

Data Distribution

So far, we've allocated data evenly among processes

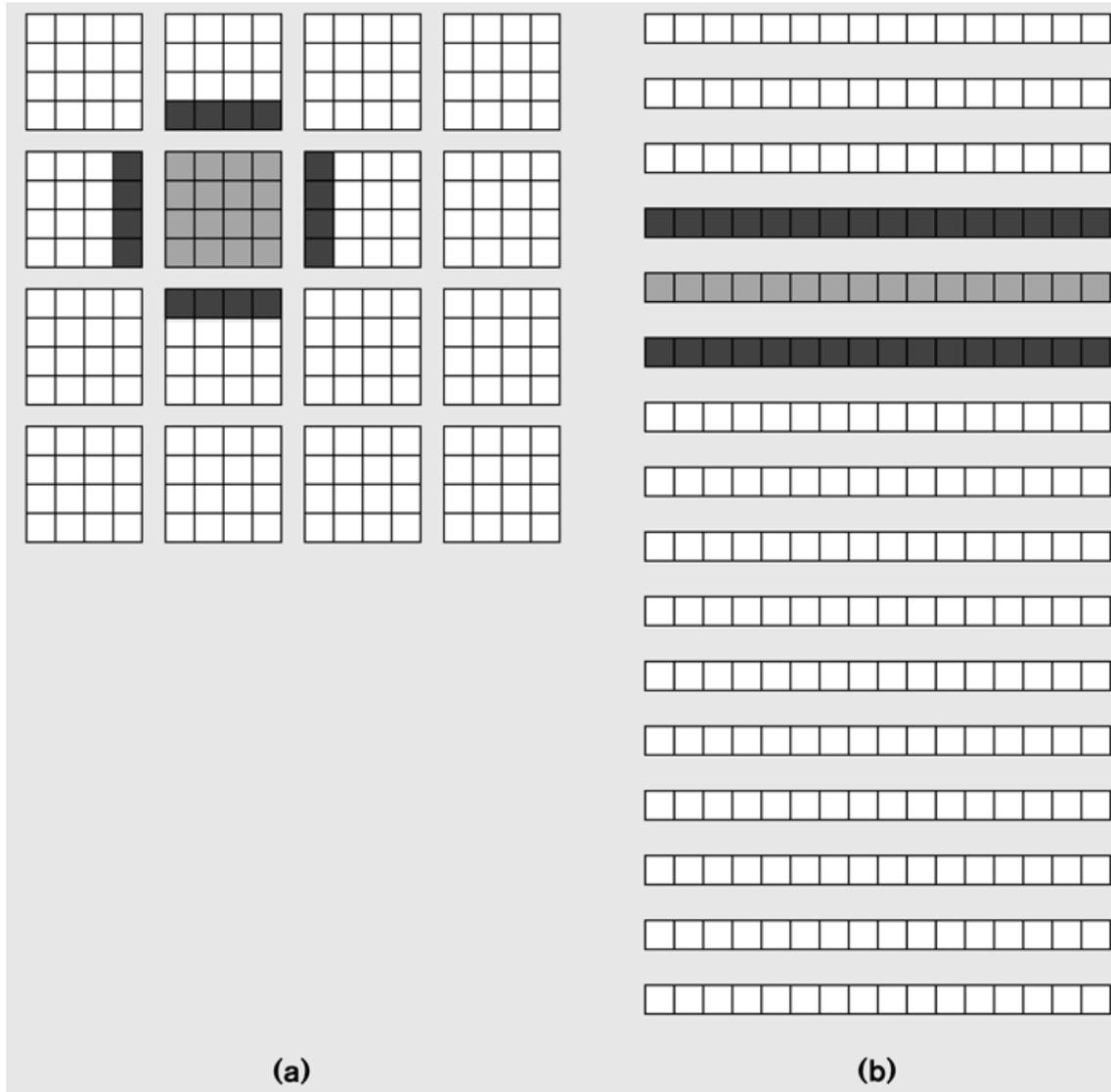
- Some distributions might lead to less sharing

Can't happen for a true reduce or scan

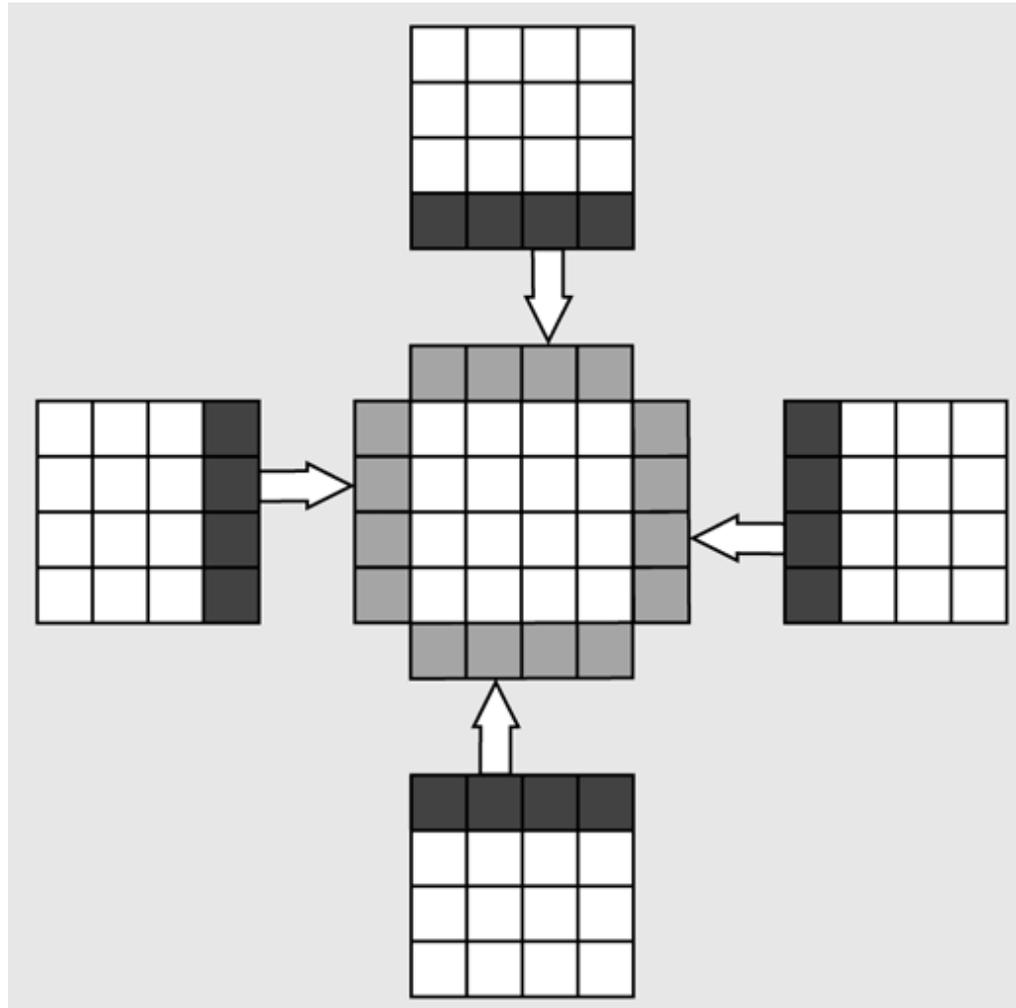
- Some data might be more expensive to process

Rarely happens with reduces and scans

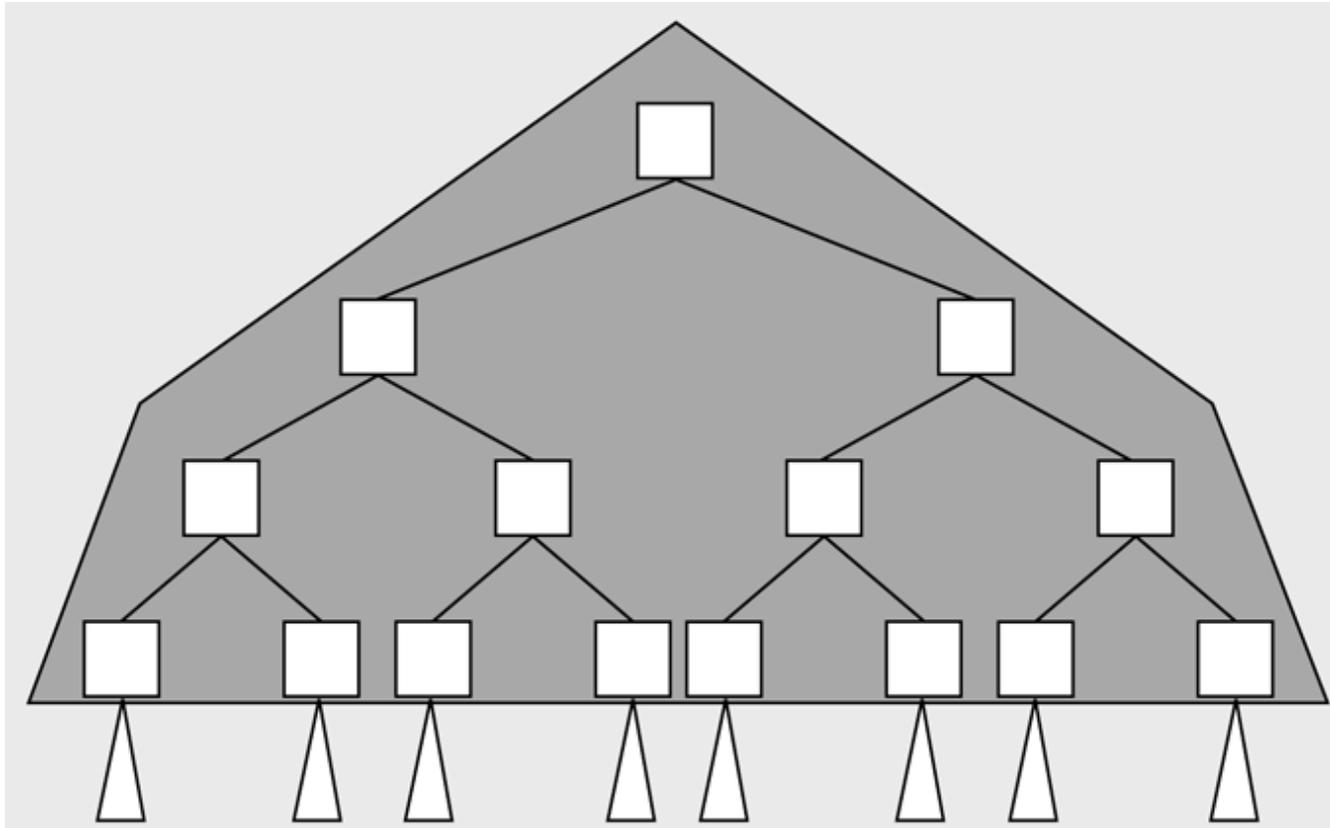
Data Sharing



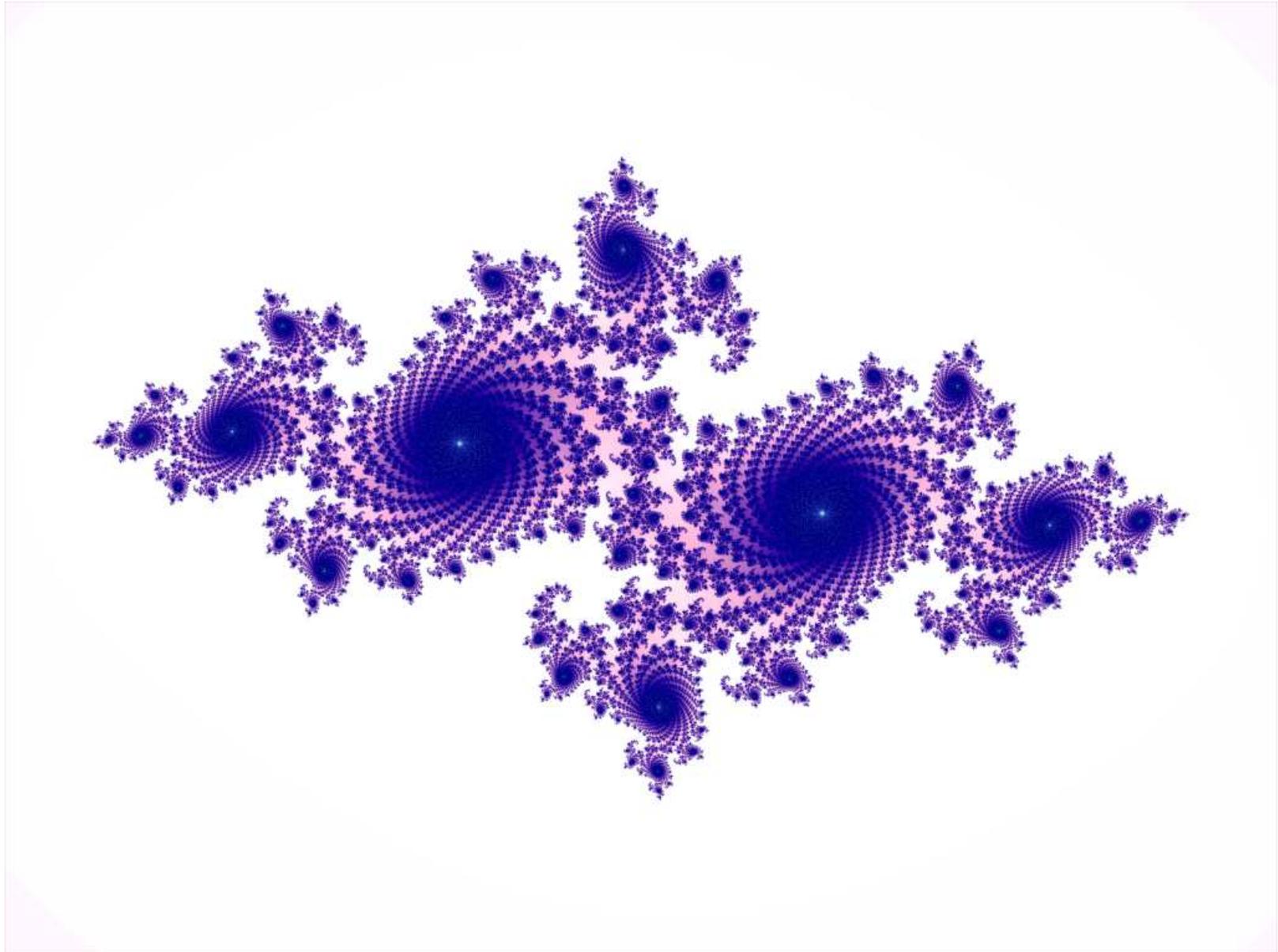
Overlap Regions to Reduce Sharing



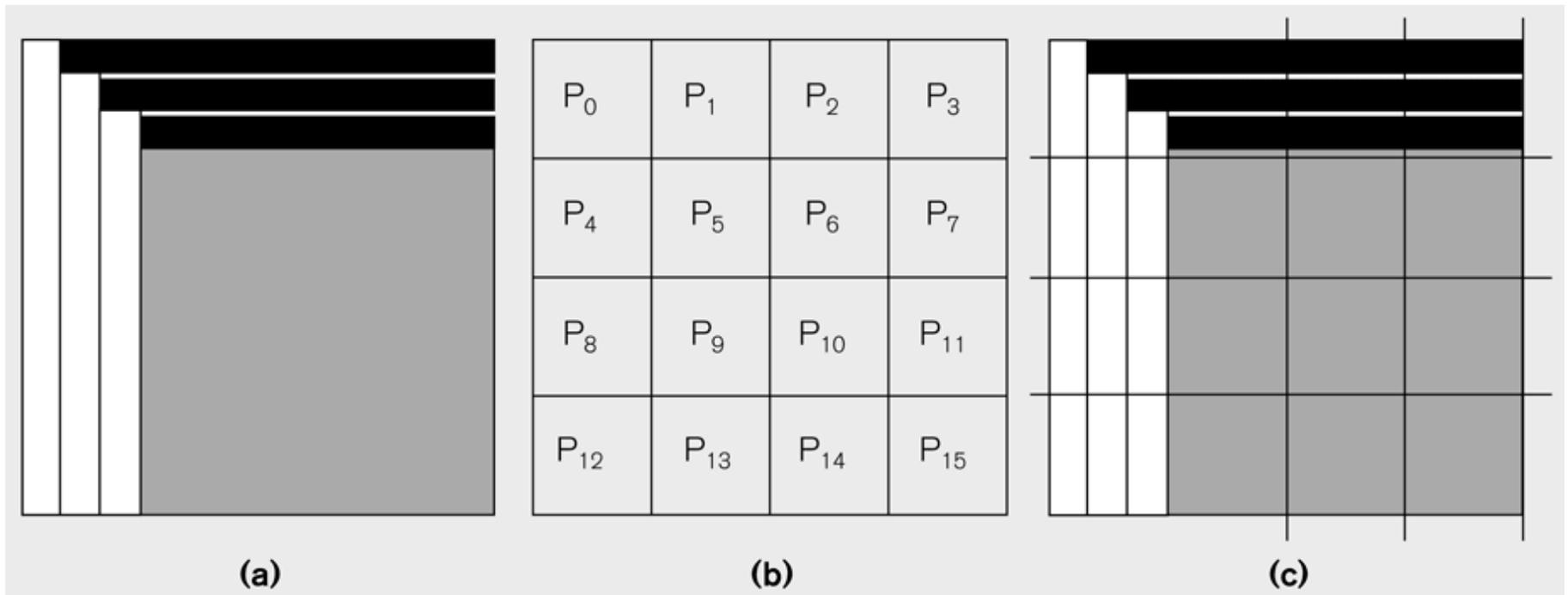
Overlap Regions to Reduce Sharing



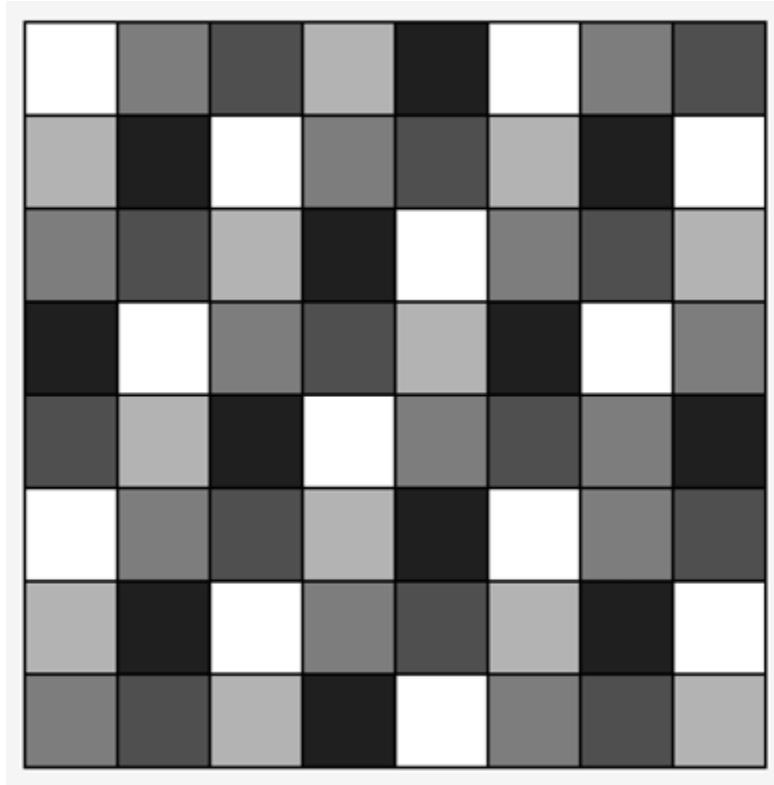
Computation Density



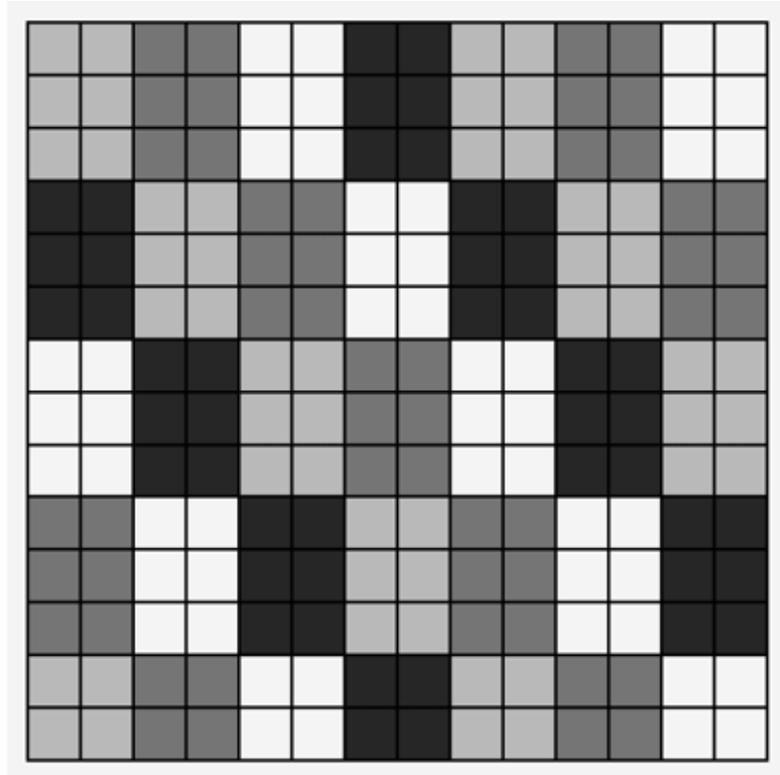
Computation Density



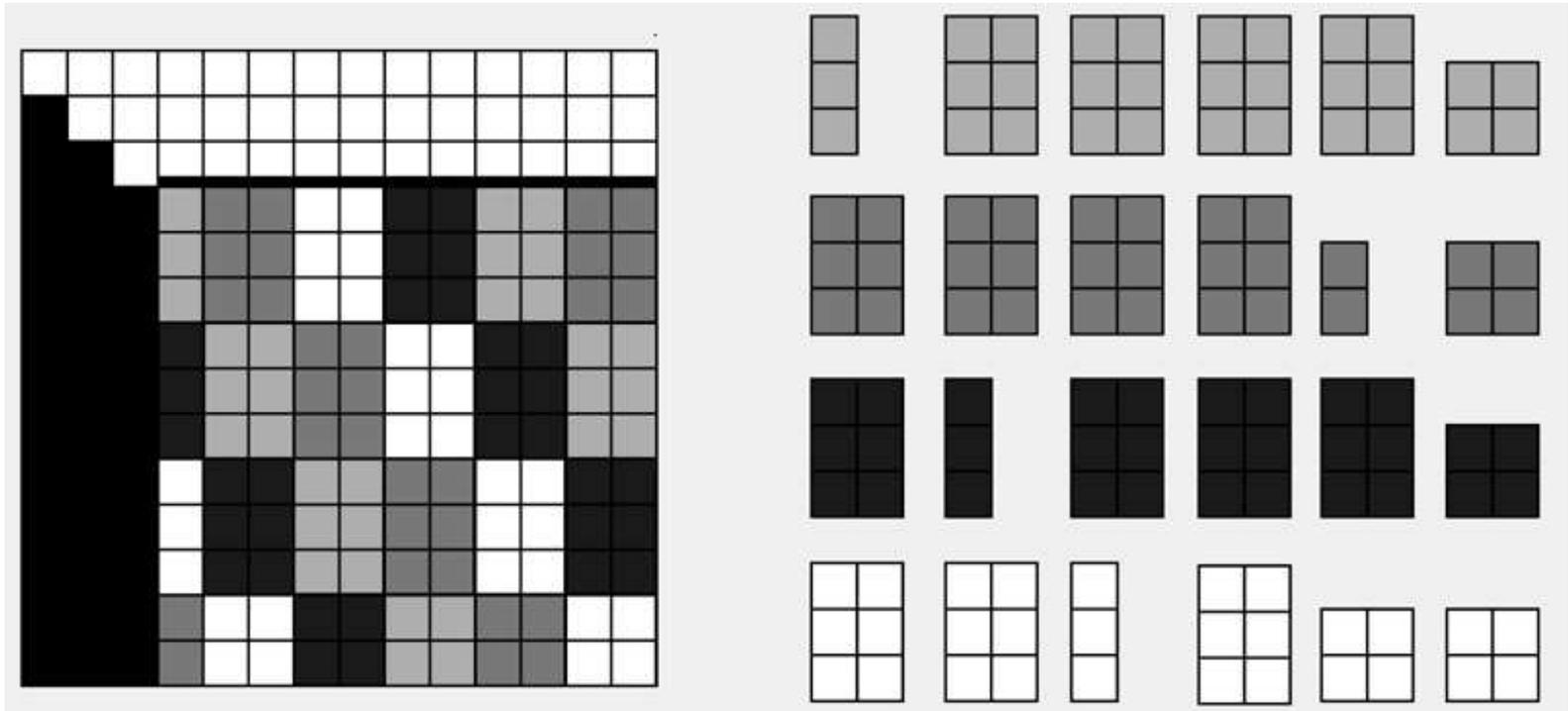
Cyclic Distribution



Block-Cyclic Distribution



Block-Cyclic Distribution for LU



Static versus Dynamic Distribution

Use a ***static allocation*** when the computational density is apparent

- ***array range***
- ***block distribution***
- ***cyclic distribution***
- etc.

Use a ***dynamic allocation*** when the density is discovered later

- ***work queue***

Work Queue

Break task into pieces

- Small enough that no single piece dominates
 - Big enough to make allocation overhead small
- and put them all in a queue

Each worker:

- Consume one task from queue
- Compute
- Report answer
- Repeat