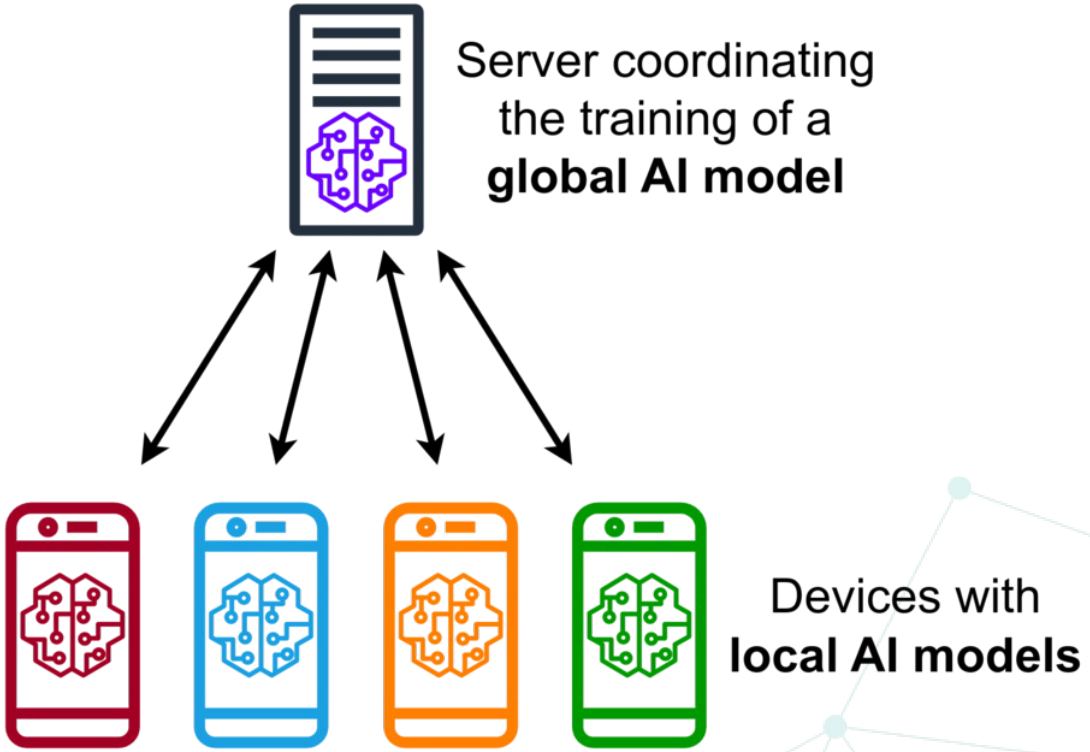
FaLaFEℓS: A tool for Estimating Federated Learning Energy Consumption via Discrete Simulation

Andrew Mary Huet de Barochez
Stéphan PLASSART
Sébastien MONNET

The 39th International Conference on Advanced Information Networking and Applications (AINA-2025)

Federated Learning (FL)

- Introduced by B. McMahan et al. In 2017
- No need to move data
  - Data privacy advantage
  - Bandwidth usage reduction



State-of-the-art topologies

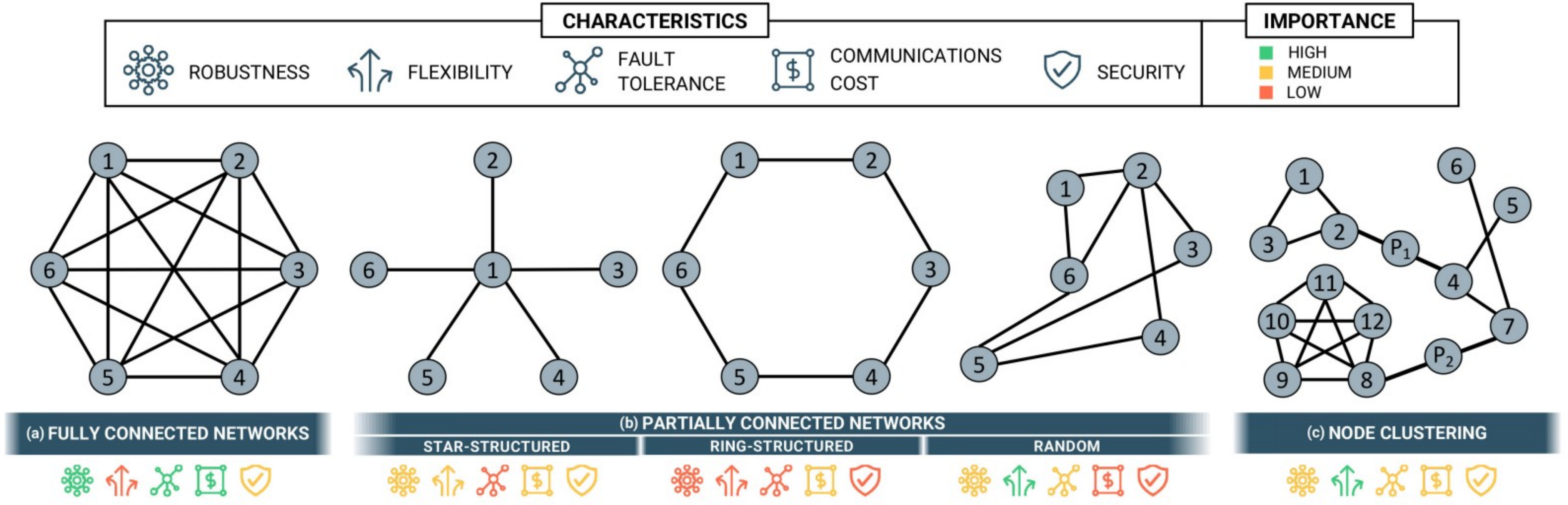


Fig. 6. DFL network topologies. (a) Fully connected networks, (b) Partially connected networks (star-structured network, ring-structured network, random network), (c) Node clustering. P_n indicates the node acting as a proxy.

Beltrán, Enrique Tomás Martínez, et al. "Decentralized federated learning: Fundamentals, state of the art, frameworks, trends, and challenges." IEEE Communications Surveys & Tutorials (2023).

How can we **compare** the **energy efficiency** of **state-of-the-art** implementations?

Related works

Analytical models

Pilla, L.L.[2023], Savazzi, S.[2022], Qiu, X.[2023]

Energy consumption is **estimated** via formulas

✗ Limited expansibility

✗ Lack of systems considerations

Related works

Analytical models

Pilla, L.L.[2023], Savazzi, S. [2022], Qiu, X.[2023]

Energy consumption is **estimated** via formulas

✗ Limited expansibility

✗ Lack of systems considerations

Experimental frameworks

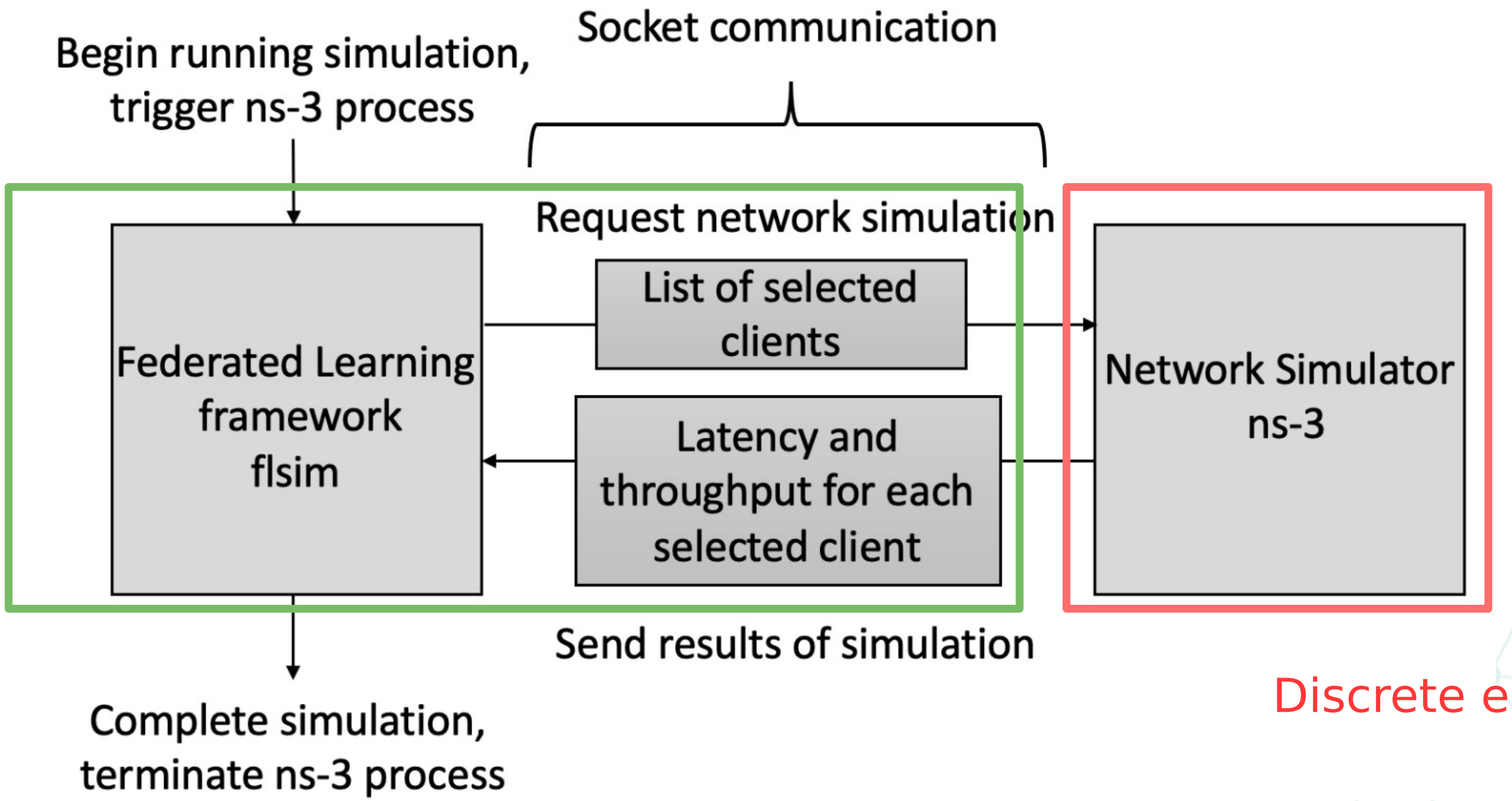
Beutel, D.J.[2020] (*Flower*), Ekaireb, E. [2022] (*ns3-fl*)

Energy consumption is **measured**, either with wattmeters or software (RAPL, NVML…)

✗ Constrained by machine learning training time

✗ Benchmarking involves actual consumption

Related works: ns3-fl Ekaireb, E. et al.[2022]



Real execution

Discrete environment

Our contribution: Fully discrete FL simulation

- ✔ Reproducible
- ✔ Fast run-time
- ✔ Fast development
- ✔ Try any algorithm on any platform for free

Federated Learning scenario

↓

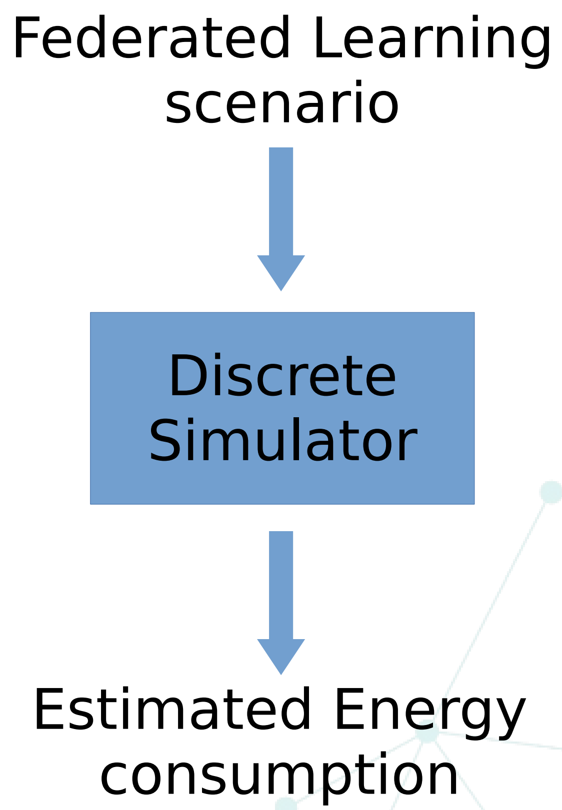
Discrete Simulator

↓

Estimated Energy consumption

Our contribution: Fully discrete FL simulation

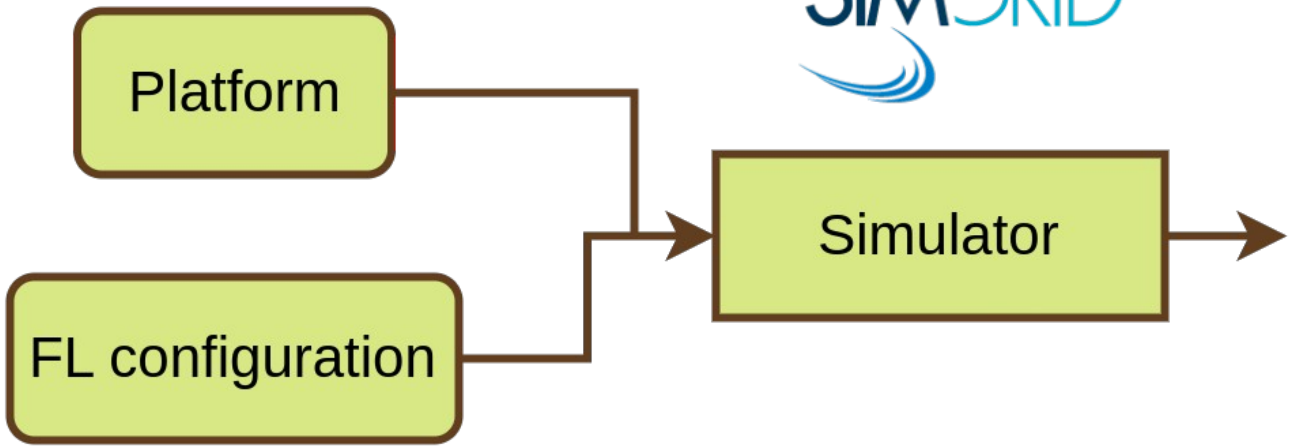
- ✔ Reproducible
- ✔ Fast run-time
- ✔ Fast development
- ✔ Try any algorithm on any platform for free
- ✘ Requires some calibration
- ✘ Learning accuracy unknowned



FaLaFEℓS Federated Learning Frugality and Efficiency via Simulation

- Number of machines
- Types of machines (with different speed and consumption)
- Network topology

Powered by SIMGRID

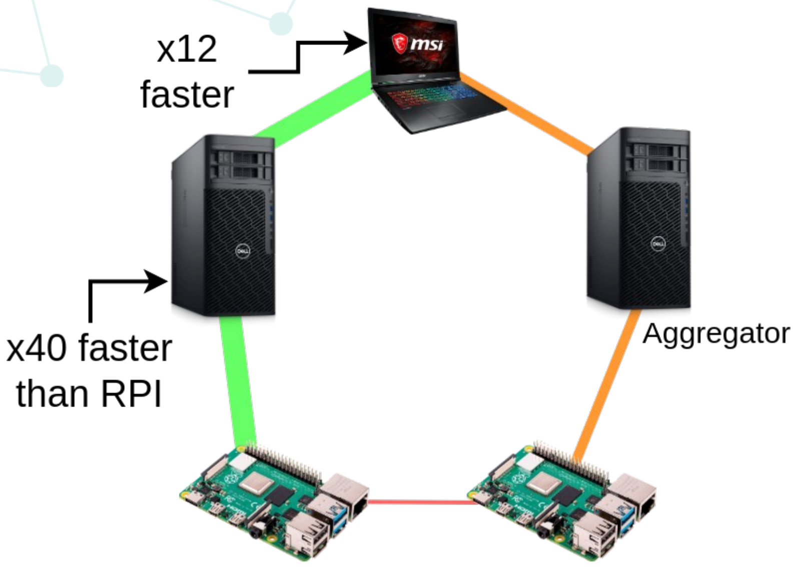


- Runtime
- Energy consumption

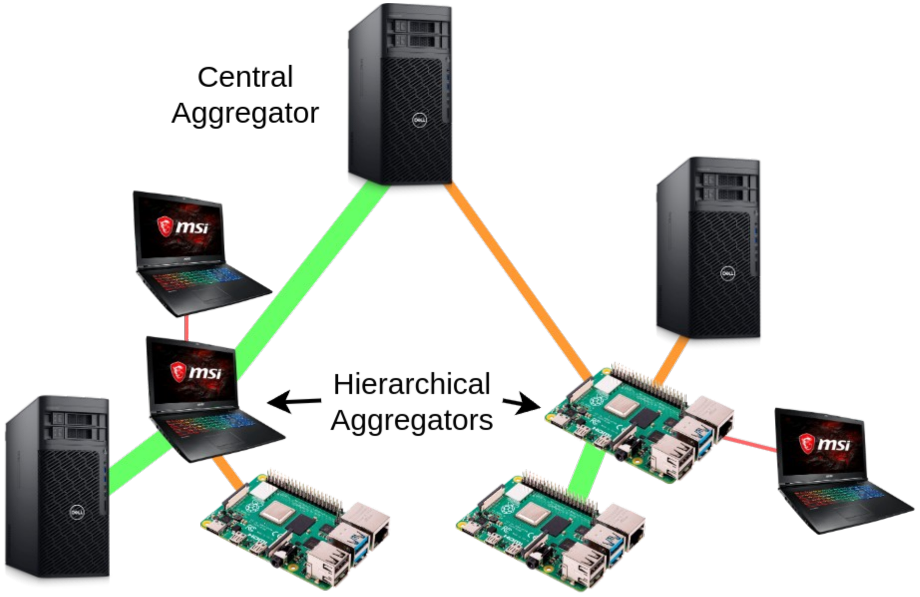
- ML Model
- FL implementation
- Applicative topology

Simulated topologies

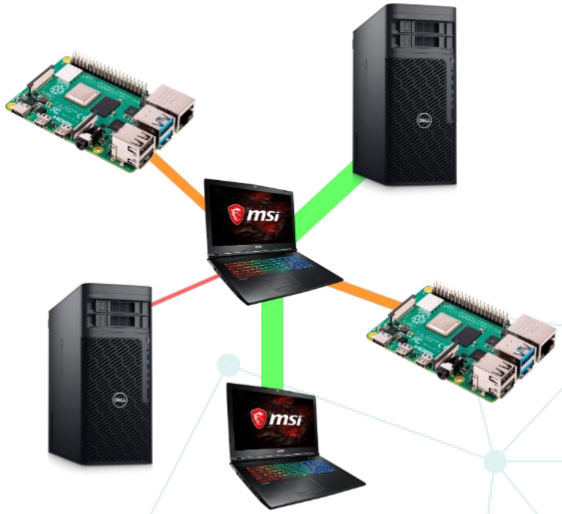
Scenario 1



Ring topology



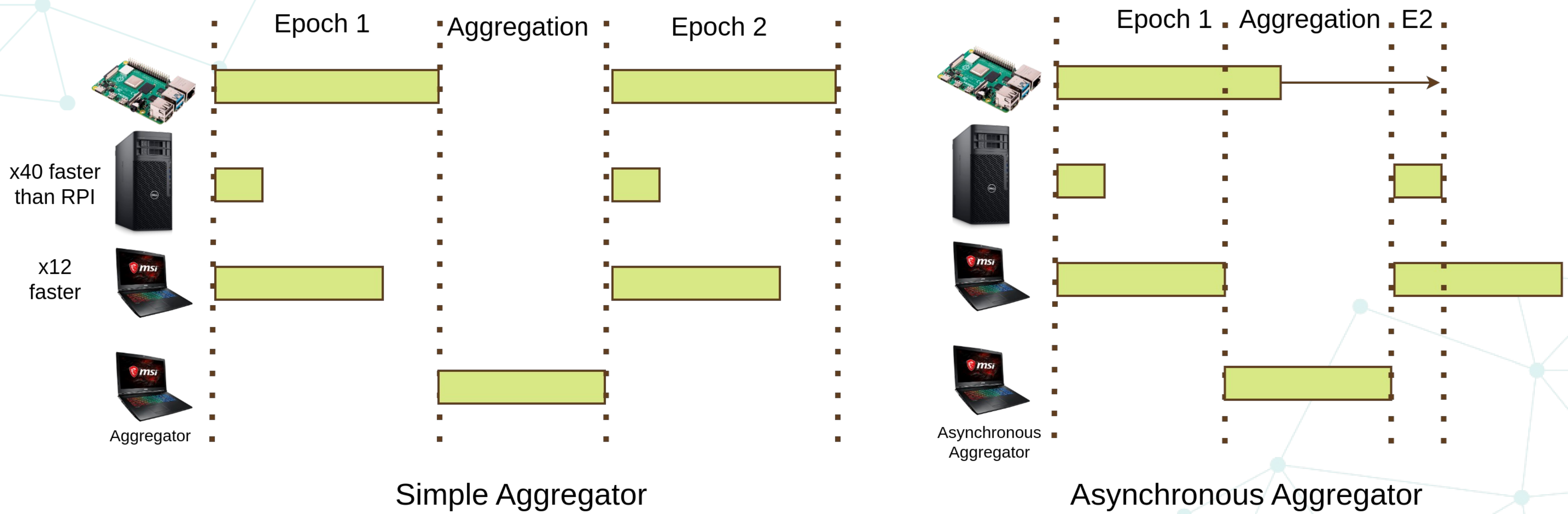
Hierarchical + star
subclusters



Star topology

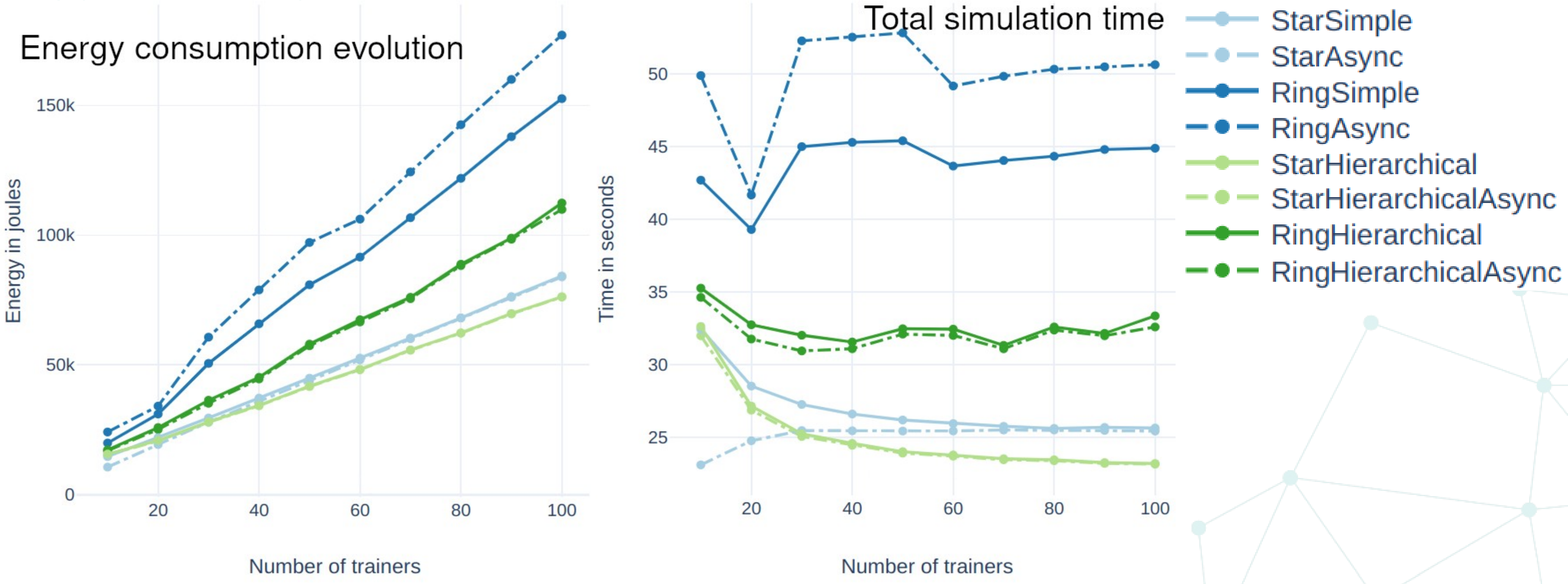
Simulated algorithms

Scenario 1



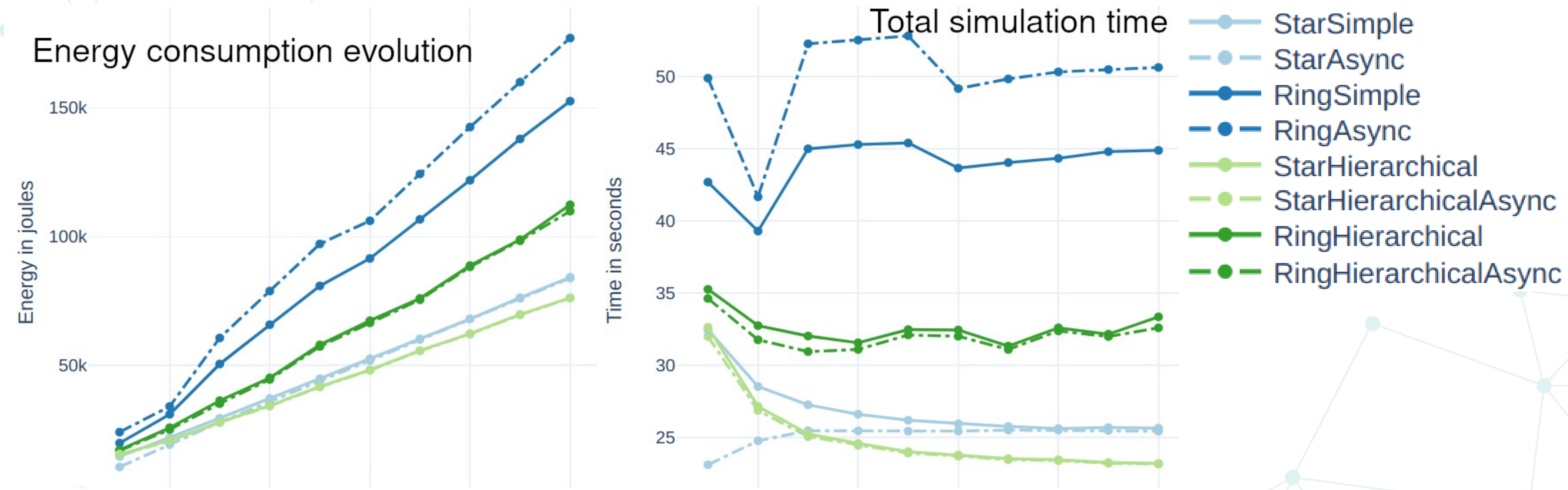
Preliminary results

Scenario 1



Preliminary results

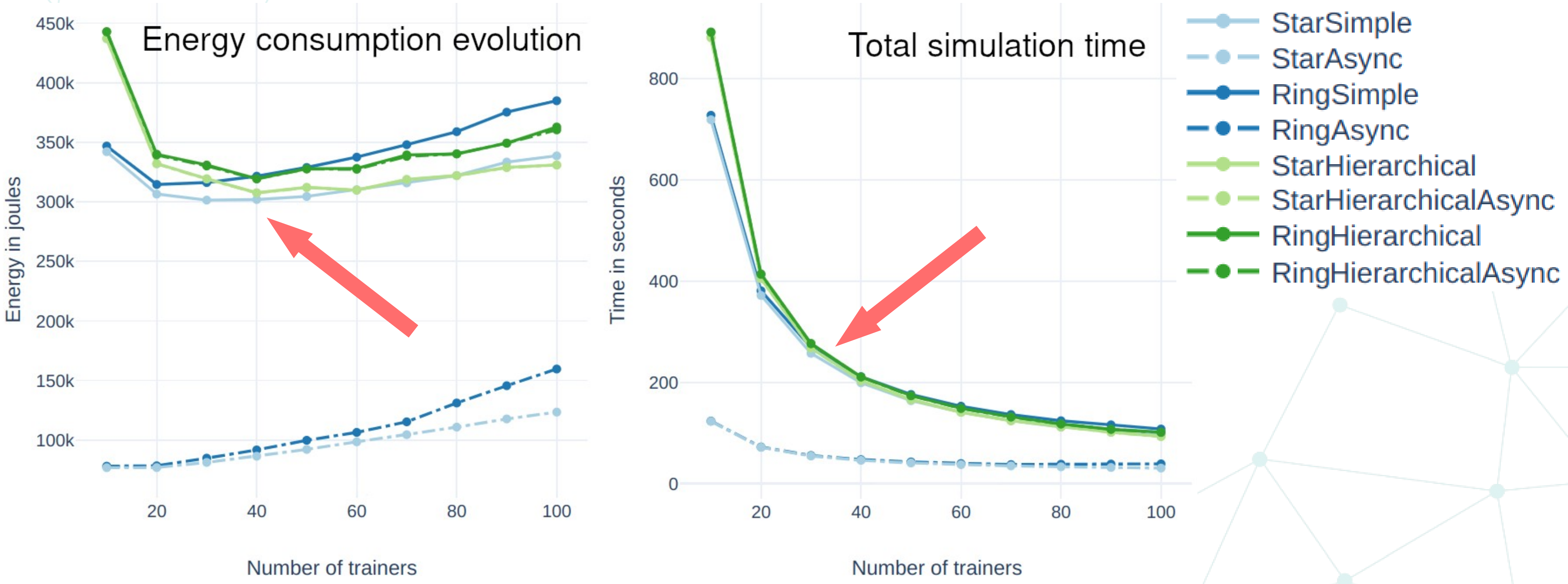
Scenario 1



Increasing the **number of machines** won’t reduce the energy consumption if **parallelization costs are higher than the workload itself**

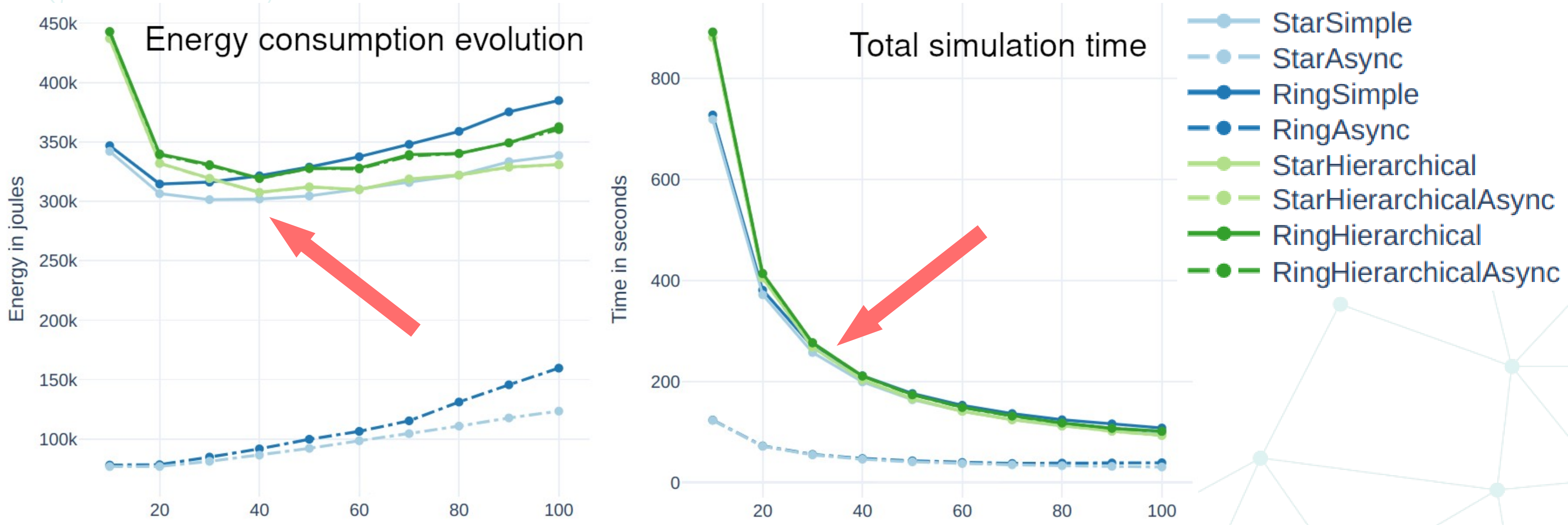
Increased workload

Scenario 1



Increased workload

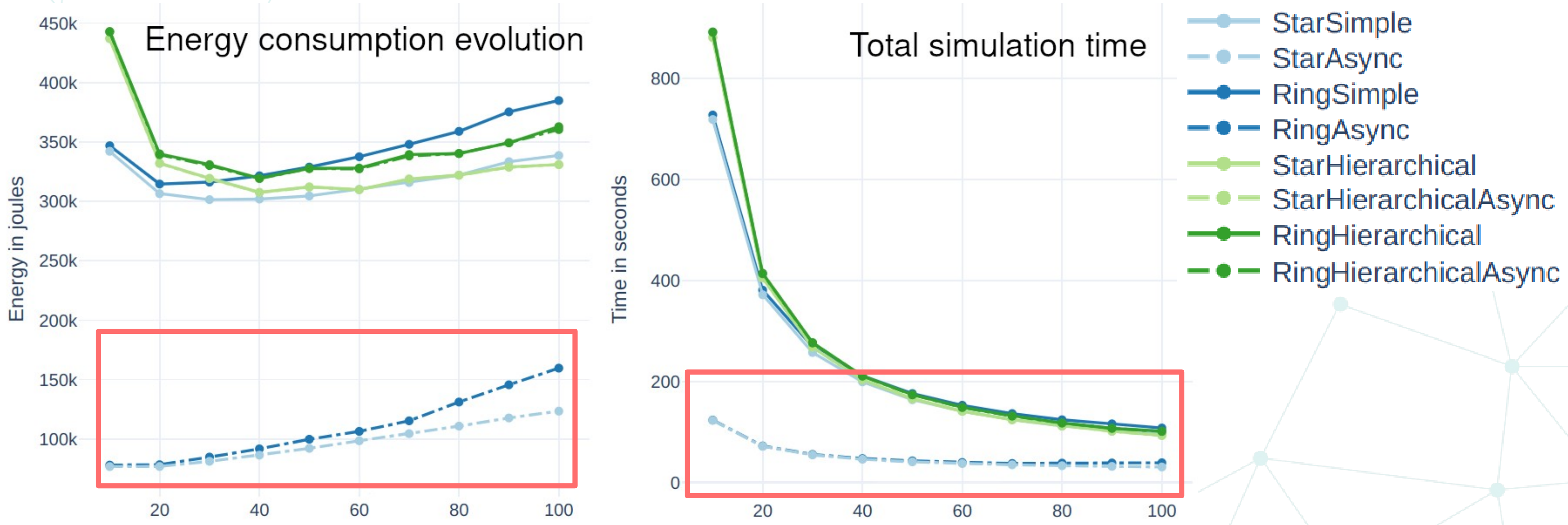
Scenario 1



Reducing runtime **does not always imply in reducing the energy consumption**

Increased workload

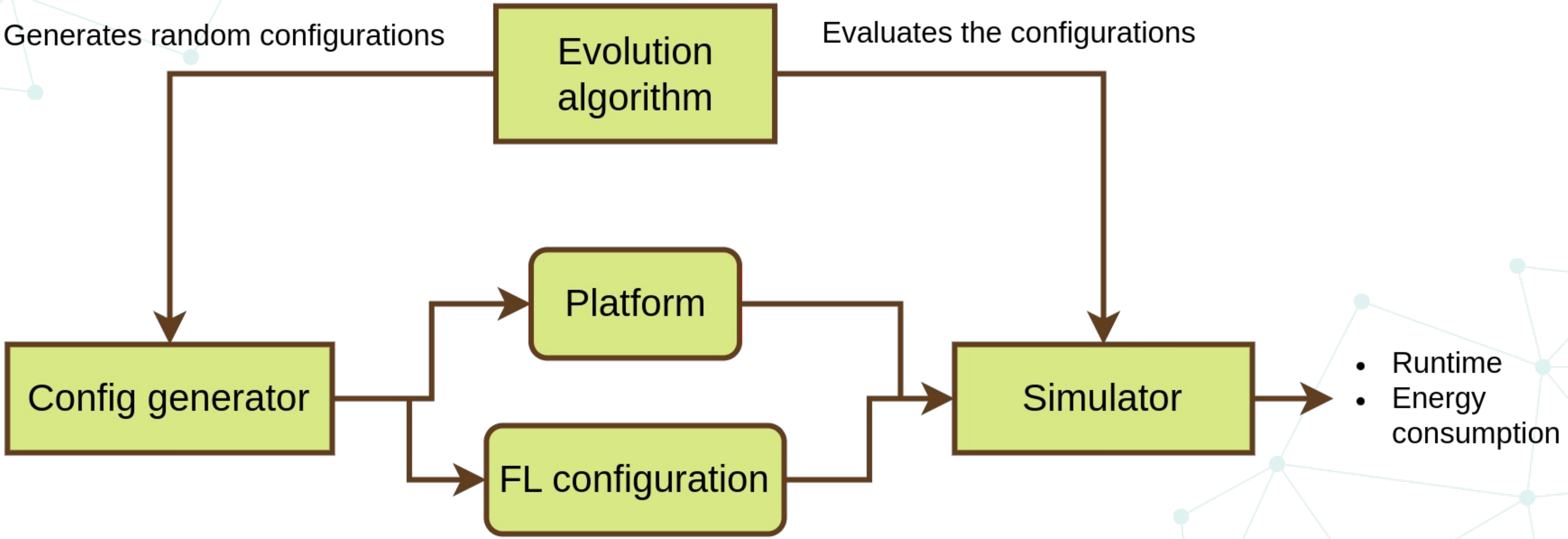
Scenario 1



As expected, **asynchronous algorithms** performs way better thanks to the **reduced idle time** of trainers.

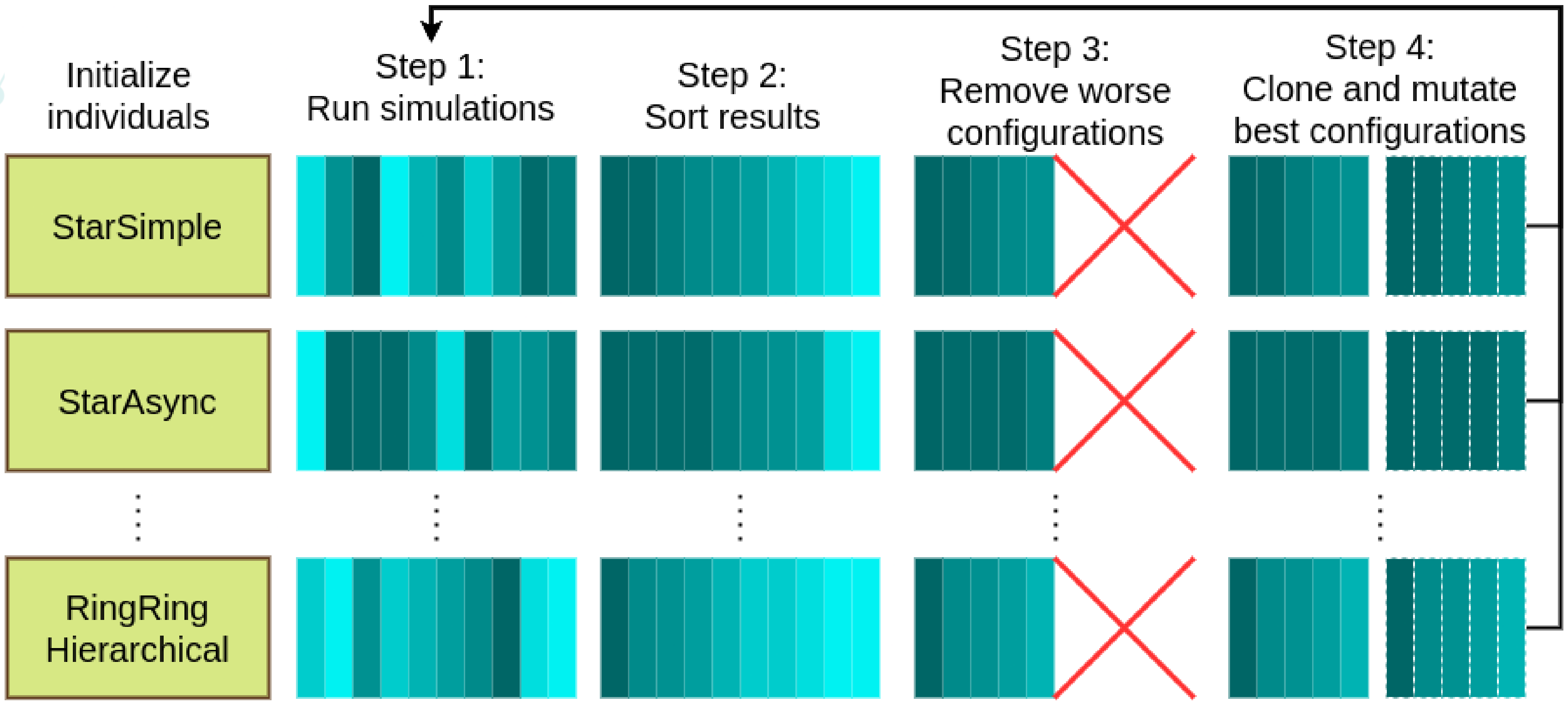
Finding the best config

Scenario 2



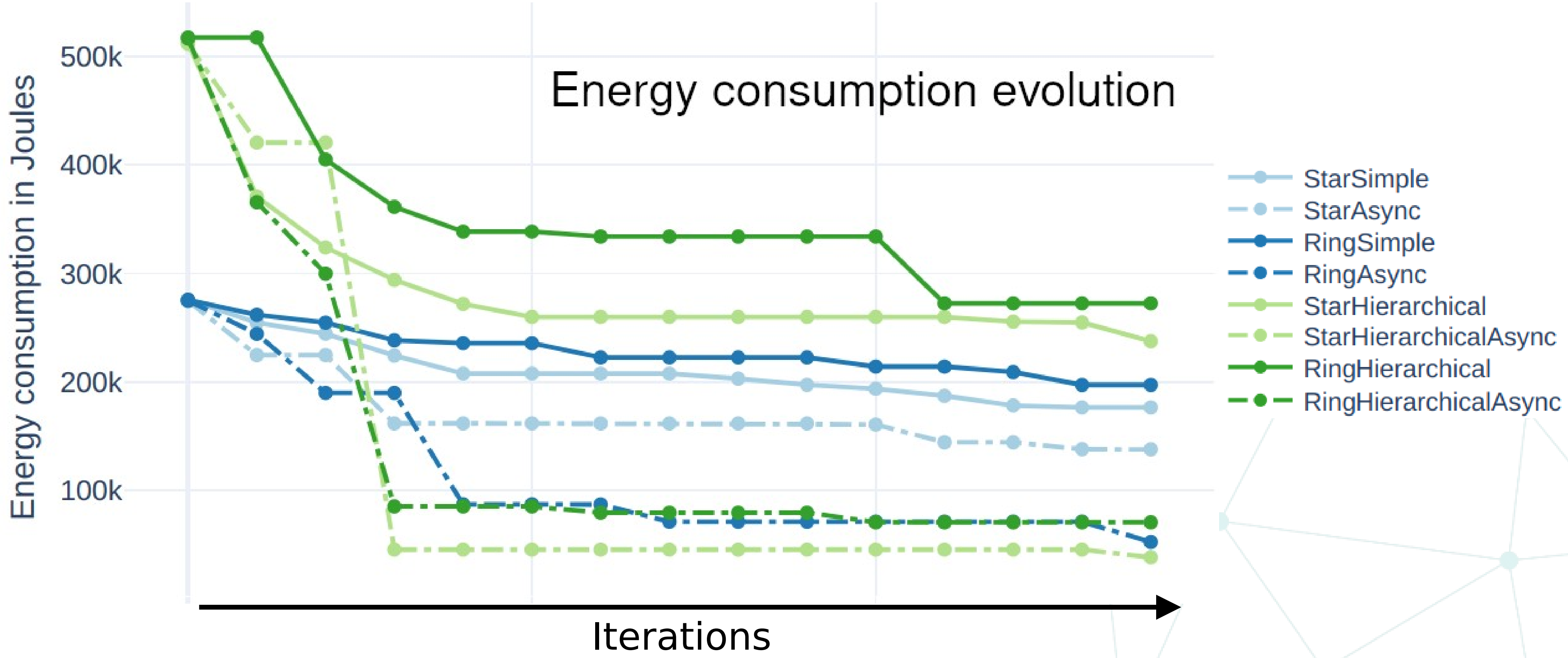
Evolution algorithm

Scenario 2



Evolution results

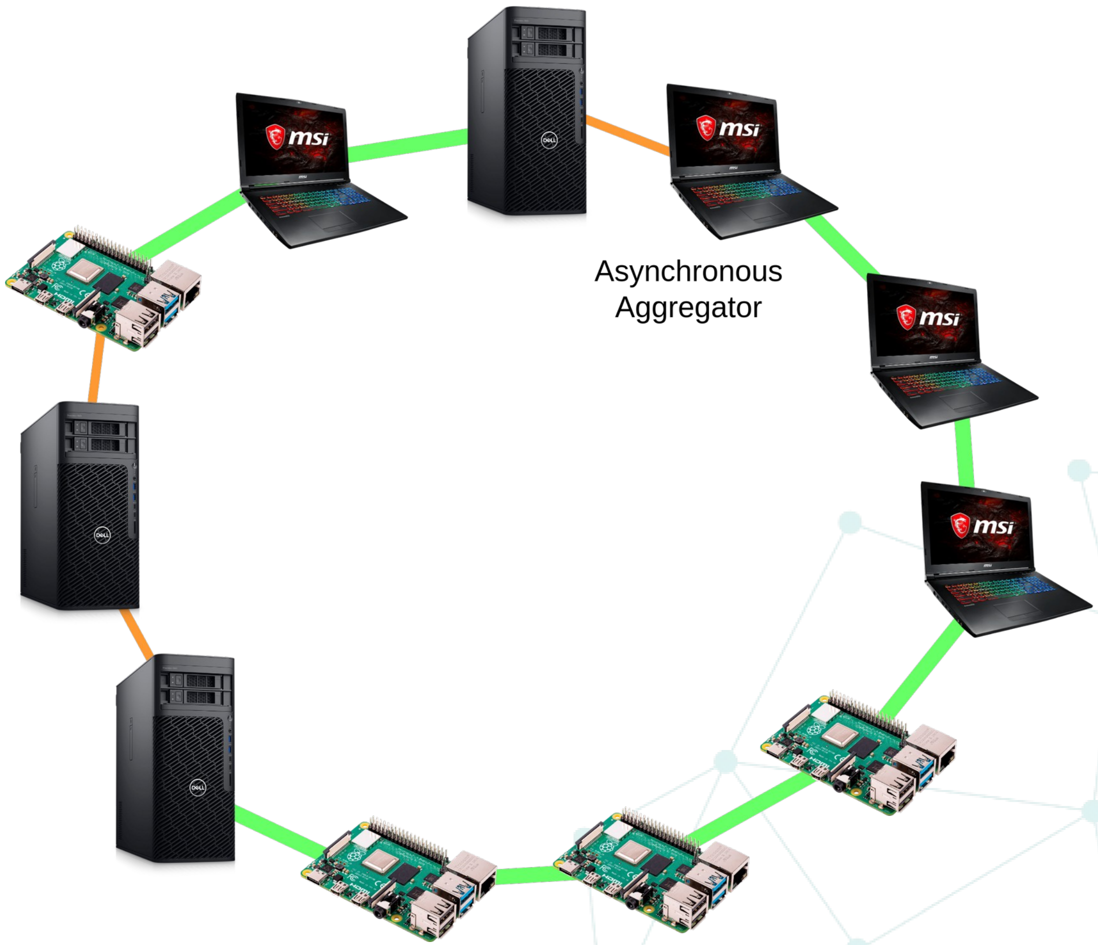
Scenario 2



Best configuration

Scenario 2

- Only 11 machines
- Prefer fast internet links
- Asynchronicity helps!



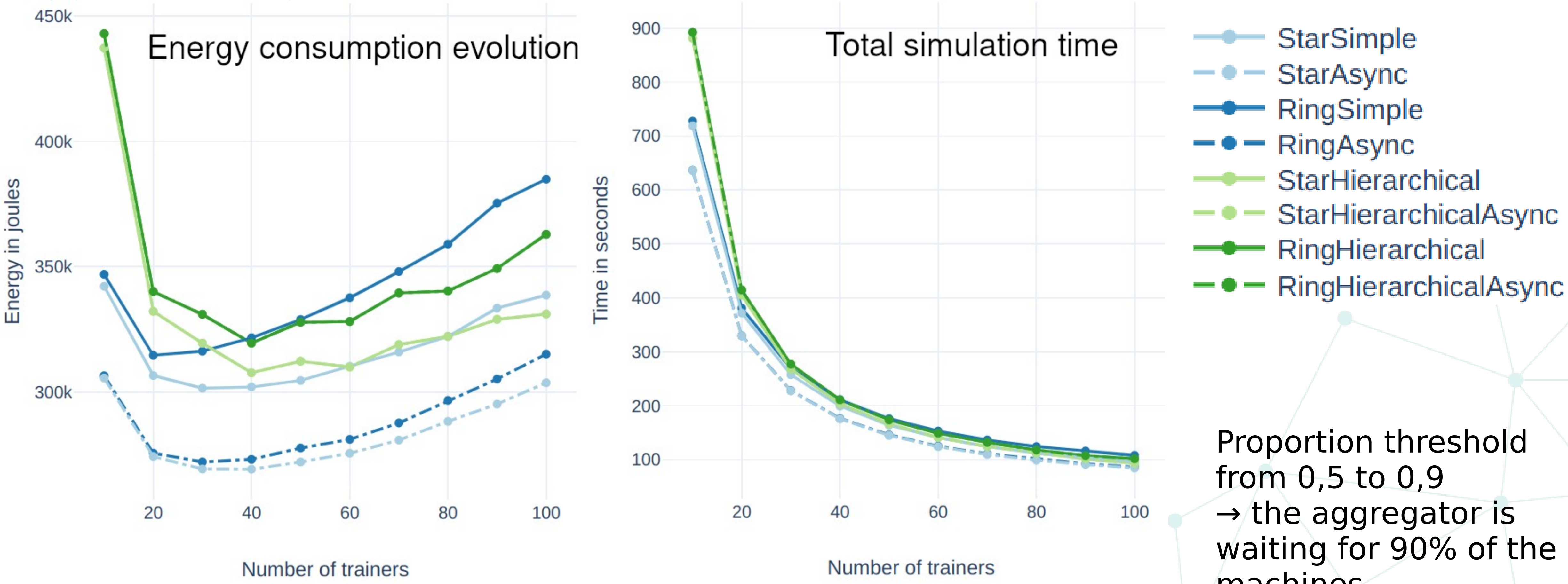
Conclusion

- **Simulation** opens up opportunities for researchers to **develop** and **experiment** new algorithms **quickly**
- *FaLaFEℓS* may help in choosing the **right amount of machines** for a given machine learning task
- Evolutionary algorithms could be used in order to find an **energy efficient** platform configuration

Future work

- Verifying results in real scenario
- Studying the impact of faults on energy consumption

Synchronicity increased



Proportion threshold from 0,5 to 0,9
→ the aggregator is waiting for 90% of the machines

Simulation parameters

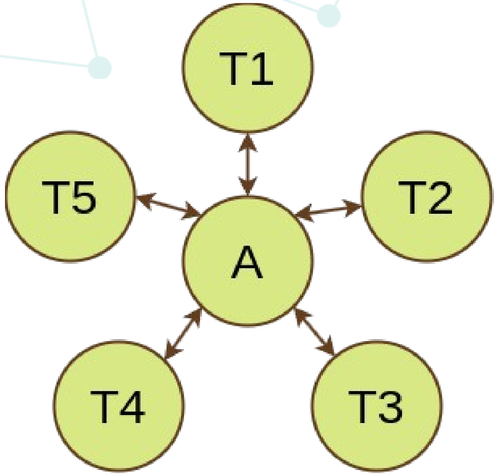
Machine name	FLOPS	#cores	Wattage per state
Precision 7865	144 Gf	16	100 : 120 : 200
MSI GP72 7RD	89.6 Gf	8	20 : 21.6 : 59.33
Raspberry Pi 4B	14.4 Gf	4	2.89 : 3 : 7.28

Link profile	Bandwidth	Latency	Wattage Range
Fiber	1.6 GBps	20 us	25 : 40
ADSL	10 MBps	190 us	15 : 30
Slow-connection	0.07 MBps	1070 us	10 : 15

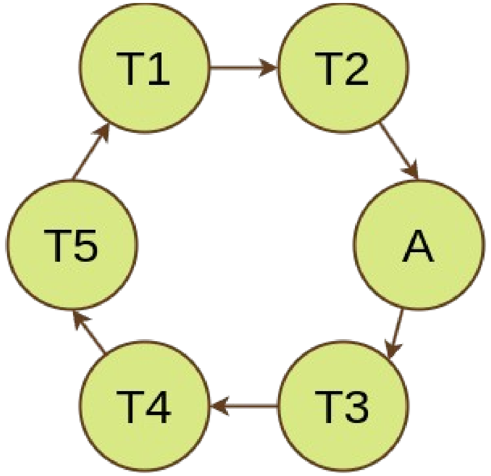
Model specs	Value	Details
#param	199,210	
Param type	FP32	
Model size	796,860 bytes	#param * type size
#samples per node	1000	
#flops for local training	0.39 Gf	#param * 2 * #samples [1]
#flops aggregation	0.0006 Gf	#param * 3

[1] See https://epochai.org/blog/estimating-training-compute

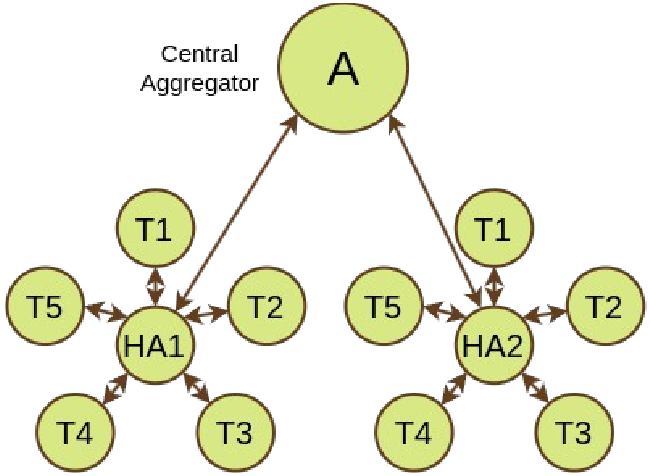
Simulated topologies



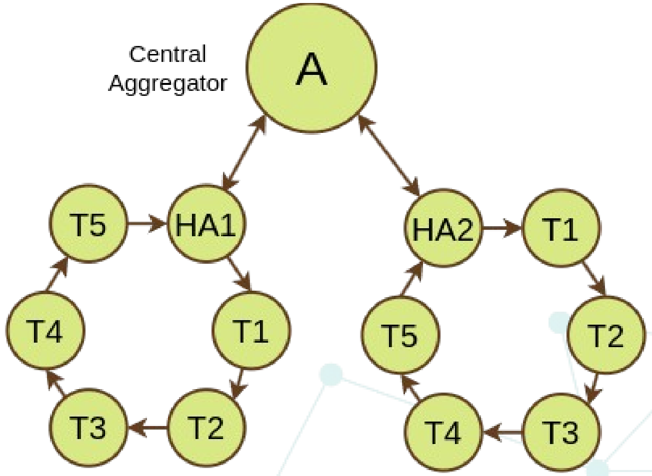
Star topology



Ring topology



Hierarchical + star subclusters



Hierarchical + ring subclusters

Evolution results