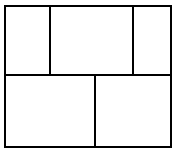
Robert Metzger 347332
Igor Viskovic 334365
Robert Waury 351431
Tobias Wölk 196364

Exercise 2.2 a

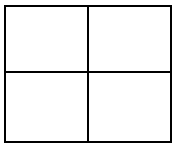
Distributed Conway's Game of Life

The matrix storing the state of the game is distributed on n processes.

For the sake of simplicity we assume no overlapping borders.



Overlapping



Non-overlapping

This way we just have to distinguish three cases assuming in each dimension there are at least two processes i.e. n is not prime.

- the corner case with one neighbor on the horizontal, one on the perpendicular and one on the diagonal
- the border case with either two neighbors on the horizontal and one on the perpendicular or vice versa and one neighbor on each diagonal
- the inner case with two horizontal, two perpendicular and four diagonal neighbors.

Every process needs to send as many messages as it receives.

Messages from a diagonal contain only information about a single field of the matrix whereas messages from or to a perpendicular or horizontal neighbor contain the content of the column or line excerpt.

Since a process needs to address its direct neighbors it has to be aware about the size of the matrix and the position of itself within the matrix.

Each message must contain the id of the sender process, the number of 1's and the position of the 1's in sparse notation. This way if no 1's are contained in the bordering excerpt a message with zero 1's is send. The 1's are send because it is assumed that in Conway's Game of Life the matrix will be very sparse and will therefore justify the overhead of sparse notation by smaller message sizes.

This is only true for message passing environments in shared memory systems storing the whole matrix in-memory and passing pointers to the different processes would be more efficient.

Robert Metzger 347332
Igor Viskovic 334365
Robert Waury 351431
Tobias Wölk 196364

Exercise 2.2 a

An iteration of the algorithm will work as follows:

1. Each process sends its outer excerpts to its 3, 5 or 8 neighbors.
2. Each process receives the bordering excerpts from its neighbors.
3. As soon as all messages are received a local iteration of Conway's Game of Life is started.
4. The process sends a FINISHED message to the dedicated coordinating process.
    a. As soon as a process send its FINISHED message it is receiving from its neighbors
    b. Therefore Phase 1 and 2 should be non-blocking since in a distributed environment The exact order of operations cannot be predicted
5. As soon as the coordinating process has received n FINISHED messages a RESUME message is broadcasted to all n processes.
6. If a process receives a RESUME goto Phase 1.

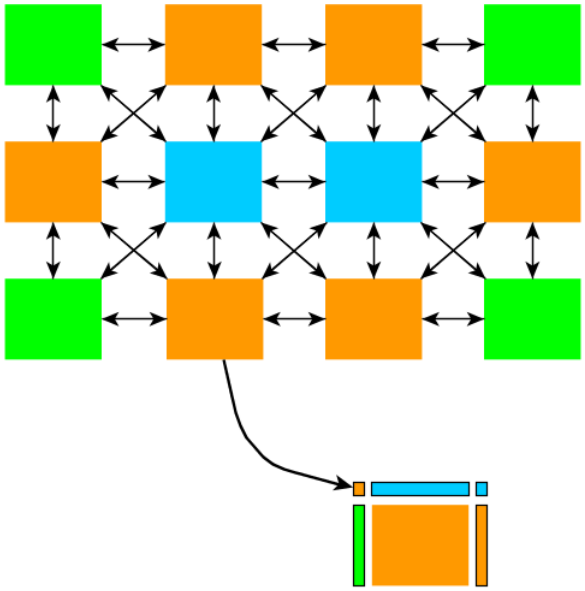


Figure 1 Phase 1 and 2 from a global perspective and Phase 3 from a local perspective

Robert Metzger 347332
Igor Viskovic 334365
Robert Waury 351431
Tobias Wölk 196364

Exercise 2.2 a

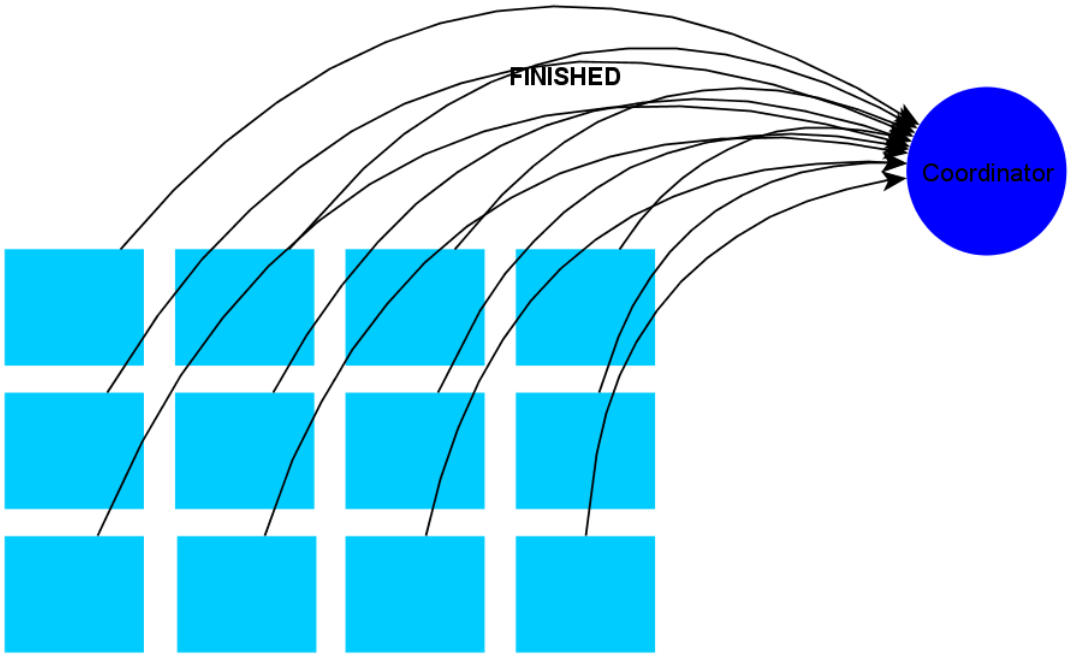


Figure 2 Phase 4

Robert Metzger 347332
Igor Viskovic 334365
Robert Waury 351431
Tobias Wölk 196364

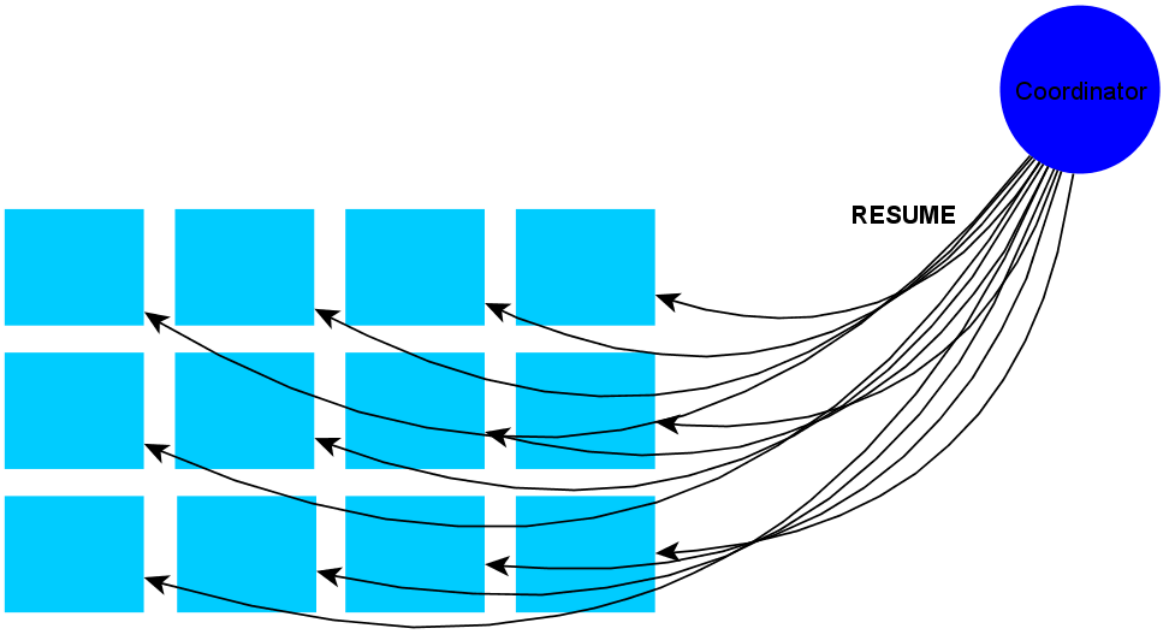


Figure 3 Phase 5