MPC for MPC:
Secure Computation on a Massively Parallel Computation Infrastructure

Elaine Shi, Cornell

Joint work with Hubert Chan, Kai-Min Chung, and Wei-Kai Lin

Massively Parallel Computation (MPC)

is a model of computation that captures MapReduce, Hadoop, Spark

Karloff, Suri, and Vassilvitskii (SODA 2010)
and long line of work in the algorithms community in the past decade

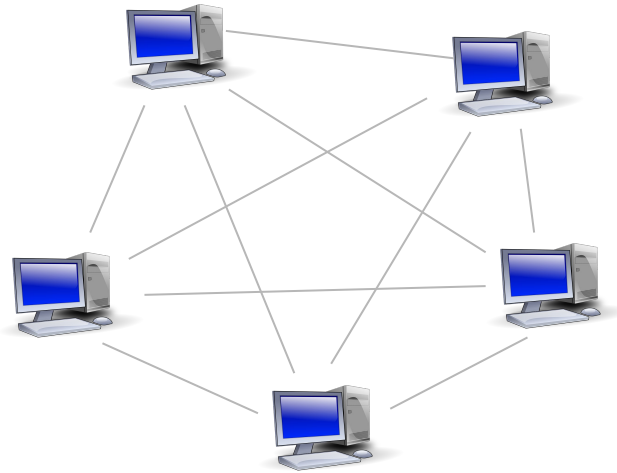
Massively Parallel Computation (MPC)

Each machine has reasonably large space

$s = N^{\varepsilon}$ e.g., 1 TB

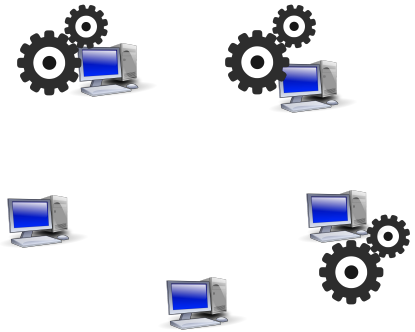
But not large enough to store all data

N e.g., 1 PB

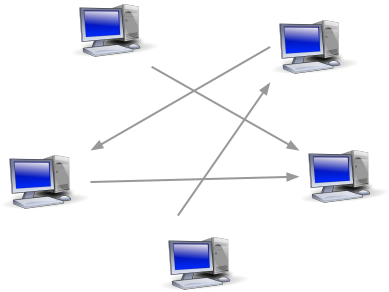


MPC computation proceeds in rounds

Round 1



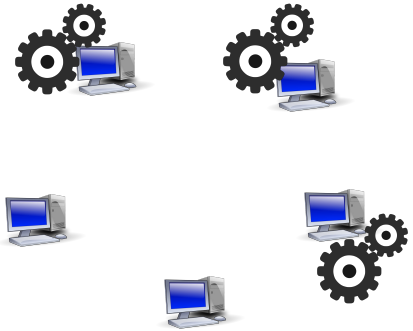
Compute locally

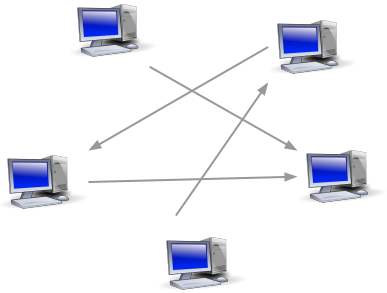


Send messages

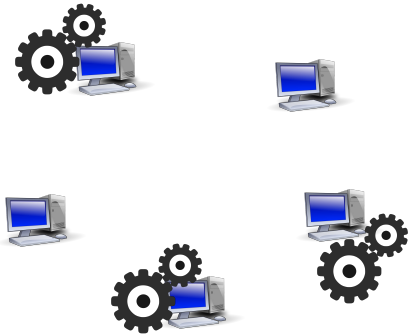
MPC computation proceeds in rounds

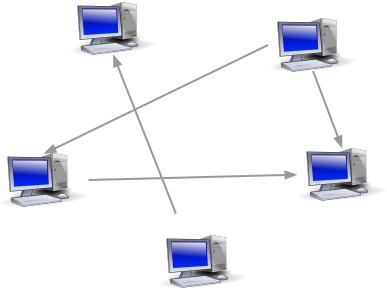
Round 1





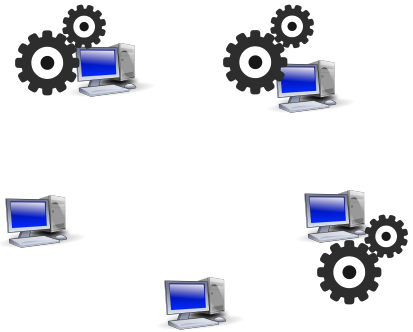
Round 2

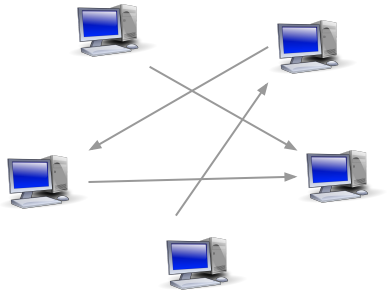


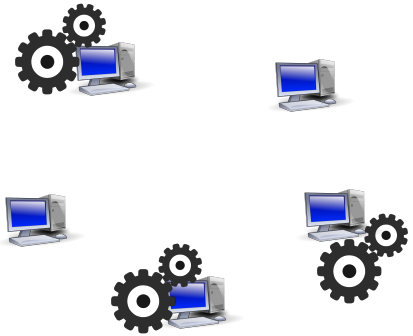


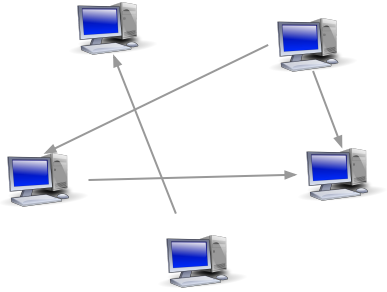
MPC computation proceeds in rounds

Round complexity is the primary metric

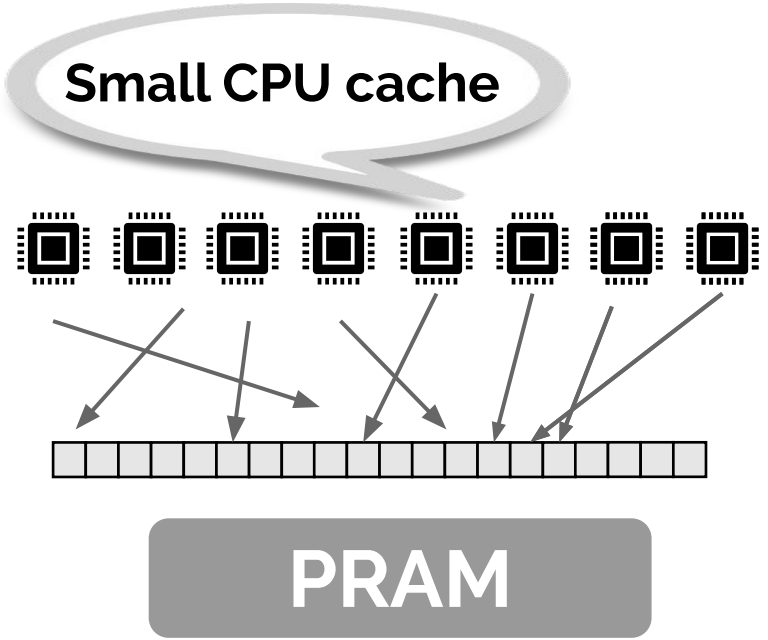




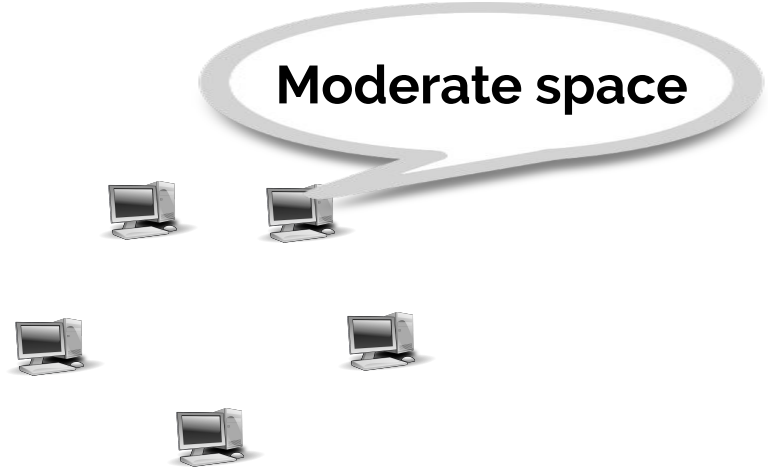




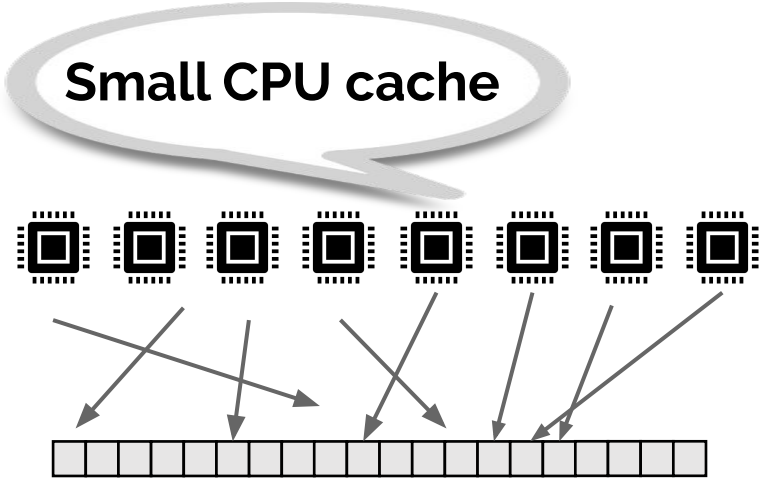
Previously, cryptography for parallel computation focused on PRAMs.



PRAMs **not** a fit for modern parallel architectures.

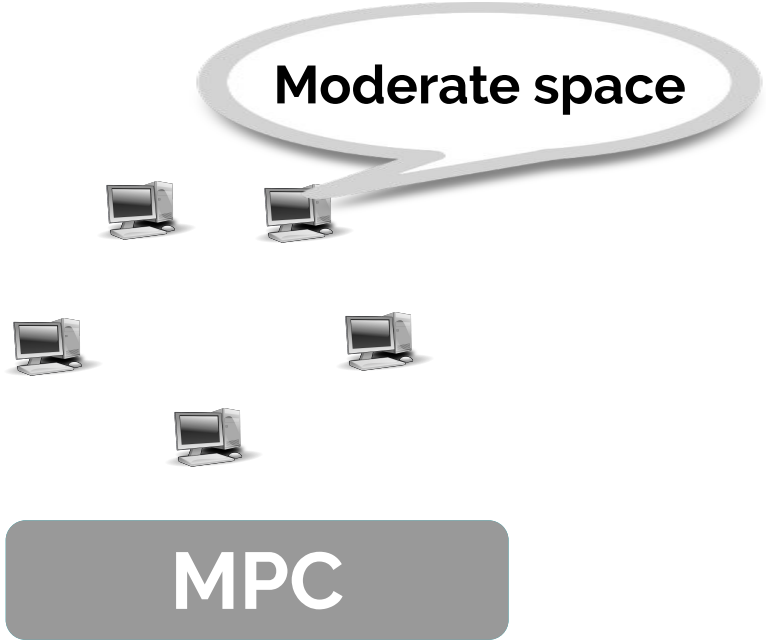


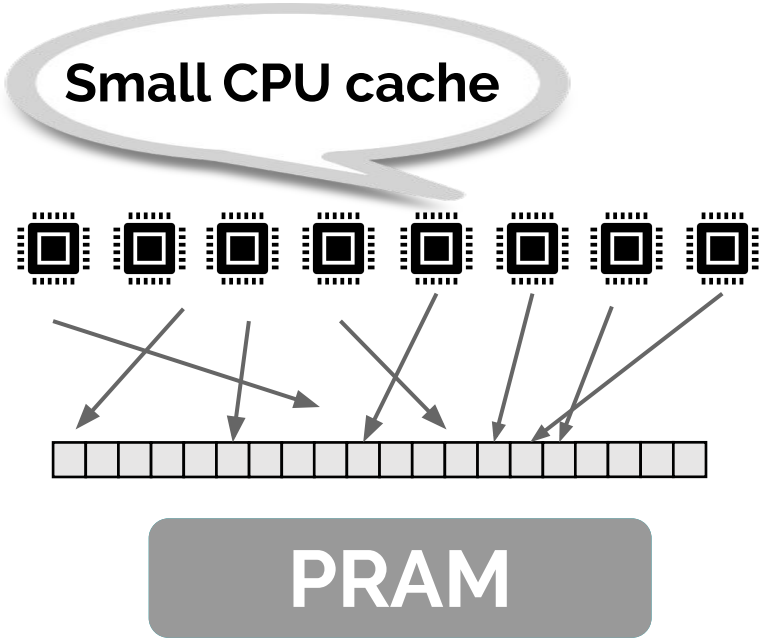
MPC



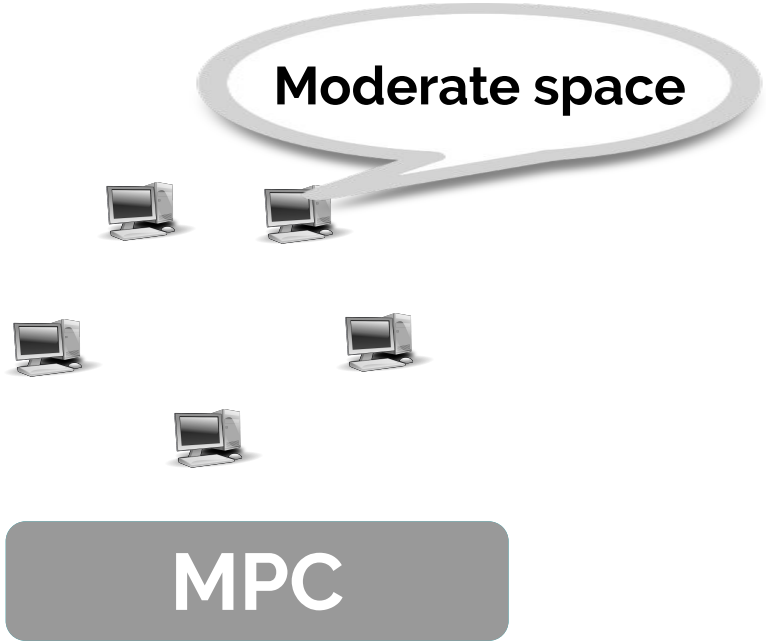
PRAM

Separation: Tasks that take **Ω(log N)** depth on PRAMs can be computed in **o(log N)** rounds on MPC



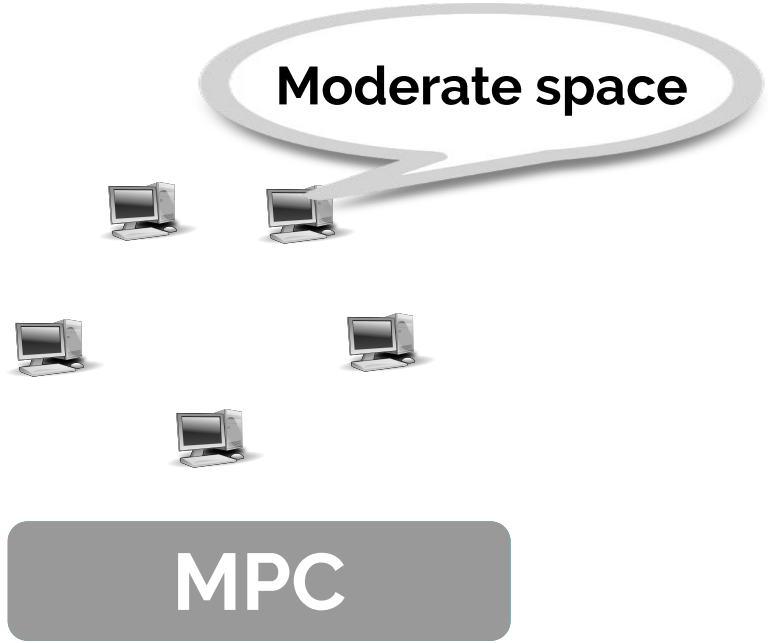


Cryptography in the MPC model?



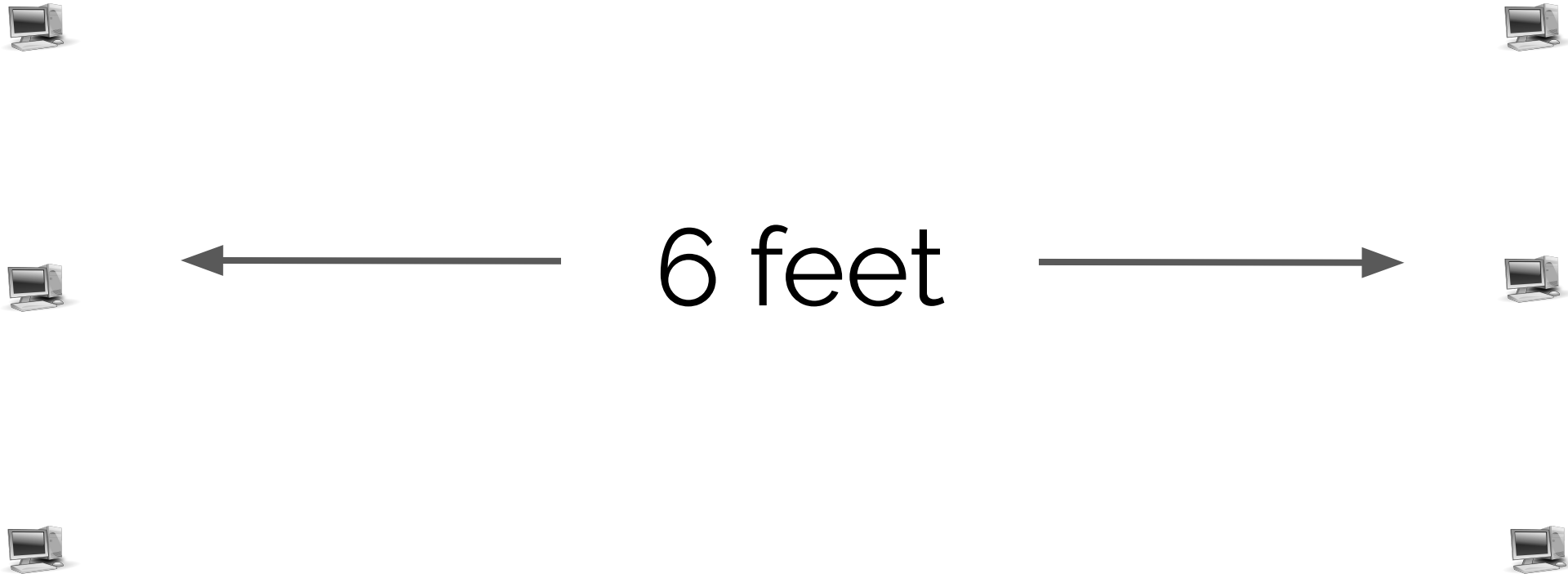
Cryptography in the MPC model?

Yes! We call it MPC for MPC.



Cryptography in the MPC model?

Yes! We call it MPC for MPC.

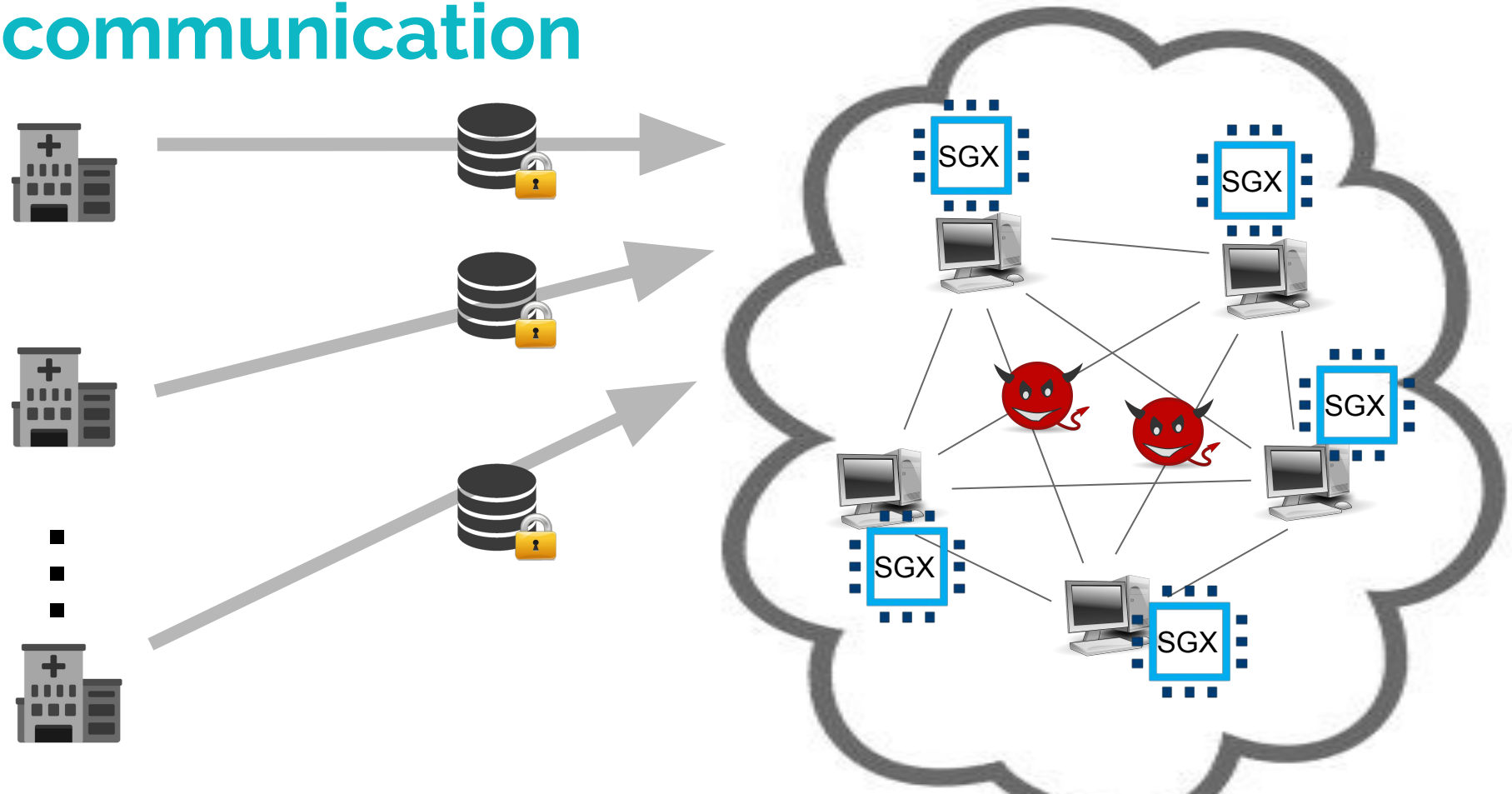


Can MPC algorithms be made secure with small overhead?

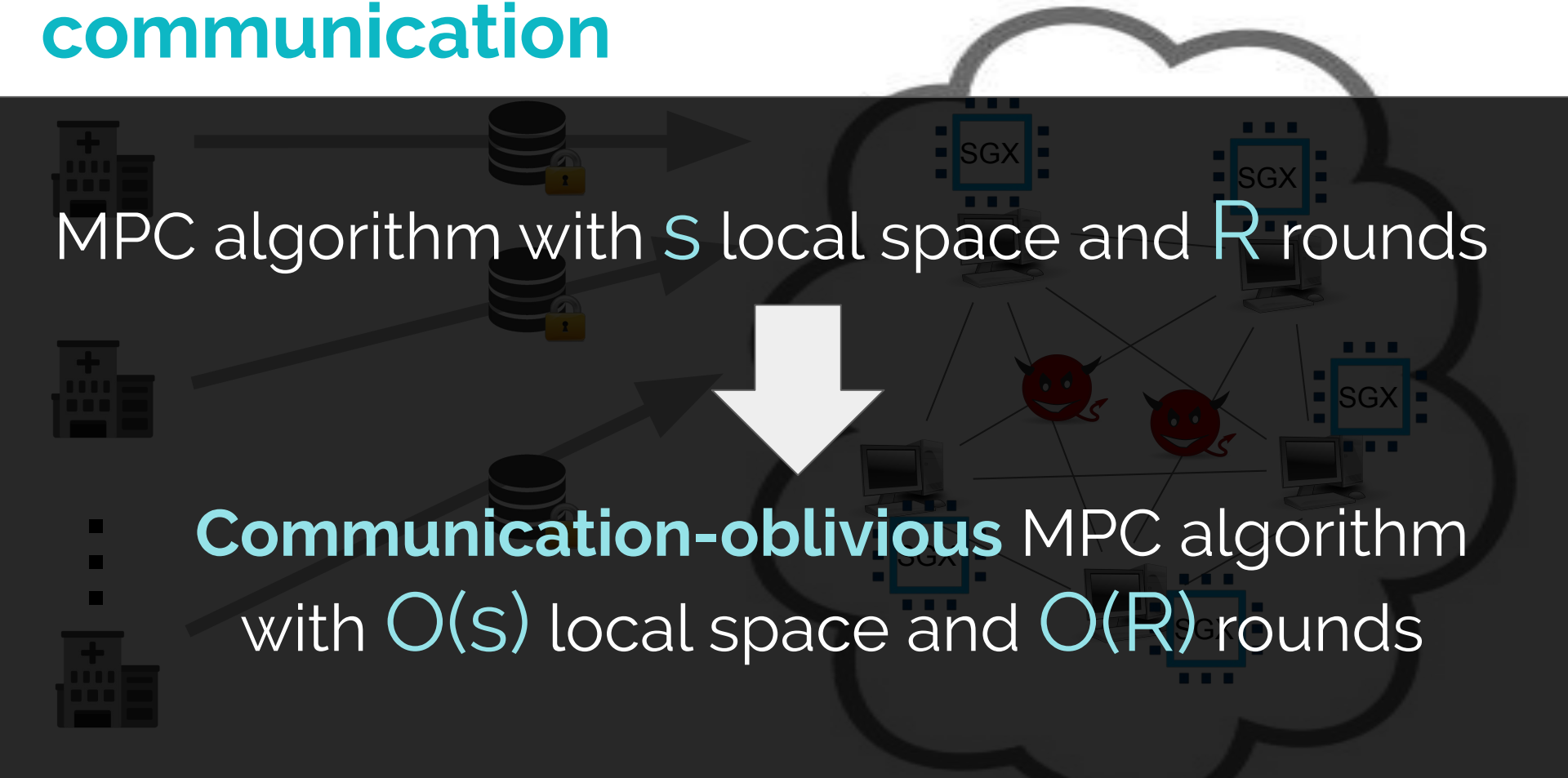





Scenario 1: secure endpoints, unsafe communication



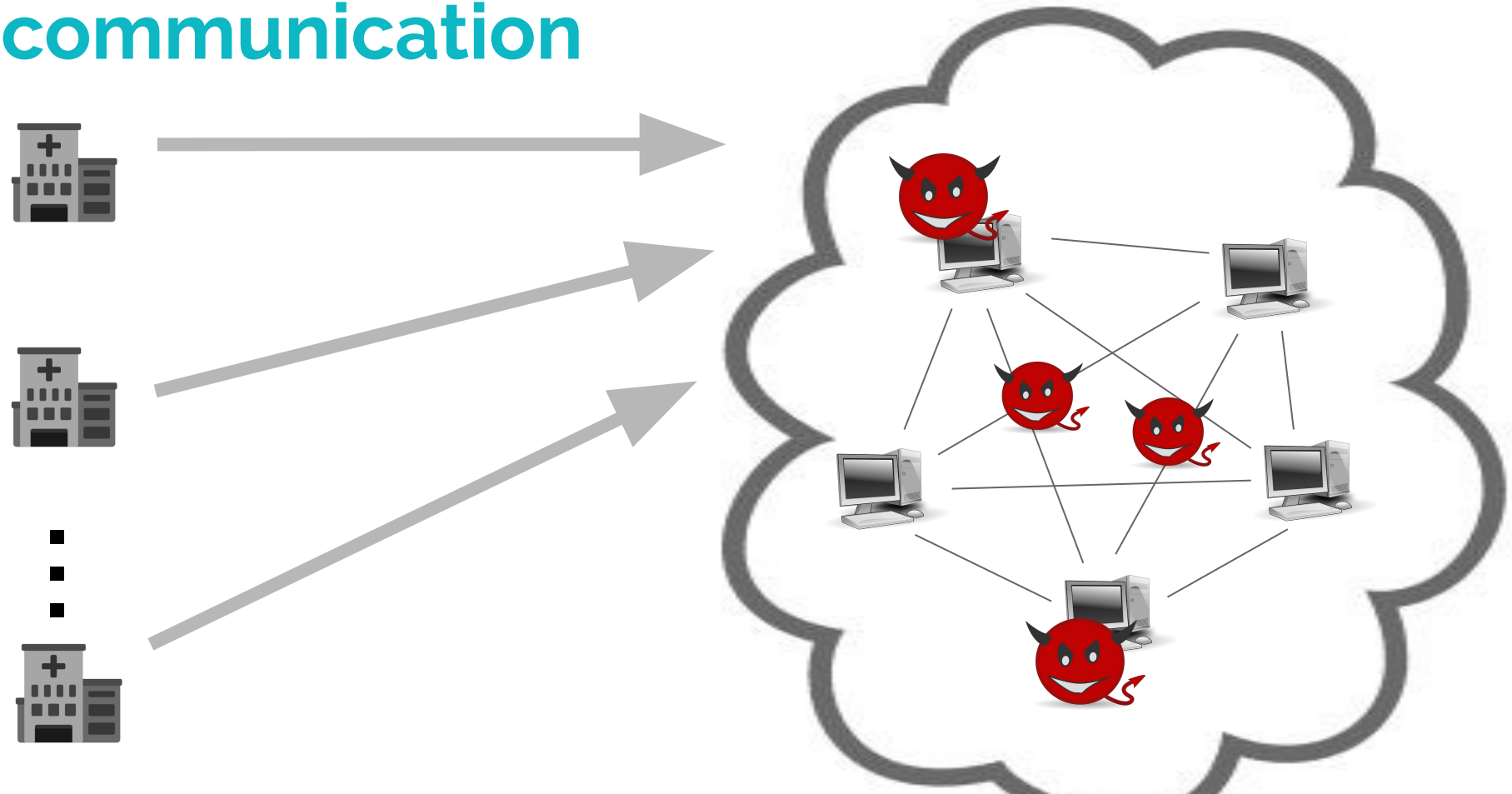
Scenario 1: secure endpoints, unsafe communication



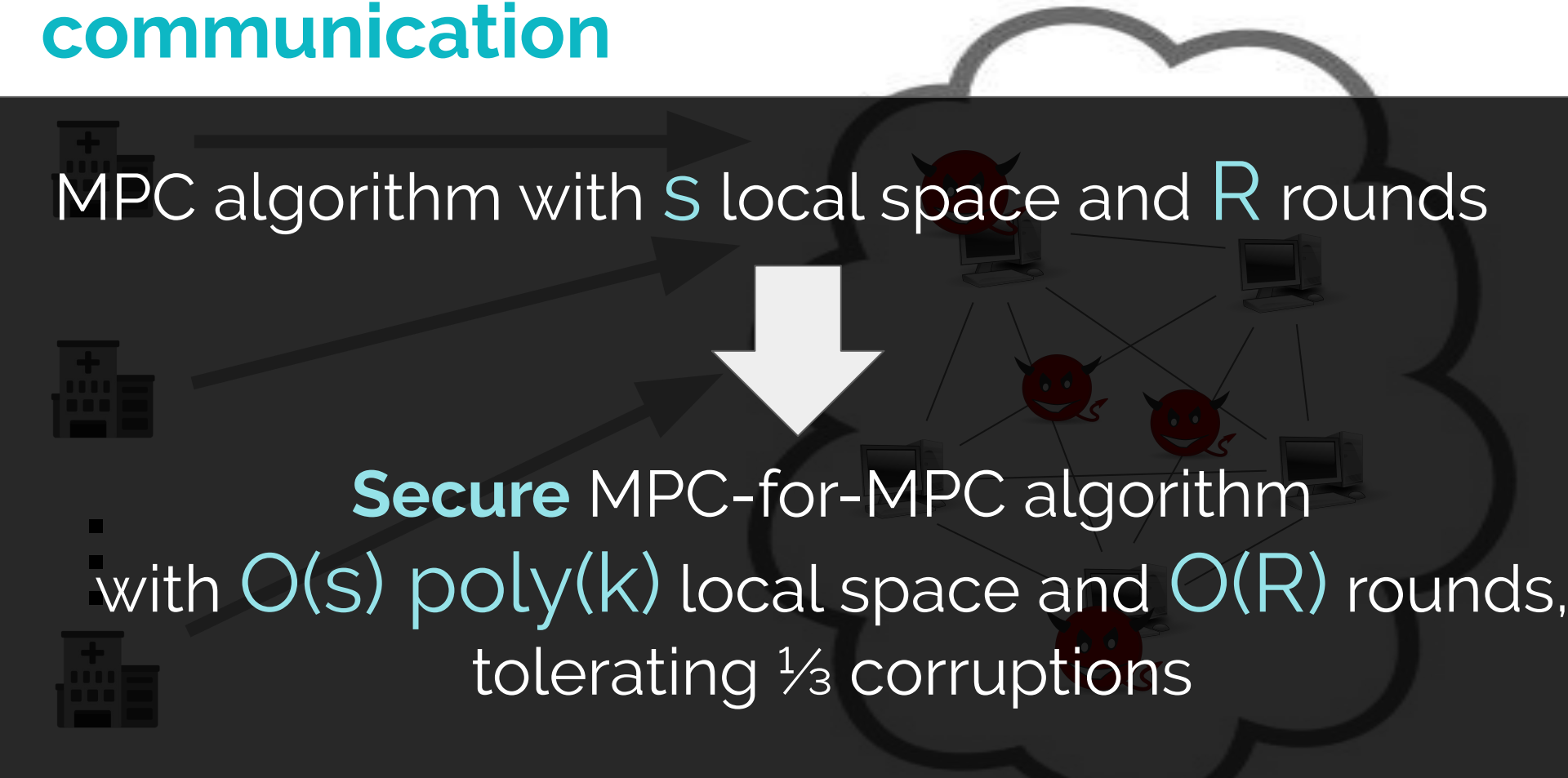
MPC algorithm with s local space and R rounds

Communication-oblivious MPC algorithm with $O(s)$ local space and $O(R)$ rounds

Scenario 2: unsafe endpoints, unsafe communication



Scenario 2: unsafe endpoints, unsafe communication



MPC algorithm with s local space and R rounds

Secure MPC-for-MPC algorithm

with $O(s)$ $\text{poly}(k)$ local space and $O(R)$ rounds,

tolerating ⅓ corruptions

Our work is an exciting beginning...

- We lay the groundwork for securing computation in realistic parallel architectures
- Show promising feasibility results, with evidence of concrete efficiency
- Rich space for future work, a bridge between algorithms and crypto

  e.g., secure large-scale AI

Thank you!

elaine@cs.cornell.edu