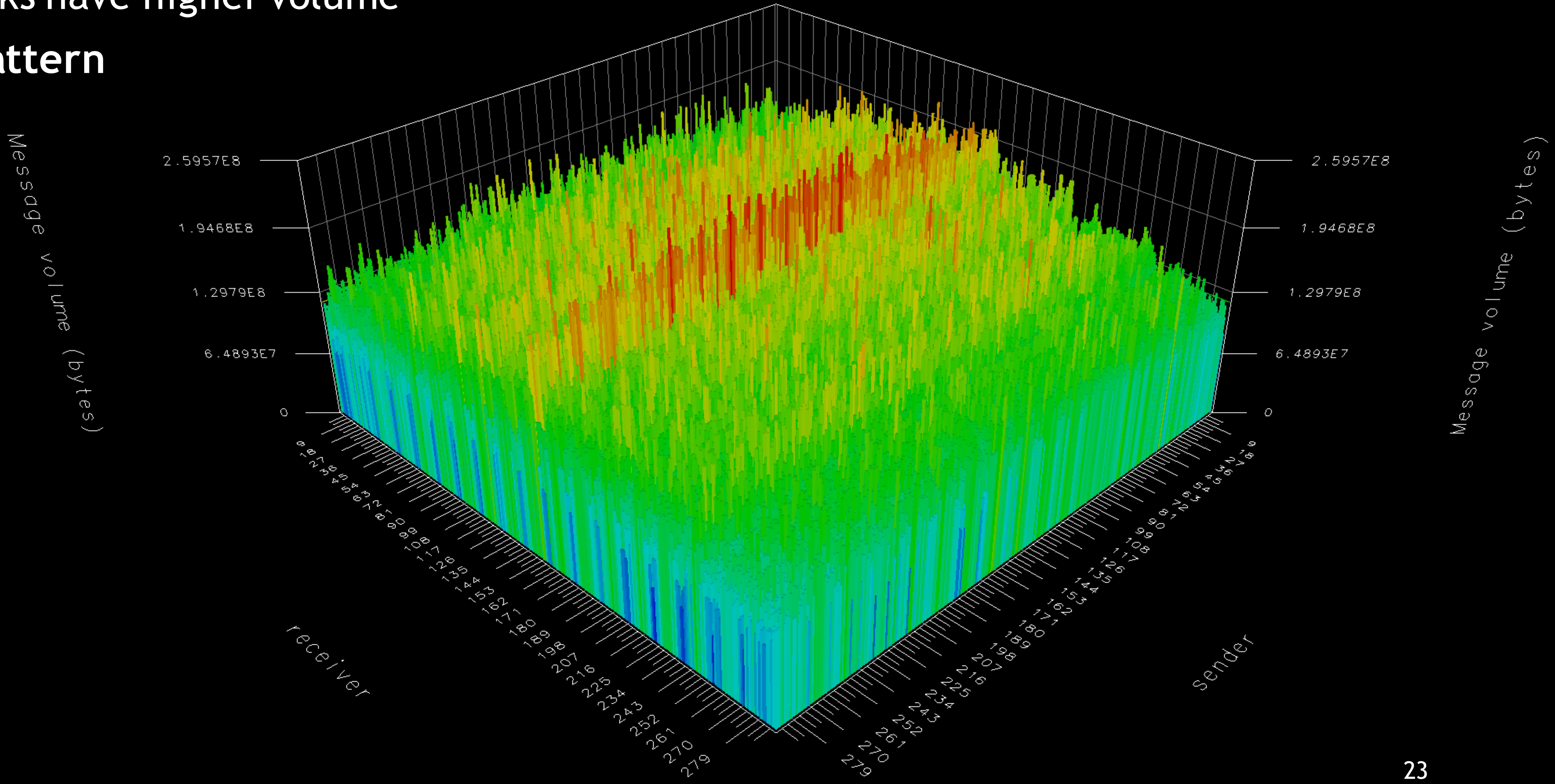
- MPI calls take up a sizeable amount of time
- **> 50% of time consumed within MPI**
- **~ 20% just receiving data**



Name	Exclusive TIME	Inclusive TIME	Calls	Child C...
.TAU application	4,097.98	7,648.951	1	542,727....
MPI_Probe()	2,698.893	2,698.893	20,230.885	0
MPI_Recv()	817.767	817.767	76,331.073	0
MPI_Allreduce()	10.787	10.787	31	0
MPI_File_write_at()	7.398	7.398	281.337	0
MPI_Allgather()	3.722	3.722	7.003	0
pthread_barrier_wait	3.508	3.509	15.826	0.424
MPI_Isend()	3.496	3.496	76,186.934	0
MPI_File_open()	1.546	1.546	2	2
MPI_Alltoallv()	0.813	0.813	7	0
MPI_Bcast()	0.593	0.593	80	0
MPI_Allgatherv()	0.436	0.436	2	0
MPI_Alltoall()	0.357	0.357	7	0
MPI_File_close()	0.335	0.335	2	0
MPI_Iprobe()	0.253	0.253	100,001	0

Communication Volume

- Intrinsic imbalance
- Central ranks have higher volume
- **No clear pattern**



Algorithmic Abstraction

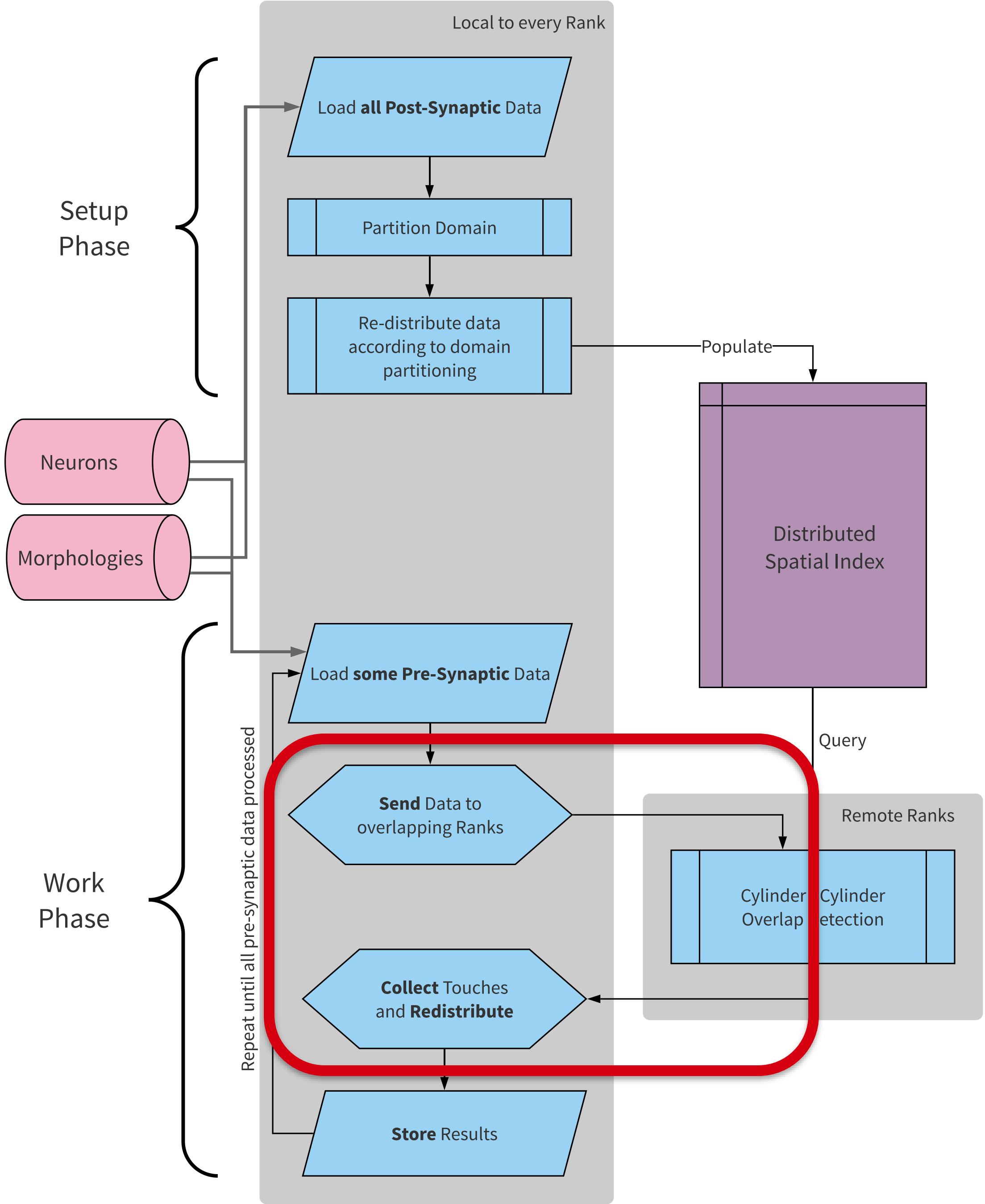
Whatever should go here?

MPI Communication Challenges

Extract the essential data transfer pattern

- Need to send data to 5-40% of the ranks
- Partners vary depending on data loaded, spatial index location

Data transferred and partners vary depending on problem size and number of ranks



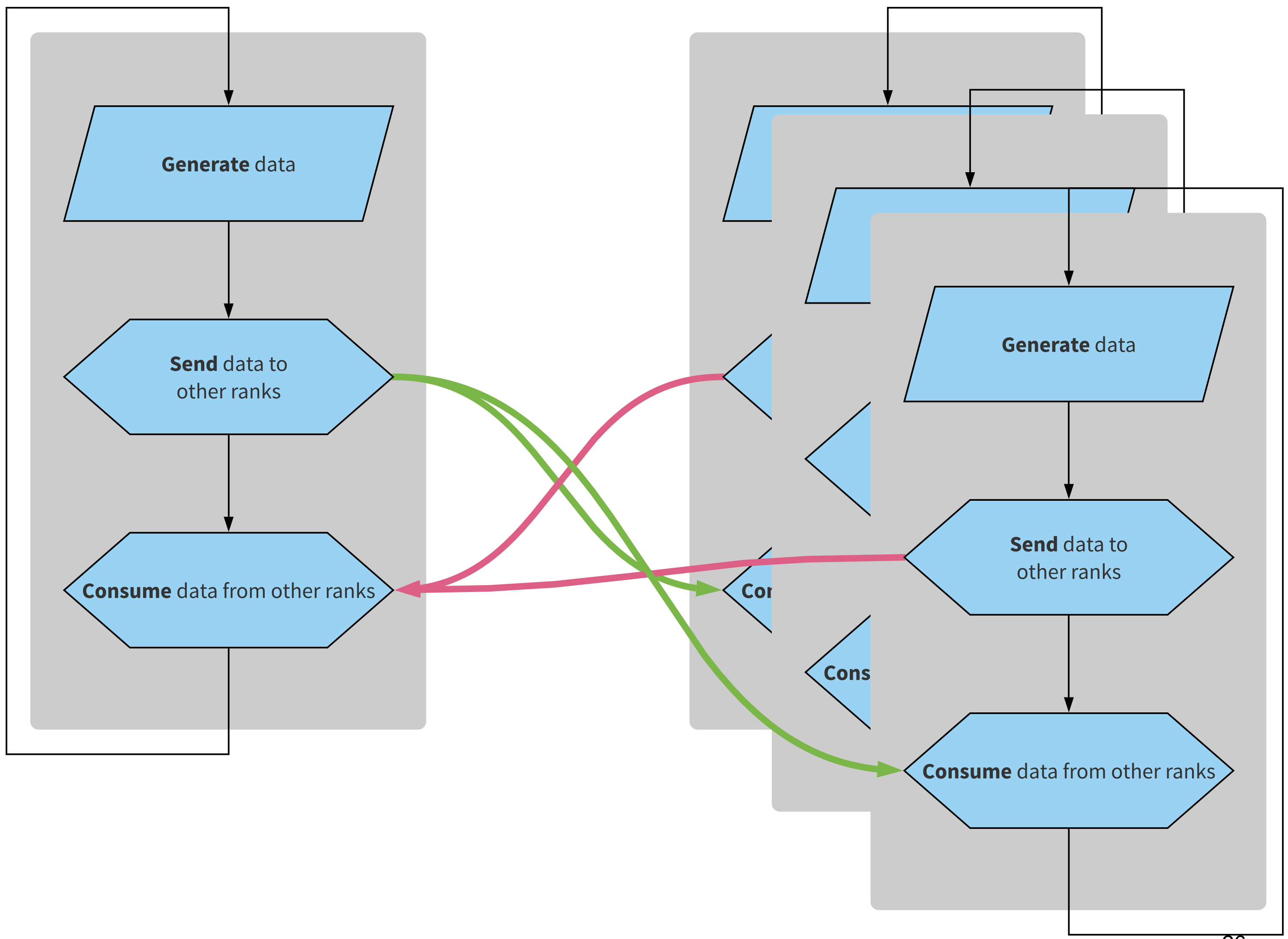
Simplifying the Communication Pattern

Representative abstraction

- Remove input data dependence
- Remove one-sided MPI calls
- Send data, no computations

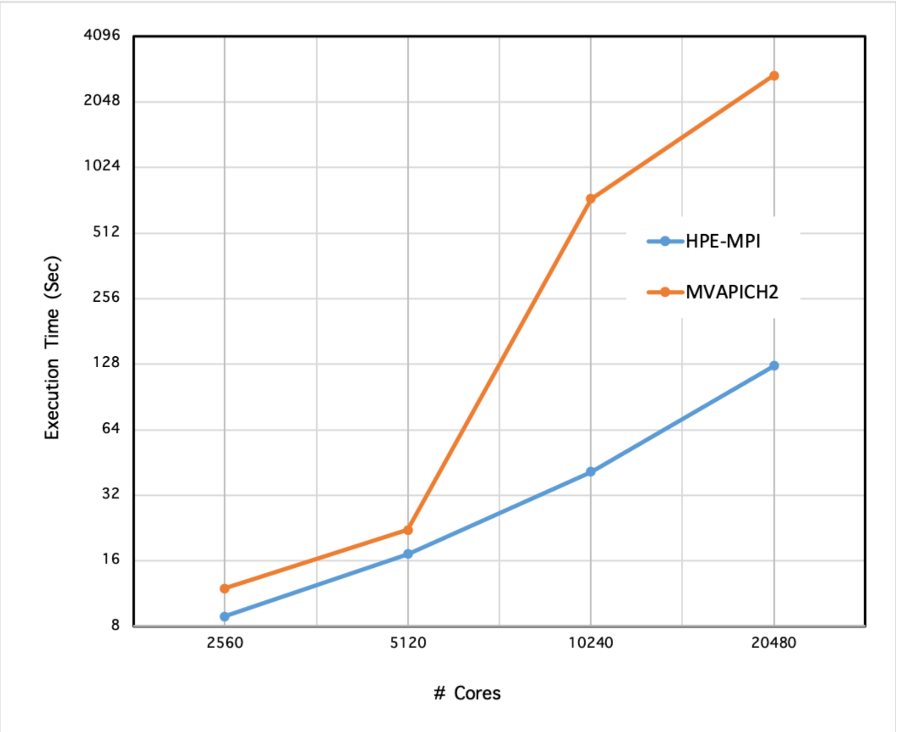
Knobs to tune

- Fraction of ranks
- Randomization
- Data size to be sent

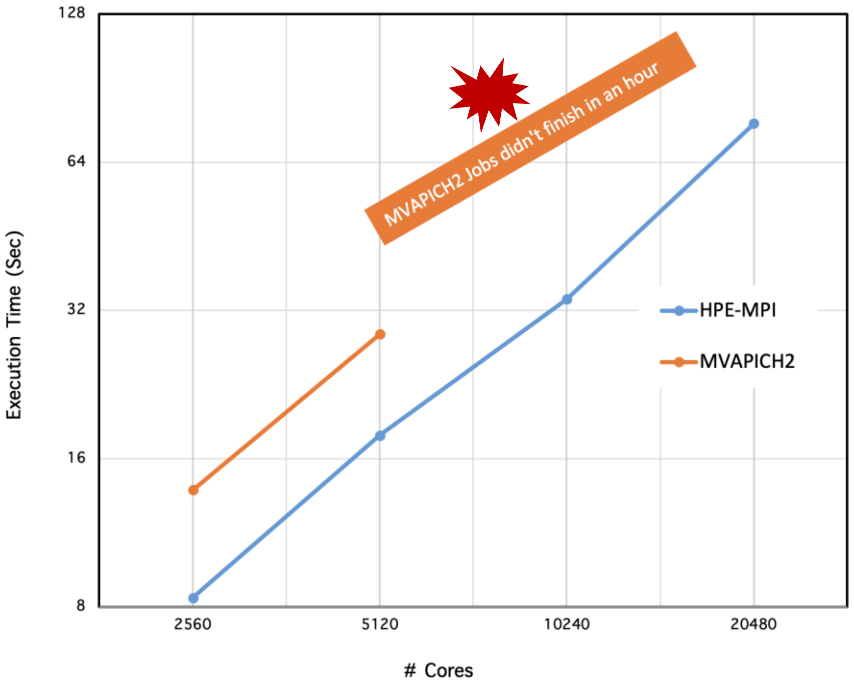


First performance numbers

- **With fixed partners :** >=256 nodes performance was significantly slower than vendor MPI
- **With random partners :** even more penalty if communication partners change every iteration



Fixed Partners

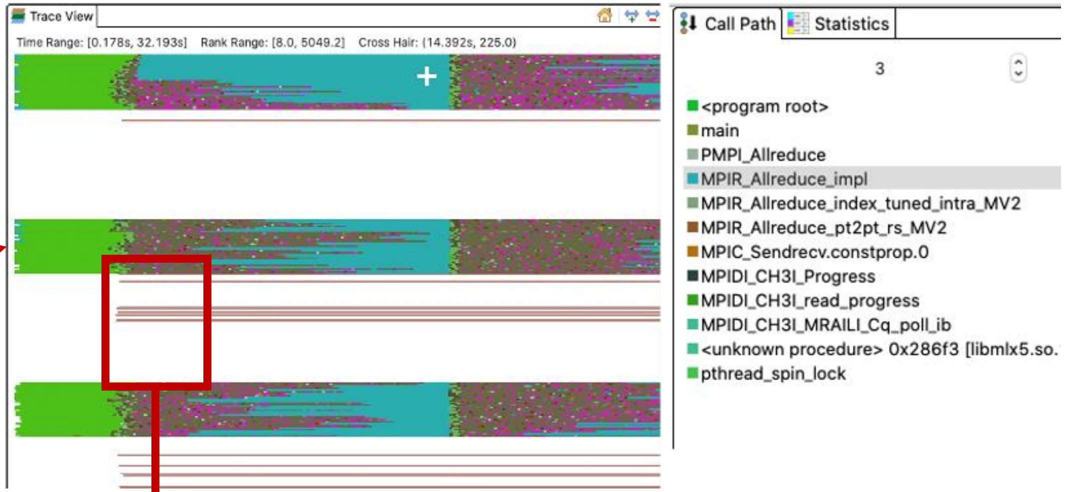


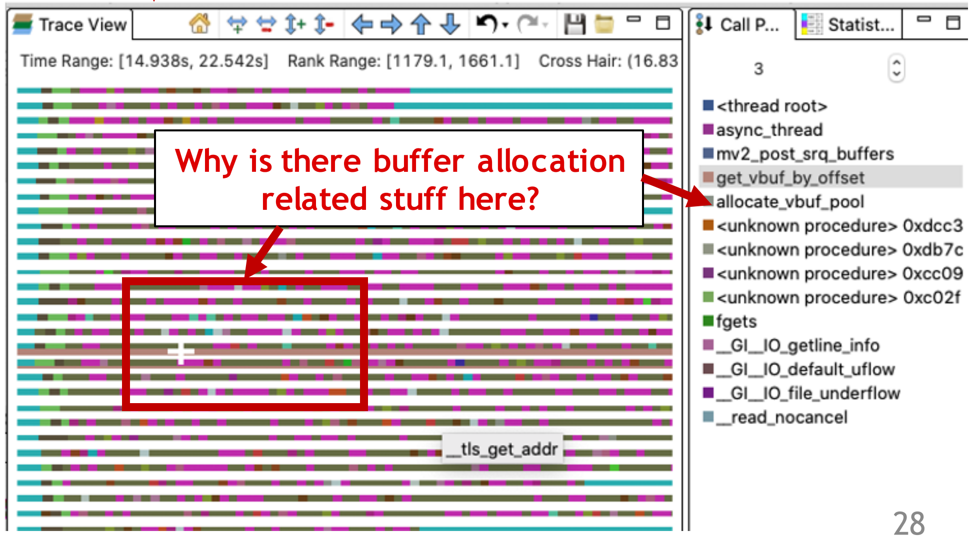
Random Partners

Why is the performance so low?

- Very unbalanced communication costs during runtime
- Allreduce calls showing high cost but doesn’t make sense

Timeline overview of all MPI processes





Performance after initial fixes / tuning

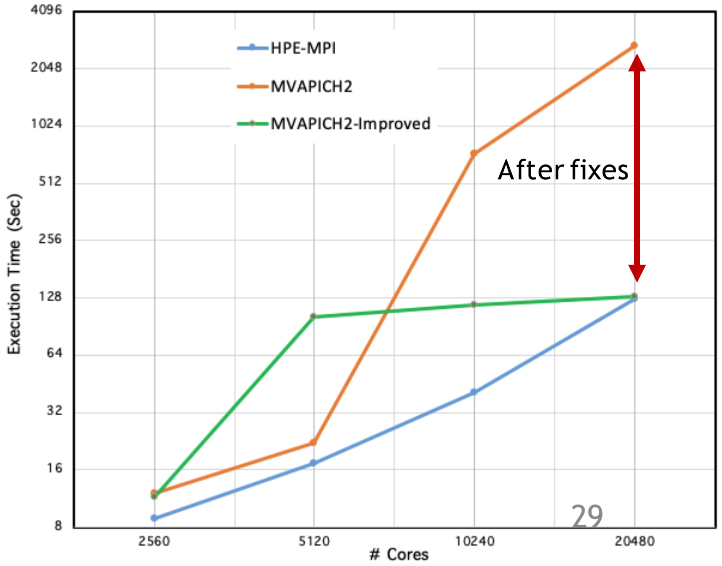
- New buffer registration during every iteration – **performance degradation with buffer registration!**

```
for(……….) {
    std::vector<double> data(bufsize);
    ….
    MPI_Isend(data.data(), …..);
}
```

```
std::vector<double> data(bufsize);
for(……….) {
    ….
    MPI_Isend(data.data(), …..);
}
```

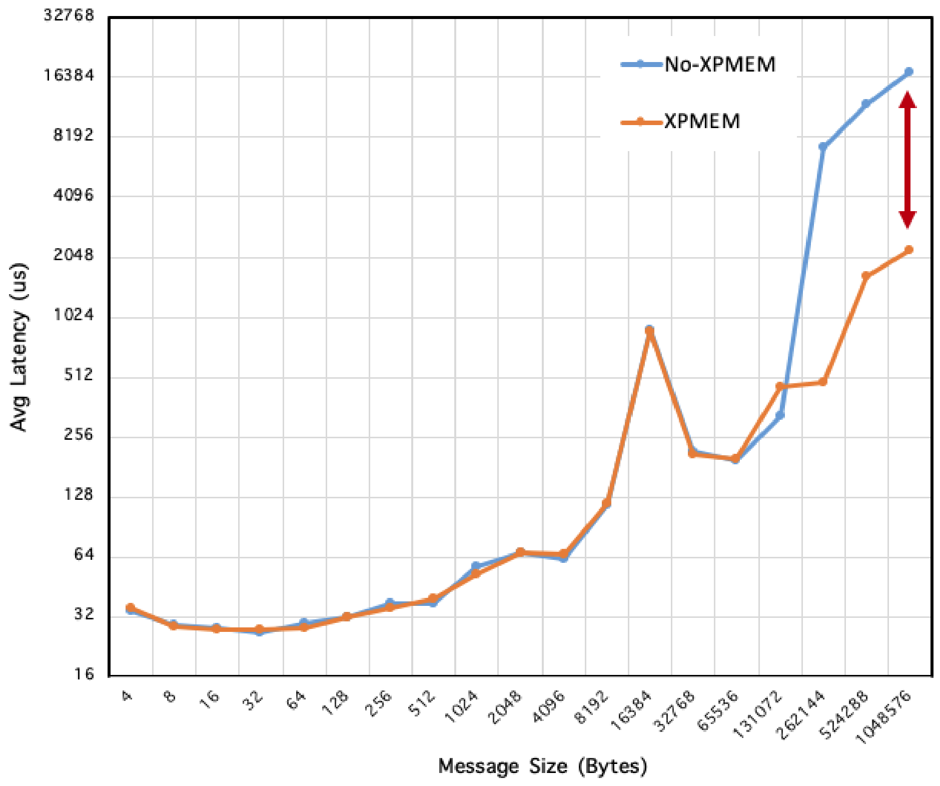
- Pre-allocation of buffer could solve the problem but <u>in practice we need dynamic buffer sizes</u>

- New version of MVAPICH2-X with fixes now provides consistent behaviour



Improvements for collectives

- HPE-MPT uses proprietary implemetation of xpmem
- Different from Cray implementation which used by other MPI implementations including MVAPICH2
  - But same kernel module loaded on the nodes
  - Can HPE's xpmem and Cray's xpmem active on the same node?
- Currently activated Cray's xpmem implementation on subset of machine for testing



Comparison of MPI_Allreduce latency on 256 nodes (40ppn, 10,240 ranks)

Summary

- Heavy reliance on MPI communications

- Problematic variance in data sizes and communication partners

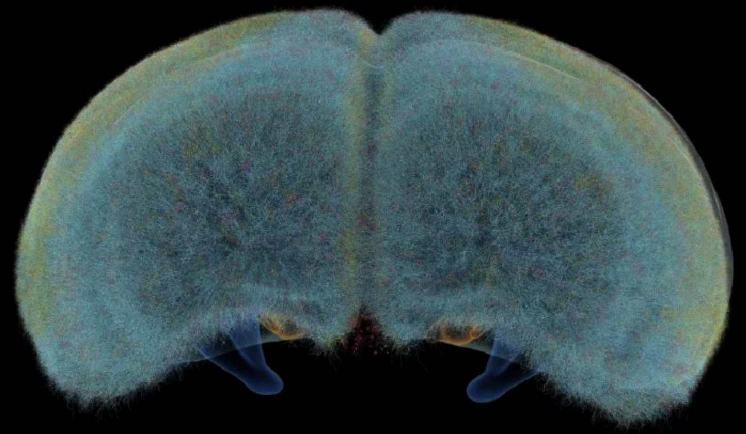
- MVAPICH2 is a viable alternative to HPE-MPI
  - Requires tuning via environment variables
  - Setup dependent on use-case?
  - Strong support for better performance

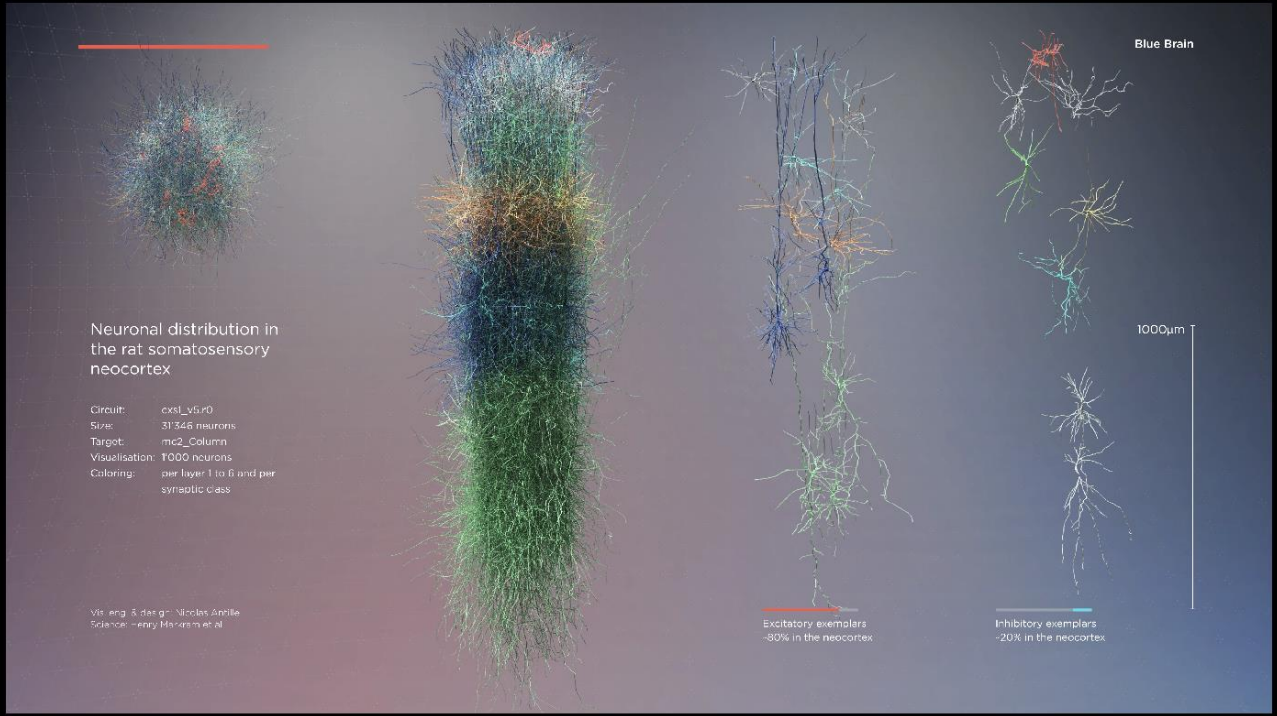
Future Work

- Complete benchmarking communication patterns, tuning of parameters

- Resume testing MVAPICH with the full version of TouchDetector

- Test integration with IME burst buffer in real-world scenario





Internships as well as full time positions in Scientific Computing!

https://go.epfl.ch/bluebrain-careers / **Email Us !**

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Blue Brain Project

Digitally reconstructing and simulating the brain


Thank you!