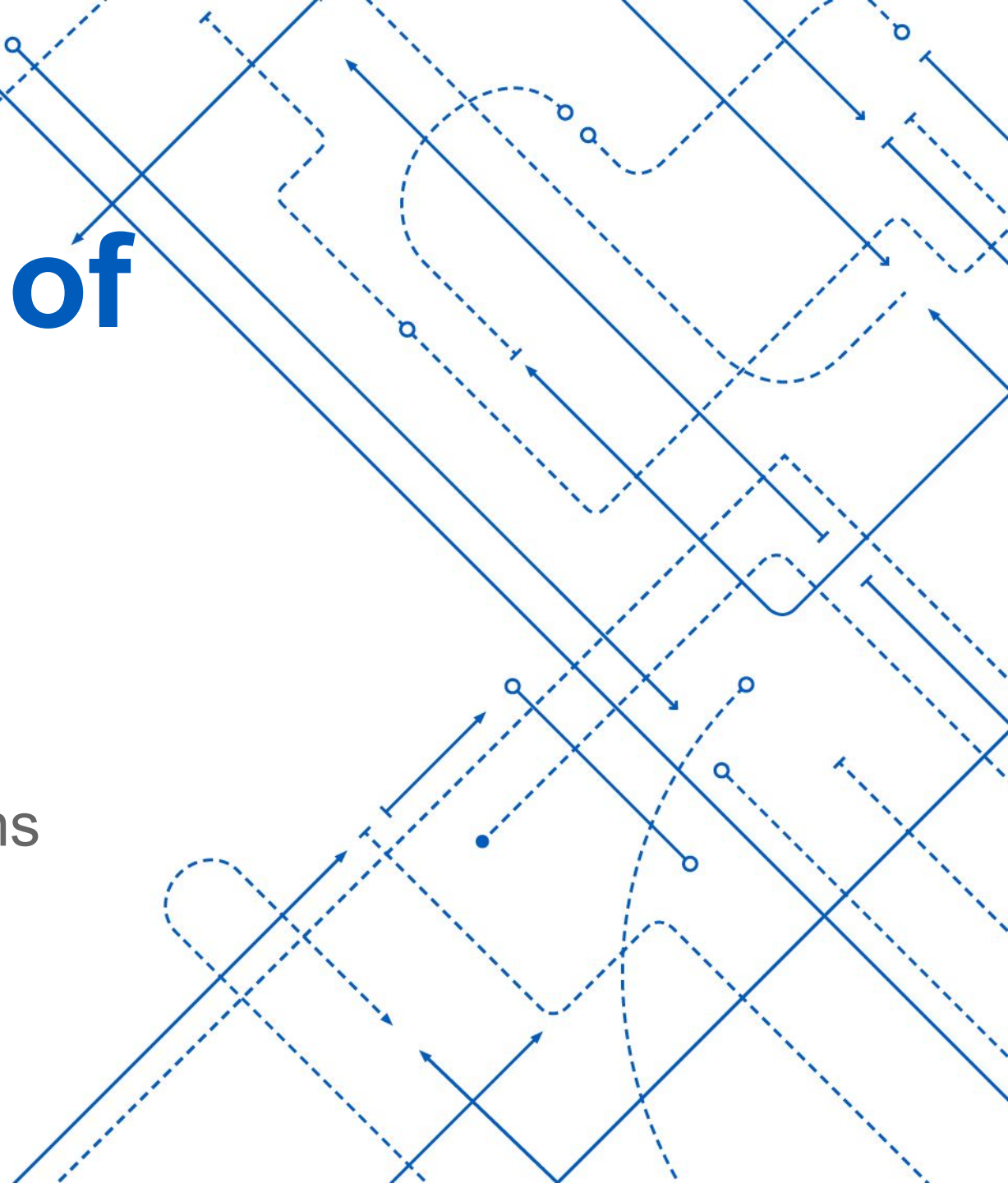
Conway's Game of Life in Parallel

Al-kesna Foster

CSE 708: Massively Parallel Systems

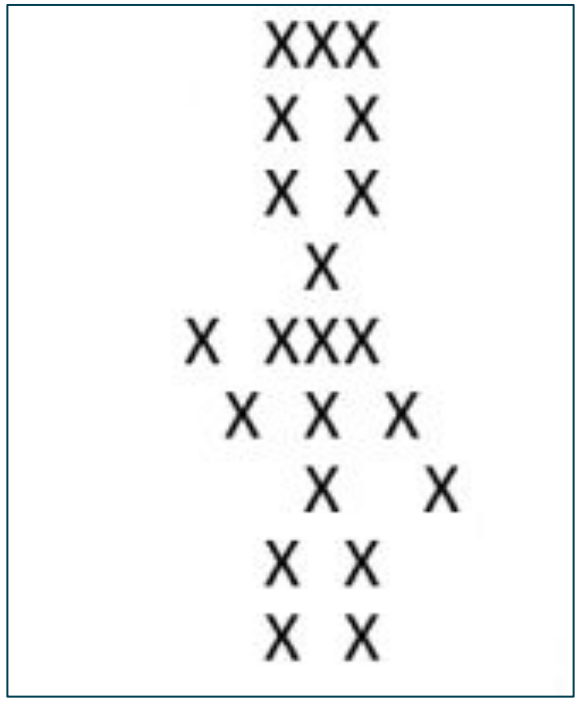
Dr. Russ Miller

University at Buffalo The State University of New York



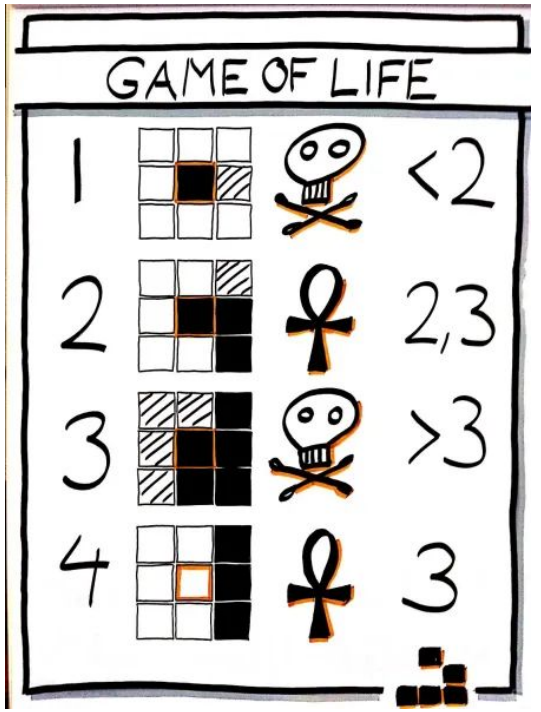
Conway's Game of Life

- Cellular automation created by John Conway
- Takes place on a 2-Dimensional grid
- A cell is either alive or dead
- Life proceeds in generations: G → G + 1
- A set of rules determines if a cell lives or dies



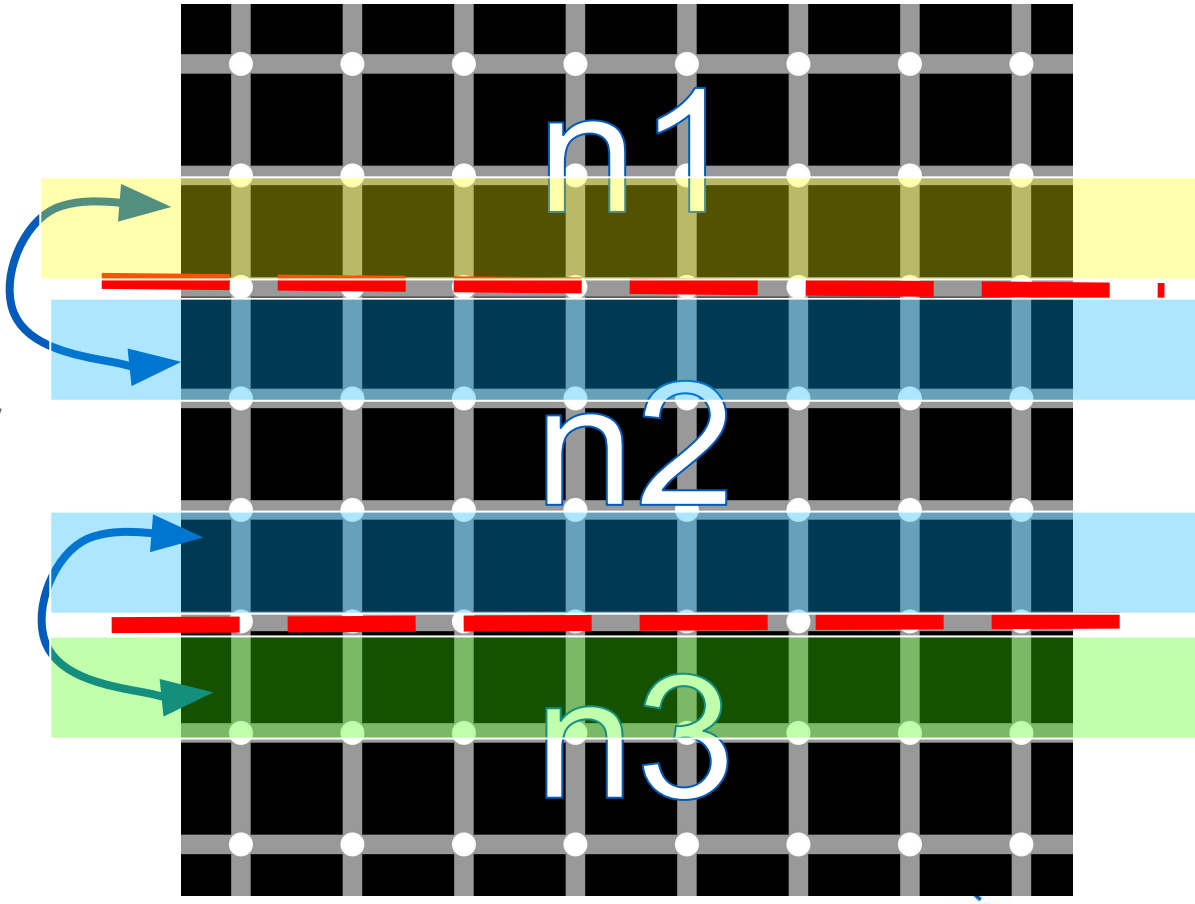
Why do we care?

- Simplicity can results in great complexity
- Turing complete!
  - You can program the Conway's Game of Life in Conway's Game of Life



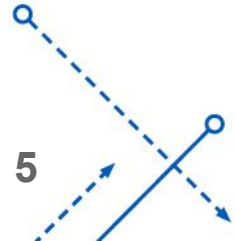
How do we parallelize?

- Split the grid among participating nodes
- Each node needs to communicate their borders to neighboring nodes
- Each node computes cell states with new neighbor information



The Algorithm

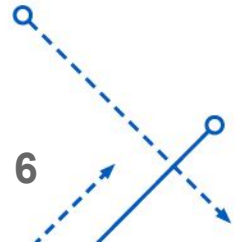
1. Each node allocates enough memory for their portion of the grid + neighboring rows(ghost rows)
2. Every nodes calls a nonblocking MPI Send and Receive in parallel (MPI_Isend and MPI_Irecv)
3. Followed by a MPI_Wait for synchronization
4. Repeat for however many generations



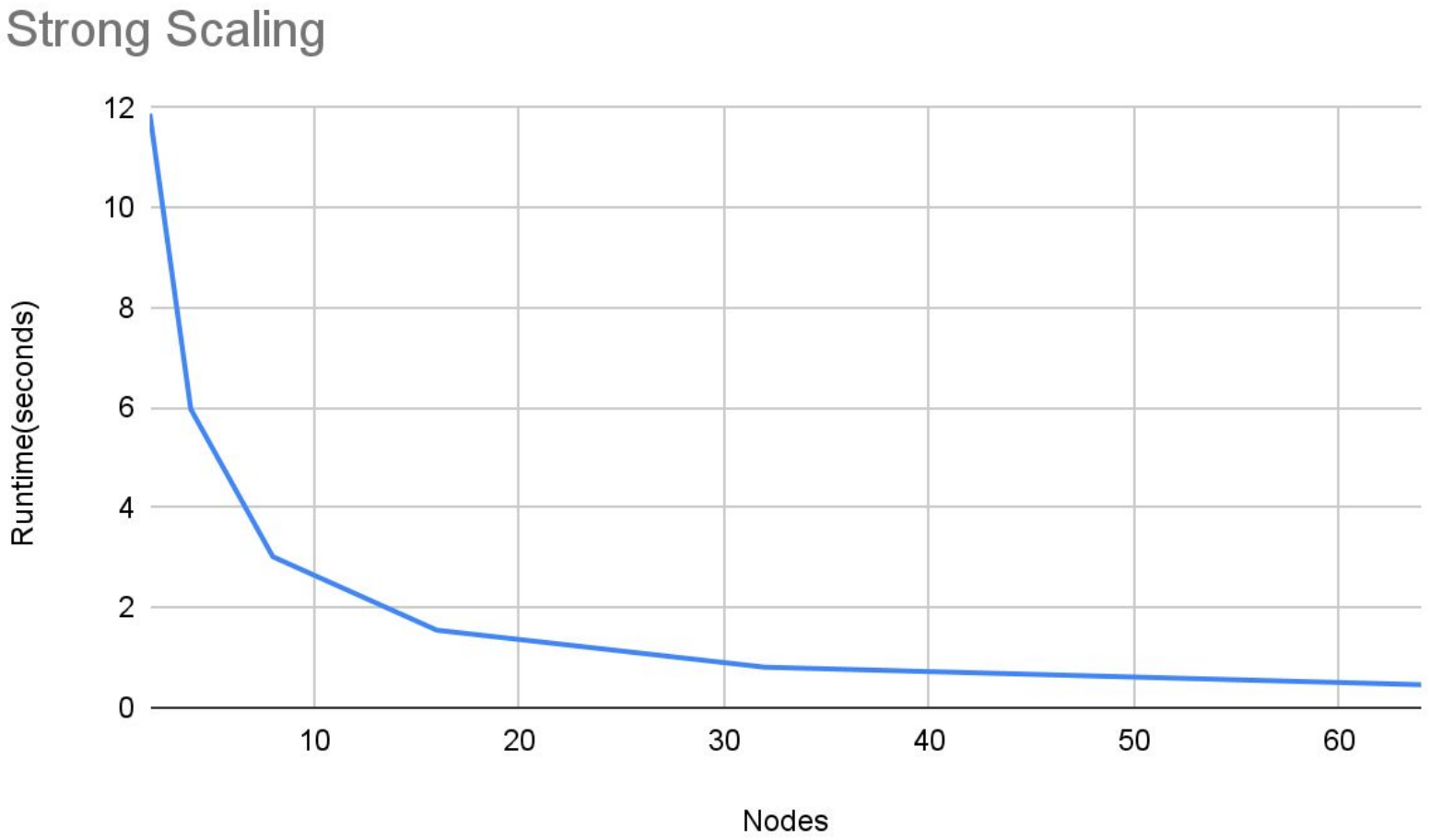
Strong Scaling

Nodes	Runtime
2	11.88s
4	5.97s
8	3.02s
16	1.55s
32	0.81s
64	0.46s

Problem Size:
- 832 rows
- 1000 cols



Strong Scaling

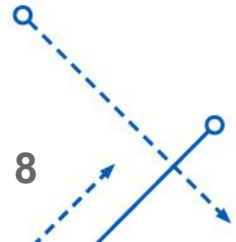


Weak Scaling

Nodes	Runtime
2	23.70s
4	23.72s
8	23.74s
16	23.75s
32	23.76s
64	23.86s

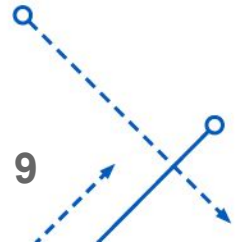
Problem Size:

- starting from 1664 rows, double # of rows as we double the # of nodes

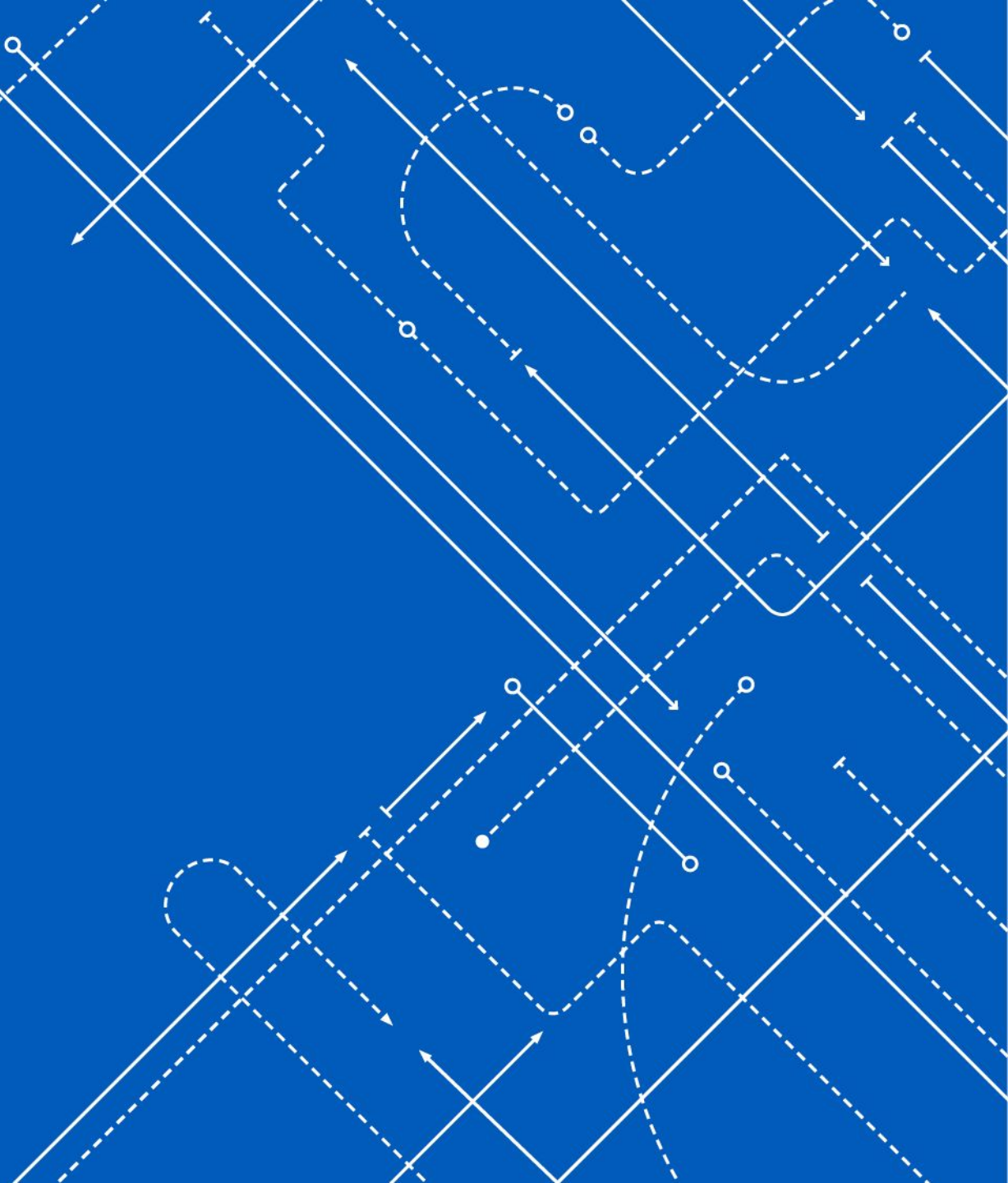


References

- https://qualityswdev.com/2011/07/31/conways-game-of-life-in-scala/



Questions?



University at Buffalo The State University of New York