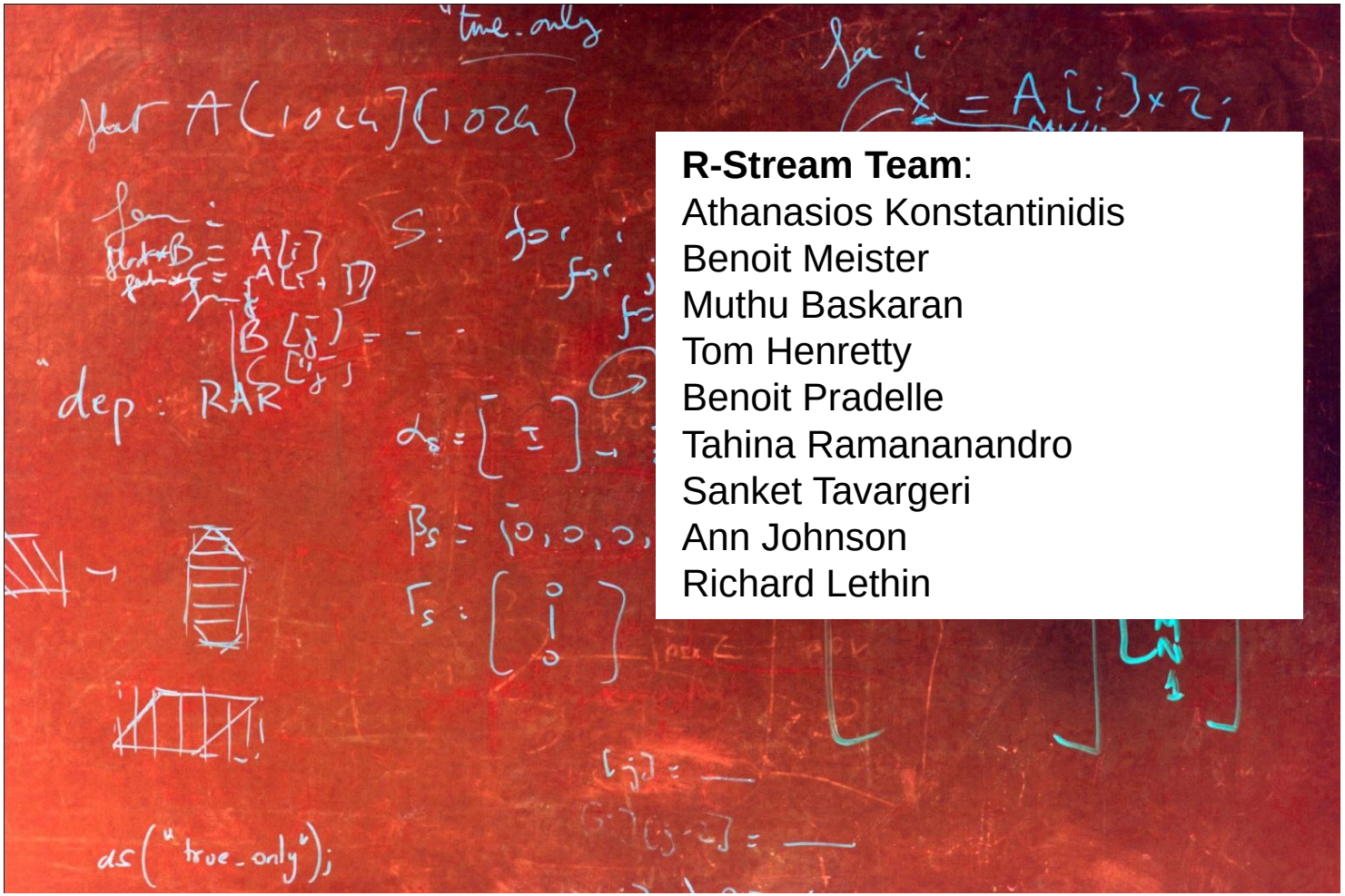
A GPU Run-Time for Event-Driven Task Parallelism

Reservoir Labs, Inc.



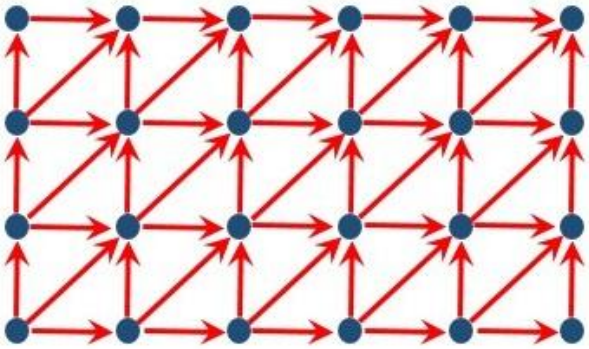
R-Stream Team:
Athanasios Konstantinidis
Benoit Meister
Muthu Baskaran
Tom Henretty
Benoit Pradelle
Tahina Ramananandro
Sanket Tavargeri
Ann Johnson
Richard Lethin

Massive data parallelism is required

- Hides global memory access latency

What if our program is not data-parallel ?

Dependence graph (DAG) of
an SPMD computation



GPU Programming with CUDA

Massive data parallelism is required

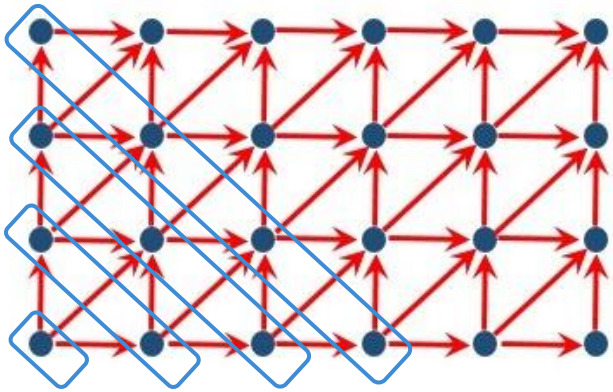
- Hides global memory access latency

What if our program is not data-parallel ?

- We find synchronous chunks of data-parallel computations i.e., wavefronts

Global synchronization overhead from repeated kernel invocations

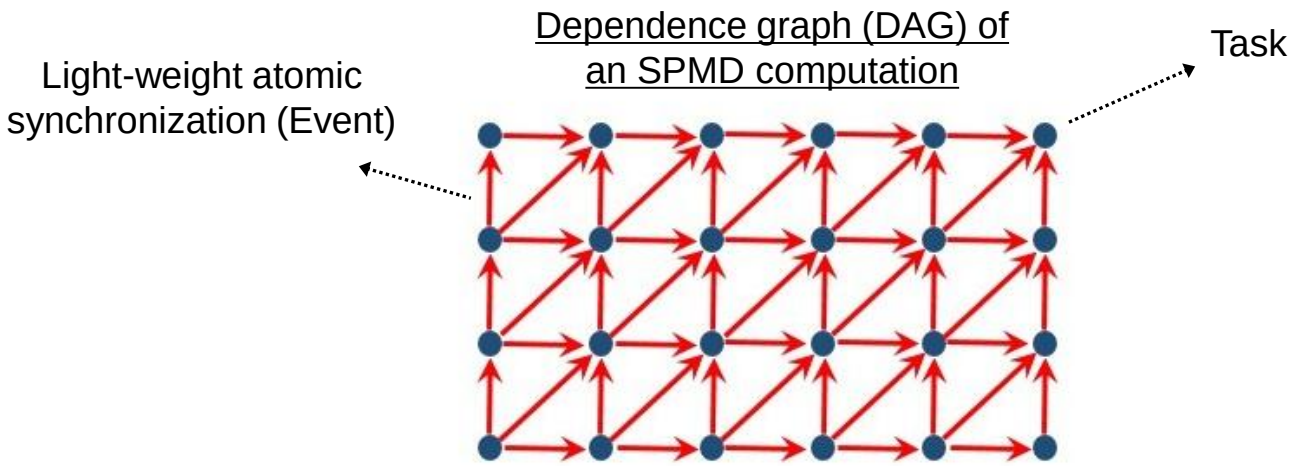
Dependence graph (DAG) of an SPMD computation



A GPU Run-Time for Task Parallelism

Implements an Event-Driven Tasks execution model

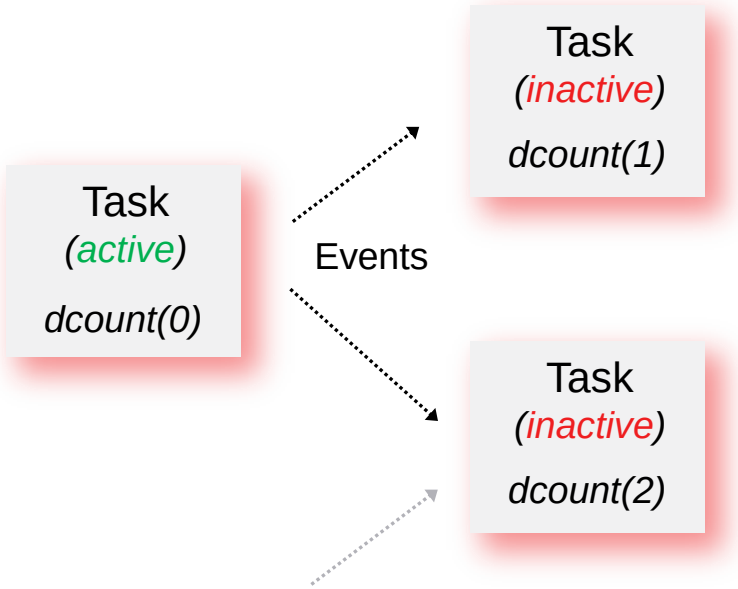
- A single persistent GPU kernel executes the entire DAG (manages thread-block-level parallelism)
- On-the-fly dependence resolution
- Light-weight synchronization based on atomics
- Work-stealing for load-balancing



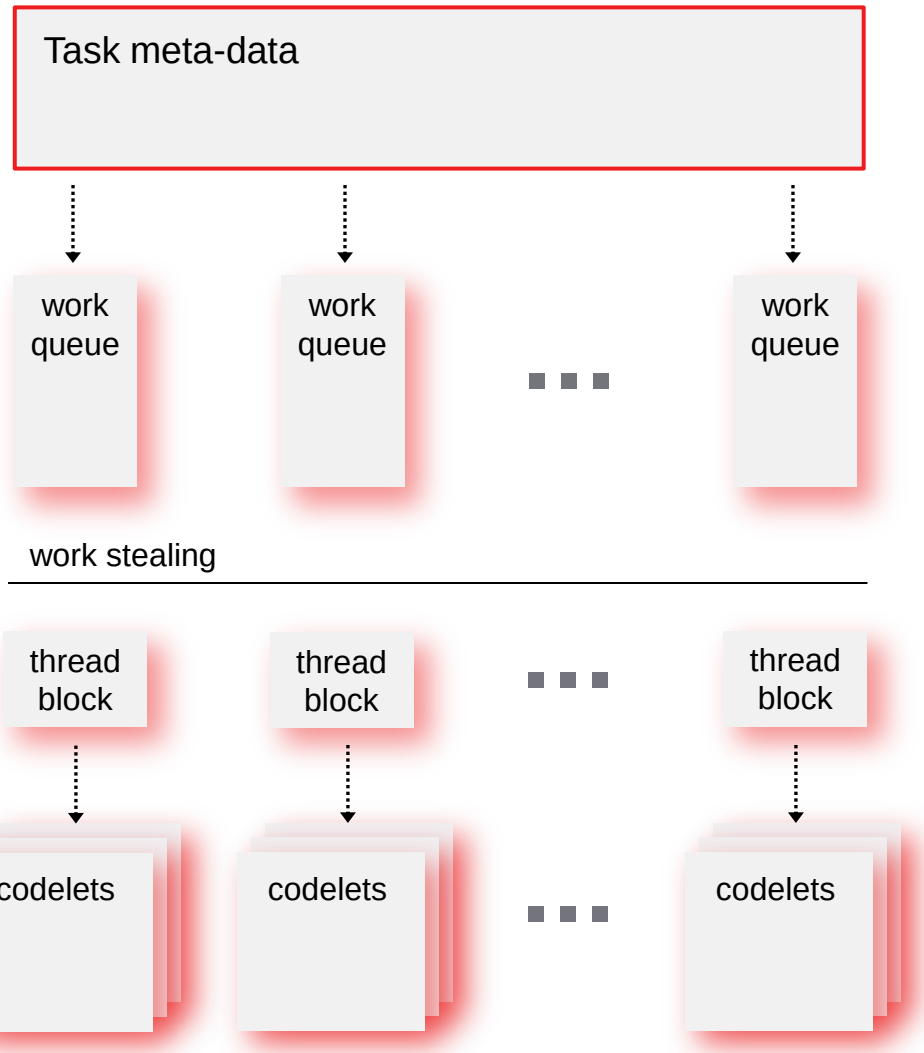
Dependence Resolution – Event-Driven Tasks (EDTs)

Dependence counters

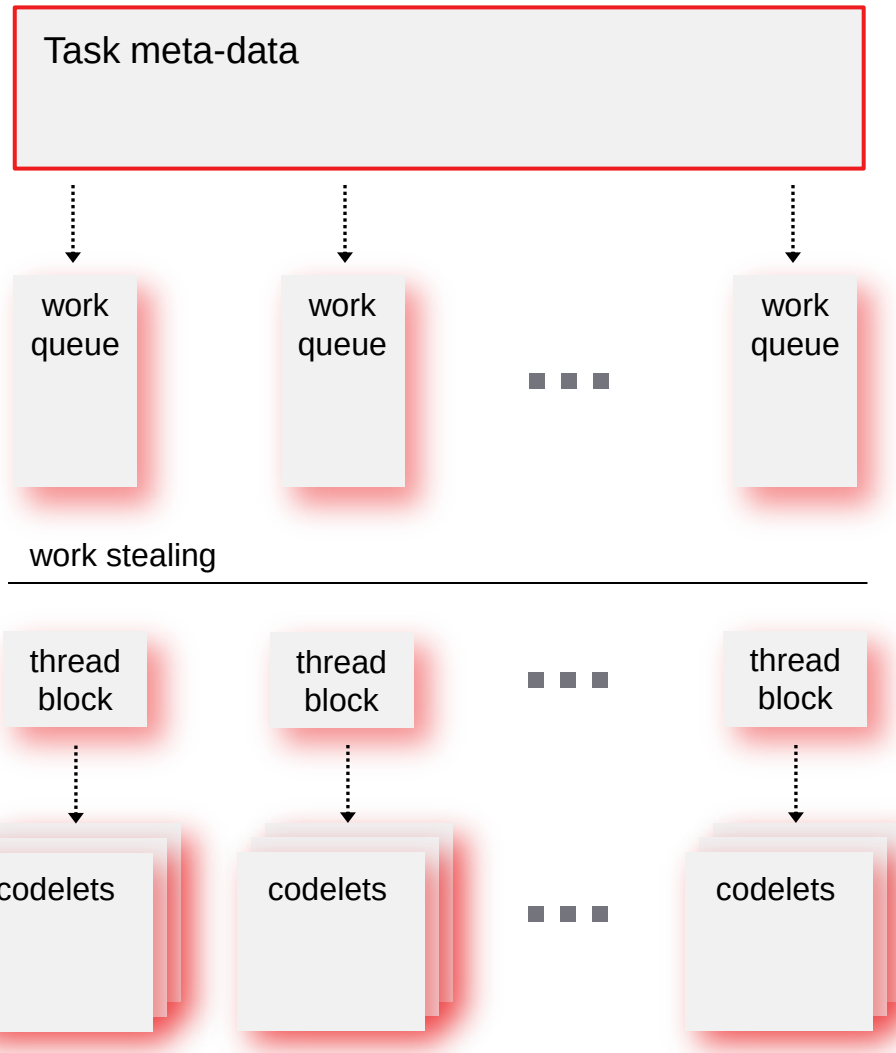
- Each task has a dependence counter (*dcount*)
- After task completion decrement successors' *dcount*
- Task becomes active if *dcount* becomes zero



Run-Time Architecture

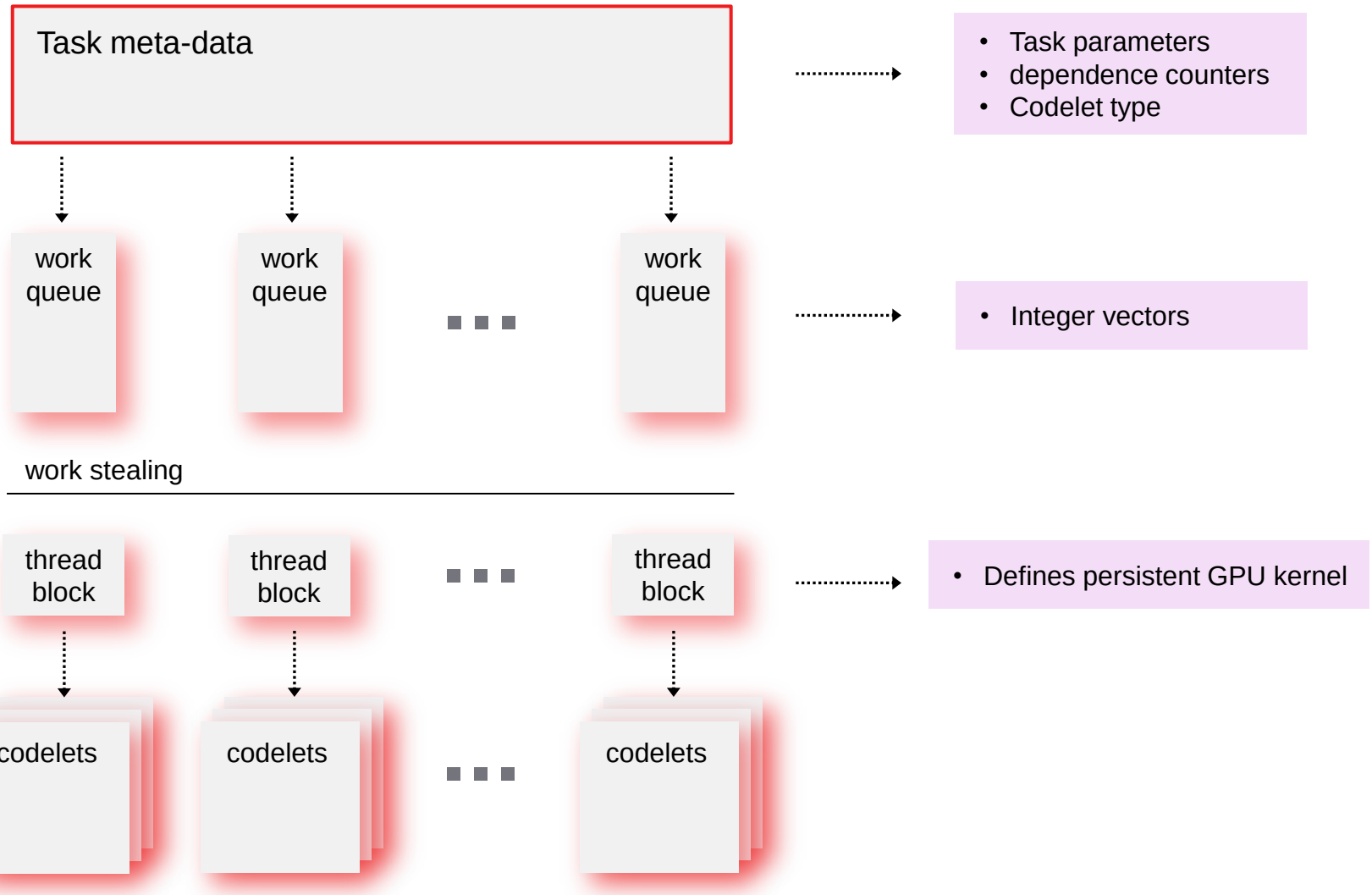


Run-Time Architecture

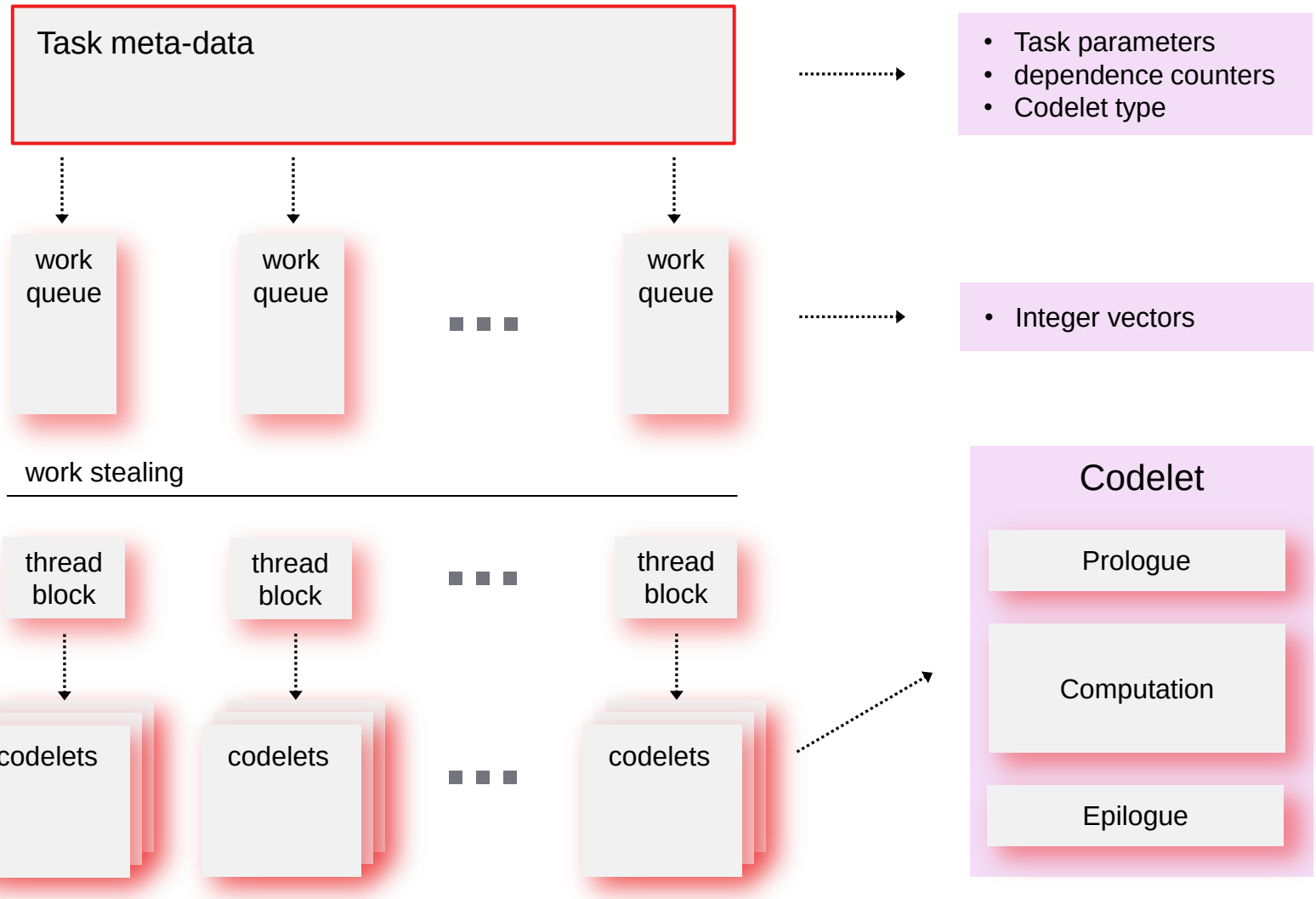


- Defines persistent GPU kernel

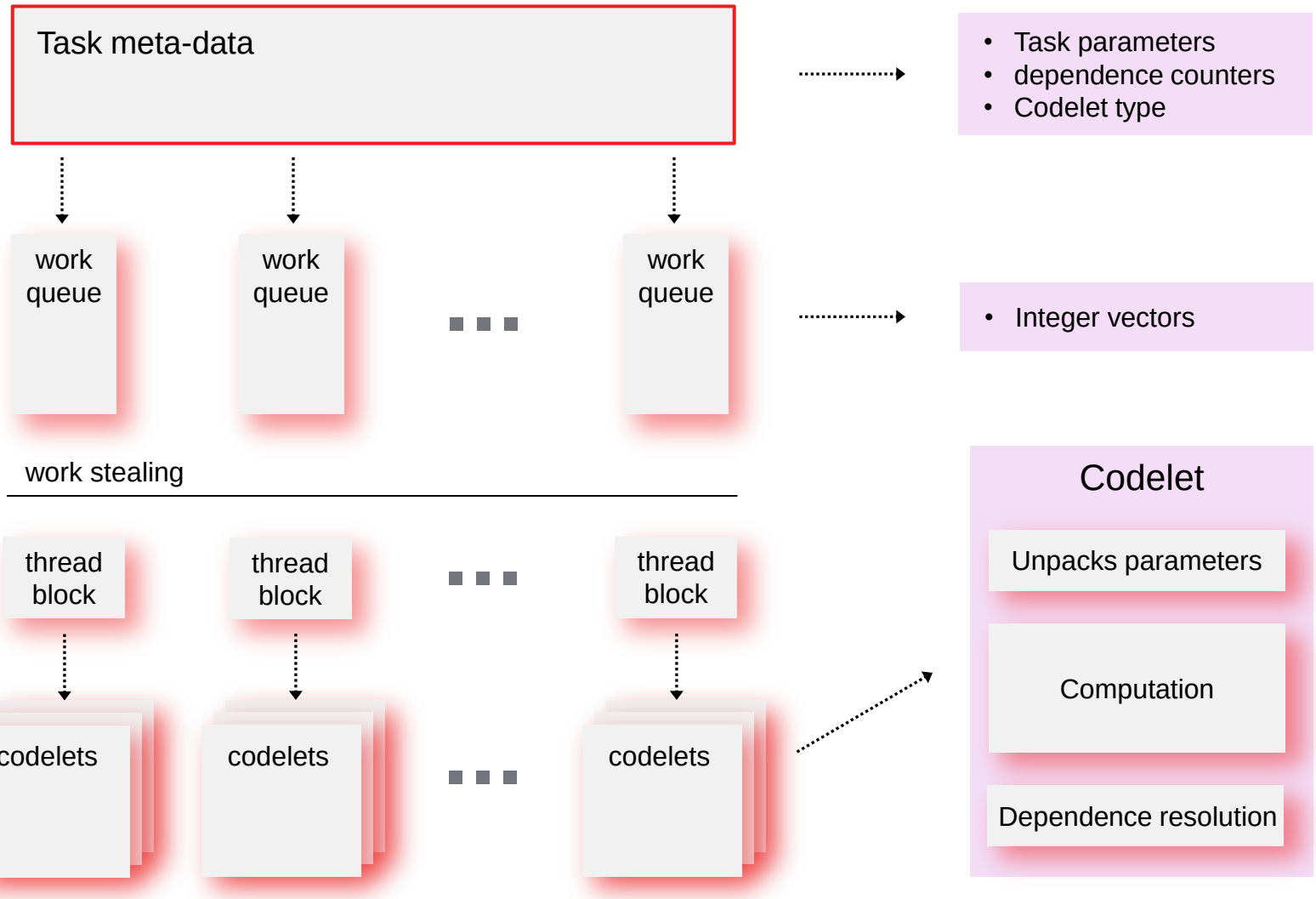
Run-Time Architecture



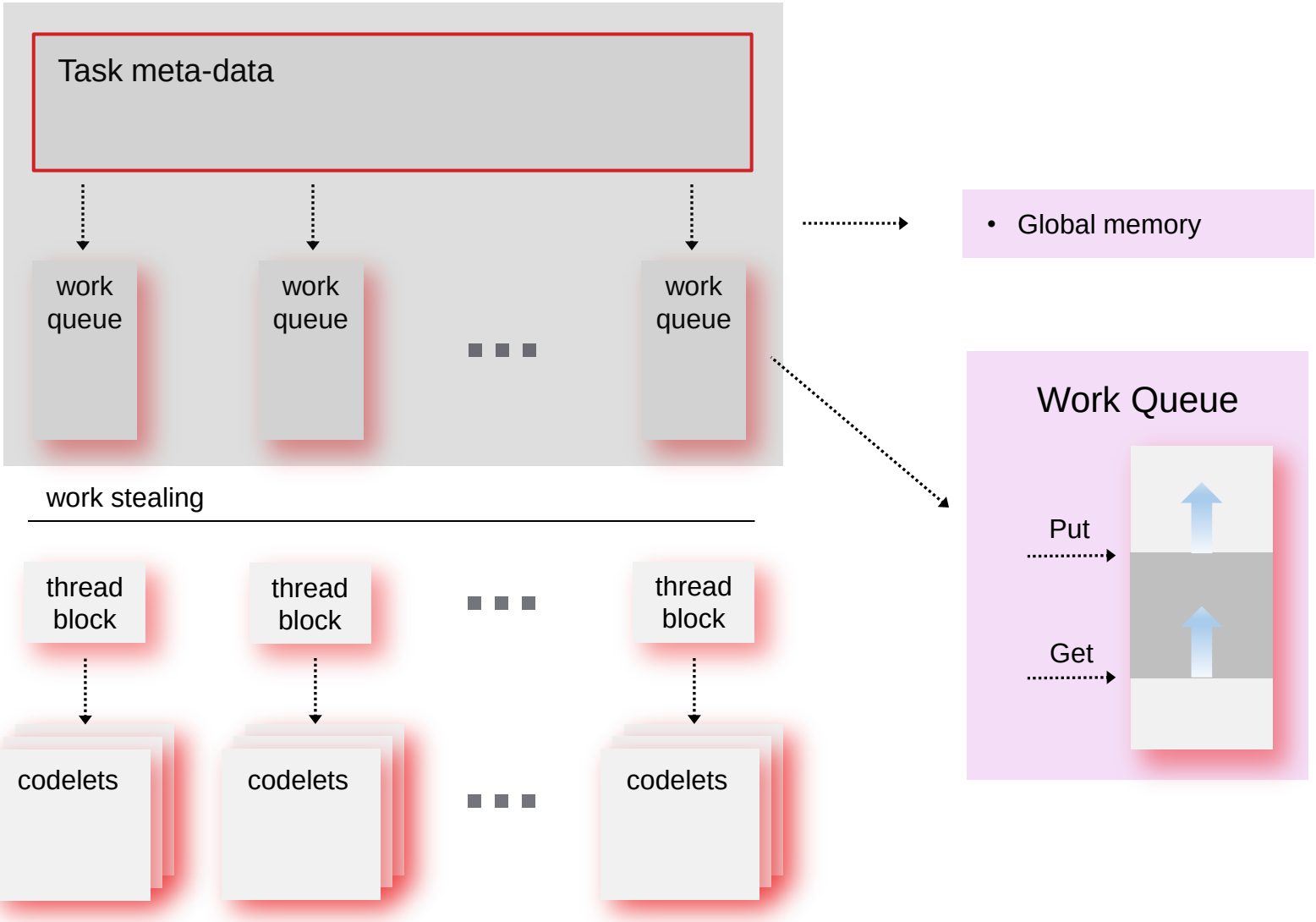
Run-Time Architecture



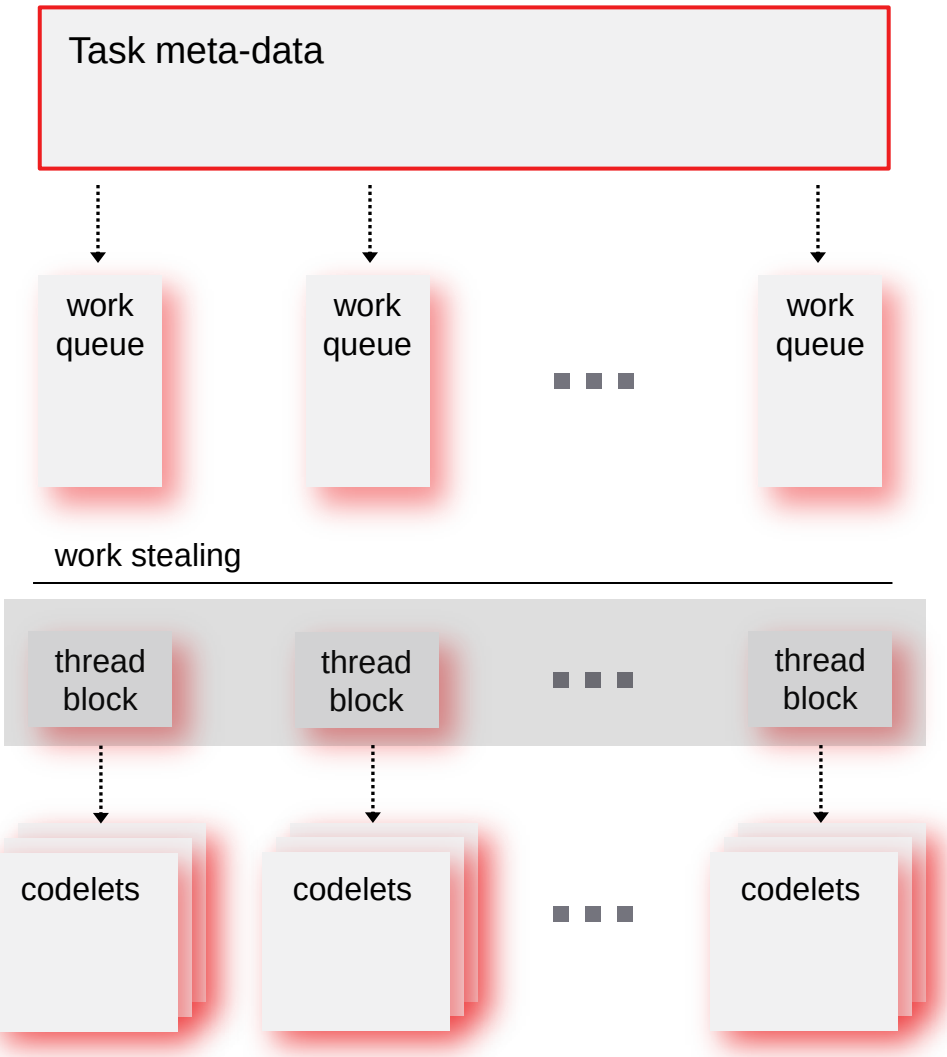
Run-Time Architecture



Run-Time Architecture



Run-Time Architecture



- Workers
- Unrestricted amount
- Max stealing rounds
- Intra-Thread-block configuration agnostic

Experimental Evaluation

Simple stencil programs from the PolyBench suite

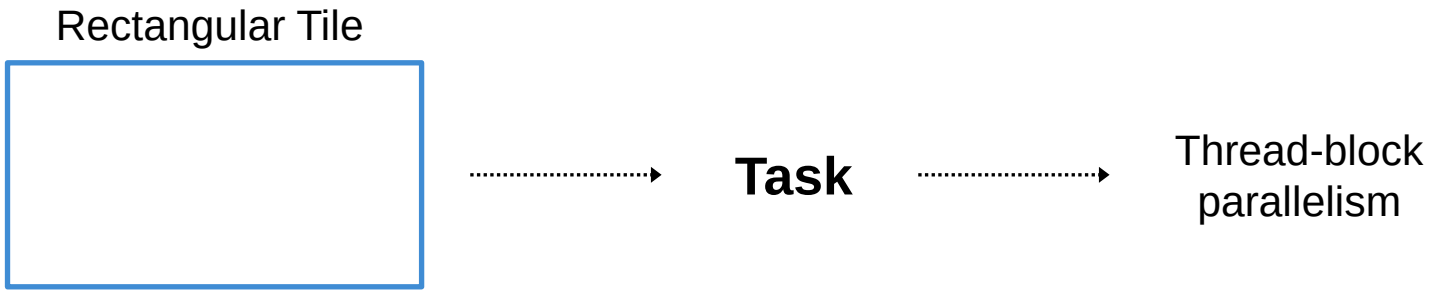
- Jacobi-2D 5pt, FDTD-2D, ADI

Compared against best known wavefront implementations

- Konstantinidis et al. LCPC 2013

Rectangular parametric tiling is applied

- For run-time tile-size exploration

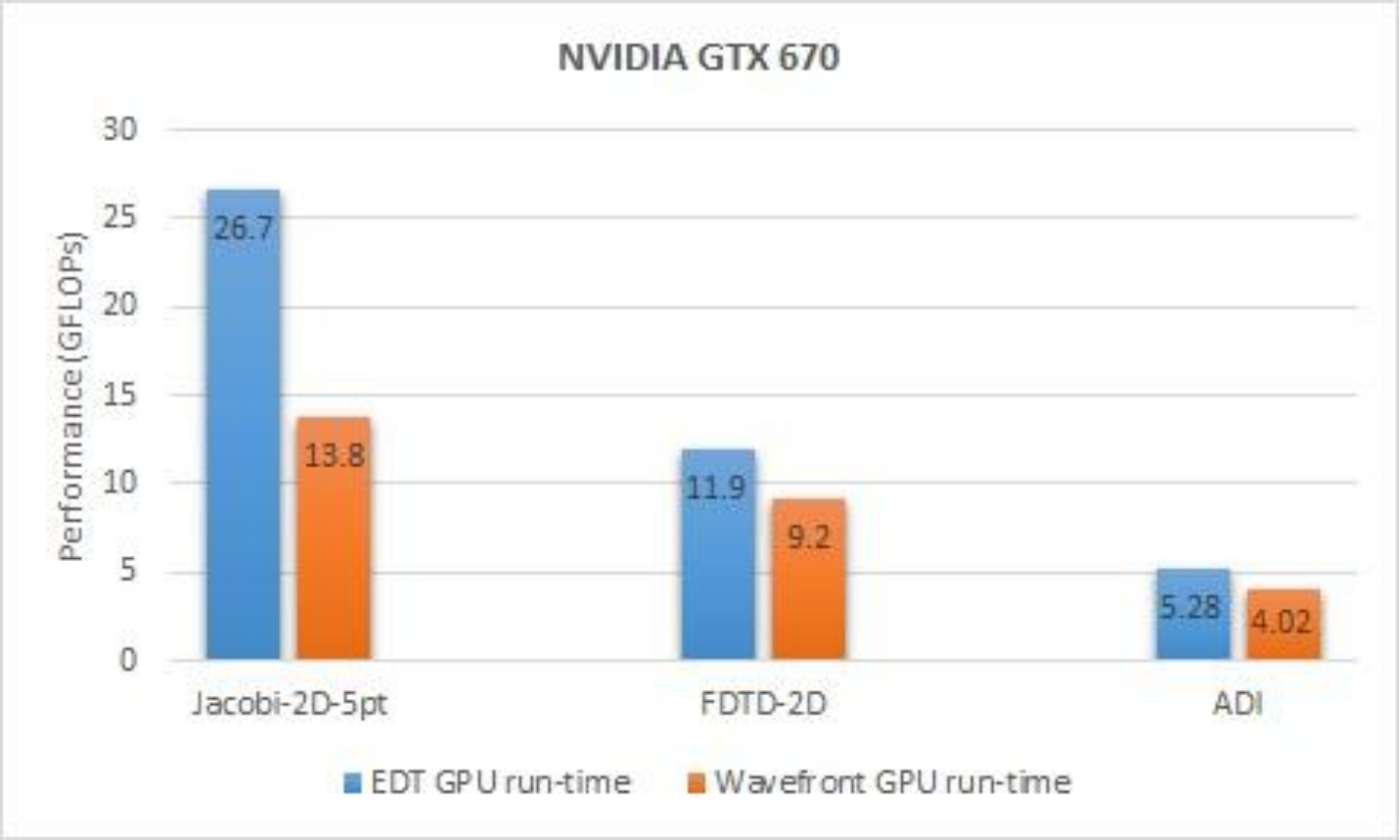


Experimental Evaluation

NVIDIA GTX 670

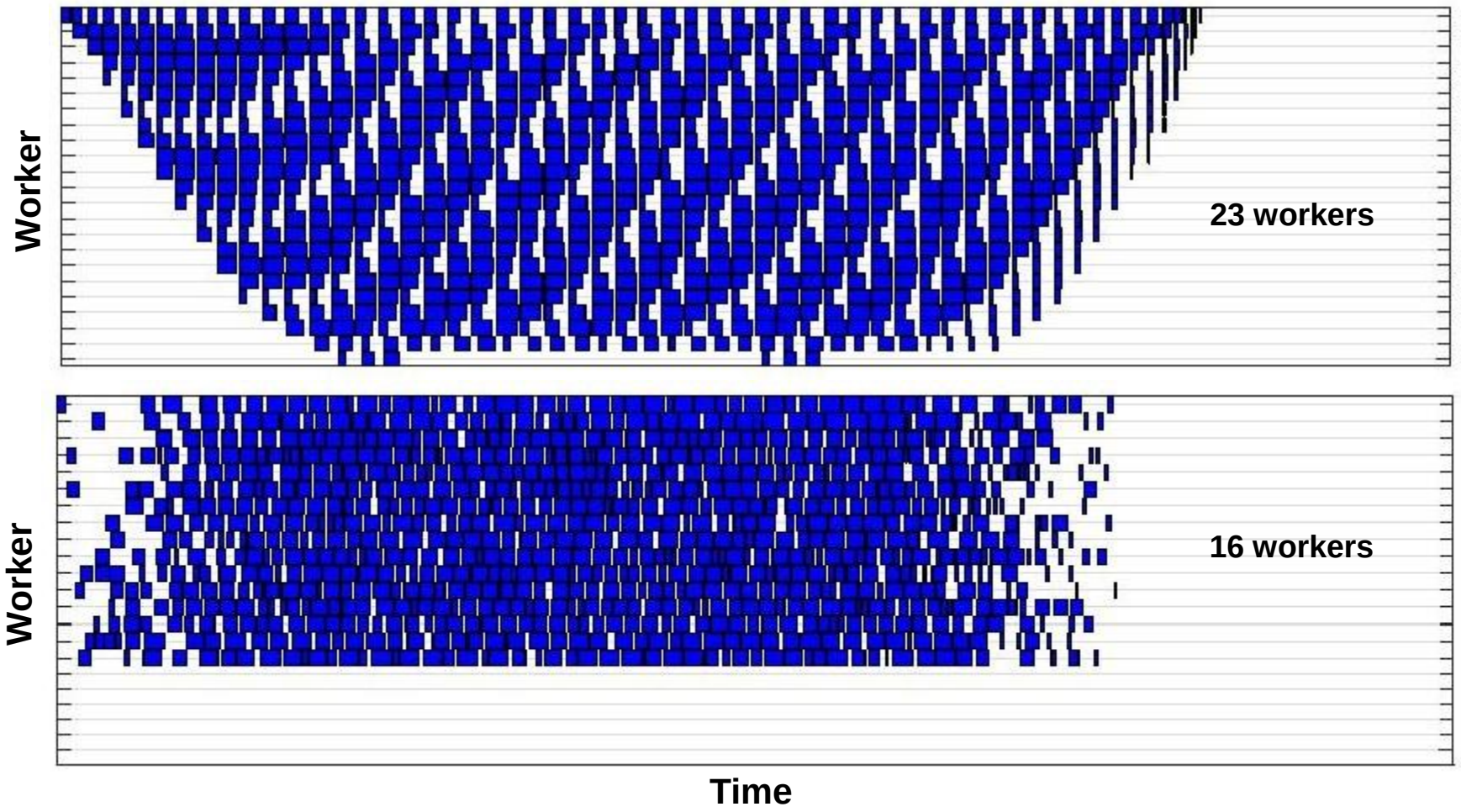
- Compute Capability: 3.0
- Driver/Runtime Version: 6.5
- Global Memory: 2GB
- Multiprocessors: 7
- ECC: OFF

Experimental Evaluation



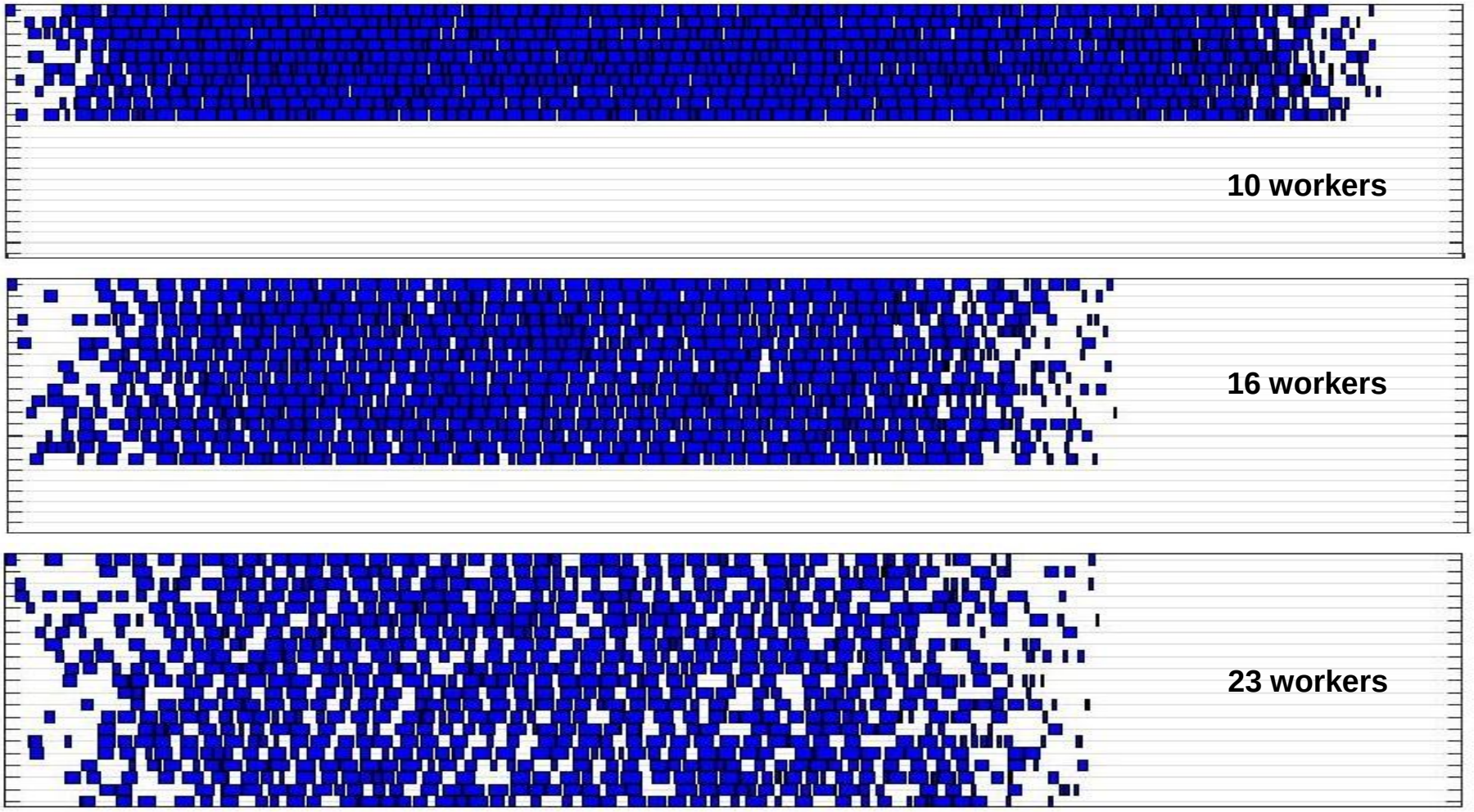
Experimental Evaluation

Jacobi 2D 5pt – Execution Timelines



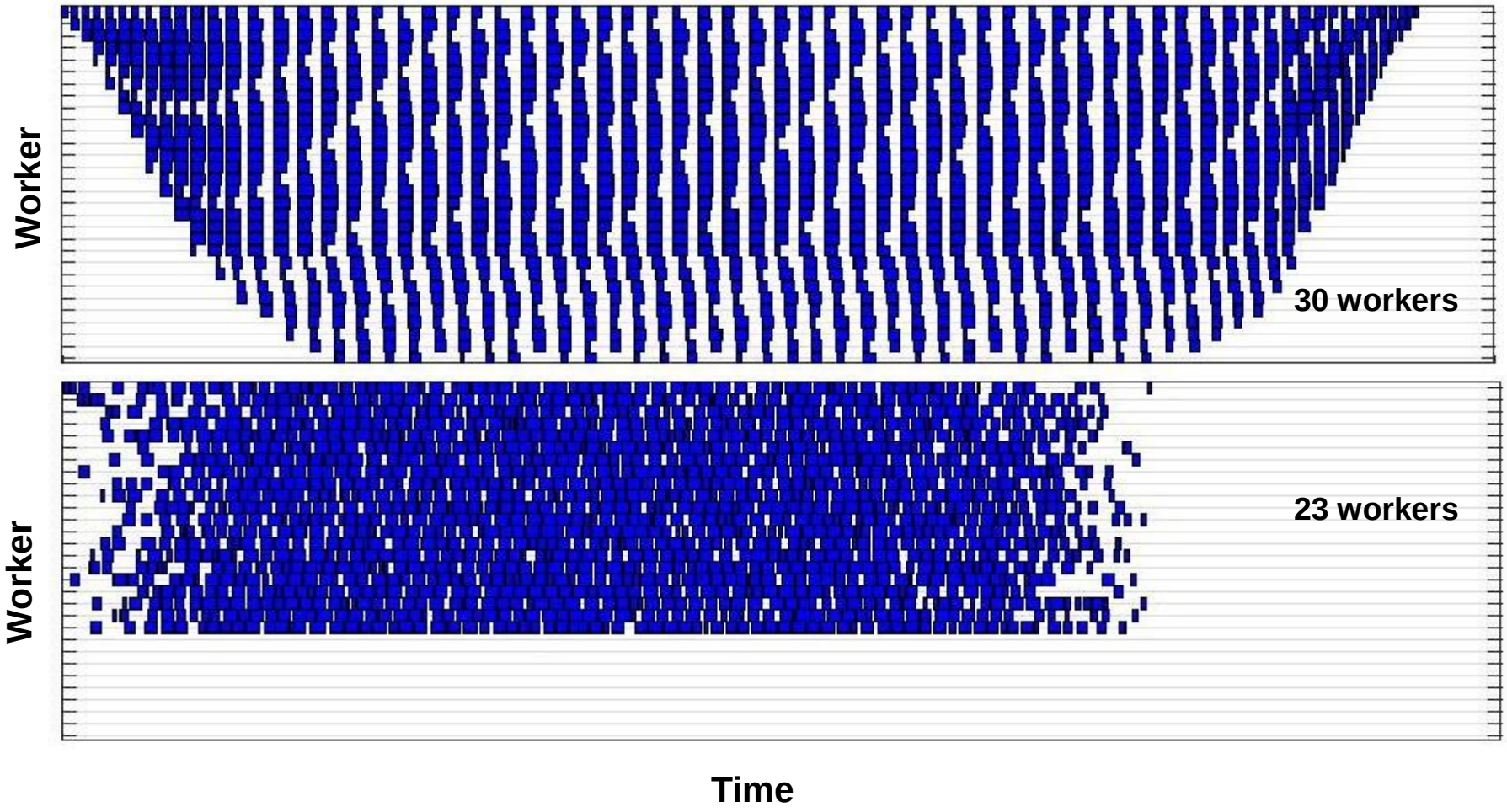
Experimental Evaluation

Jacobi 2D 5pt – Execution Timelines



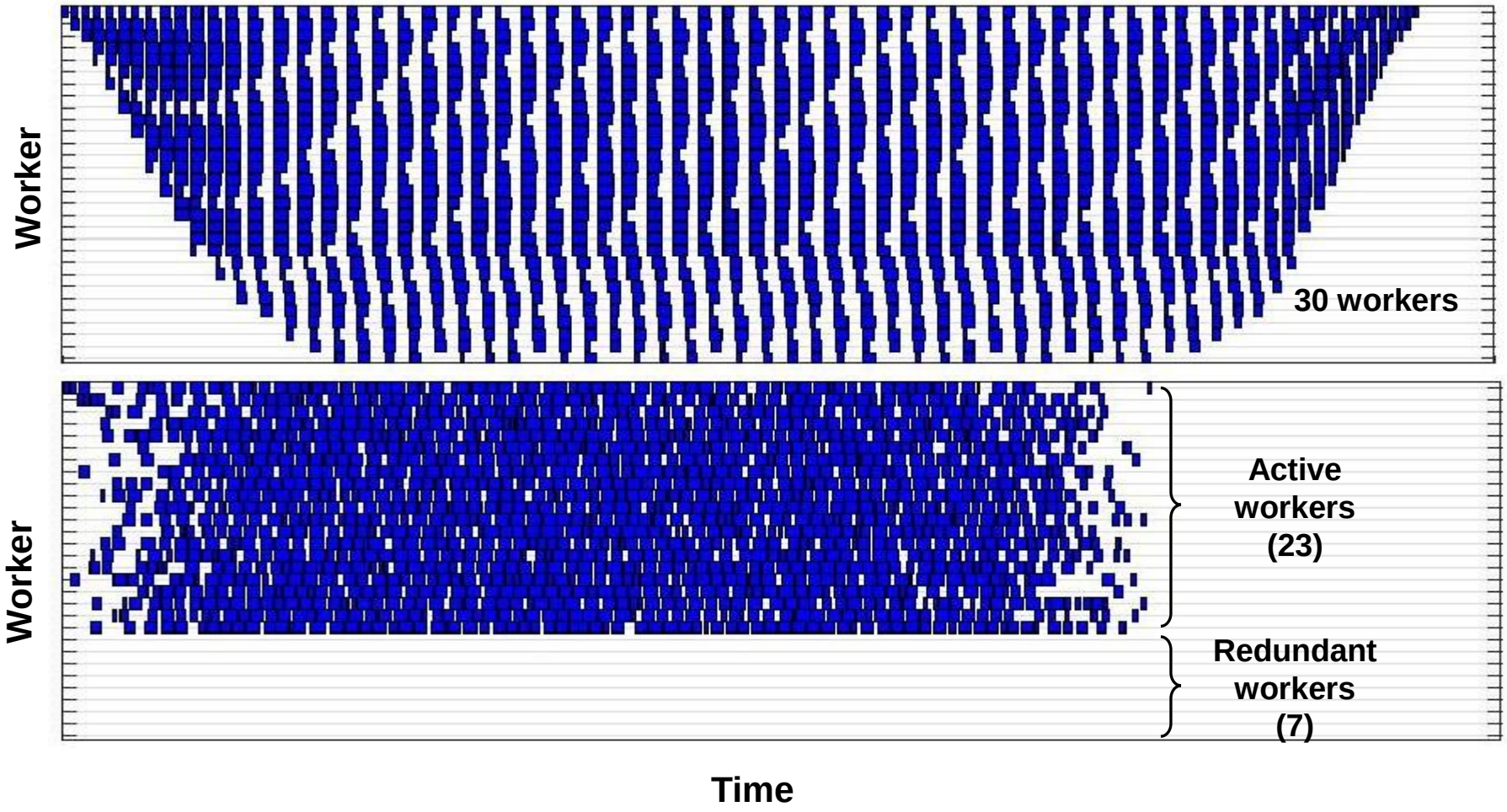
Experimental Evaluation

Jacobi 2D 5pt – Execution Timelines



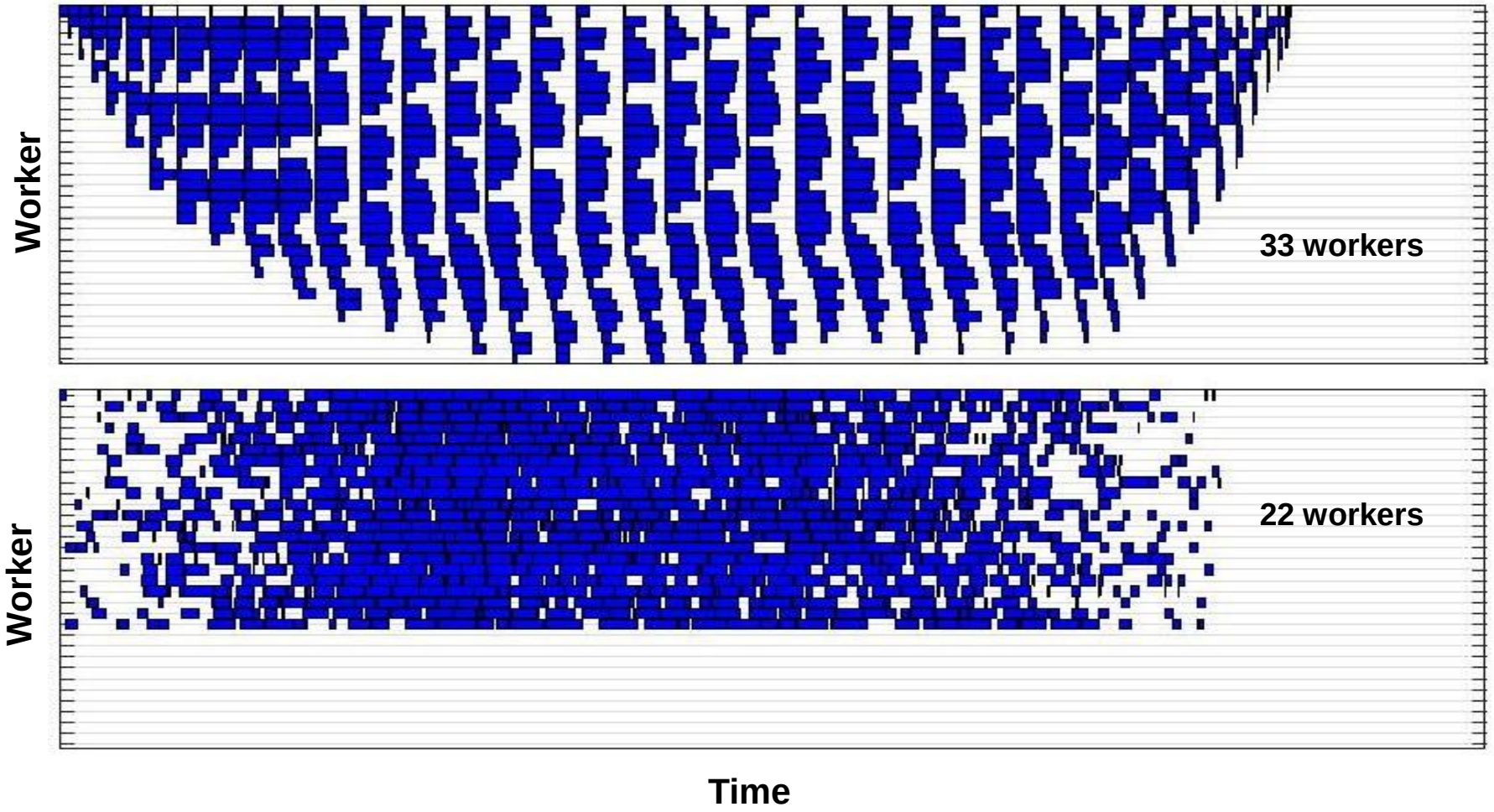
Experimental Evaluation

Jacobi 2D 5pt – Execution Timelines



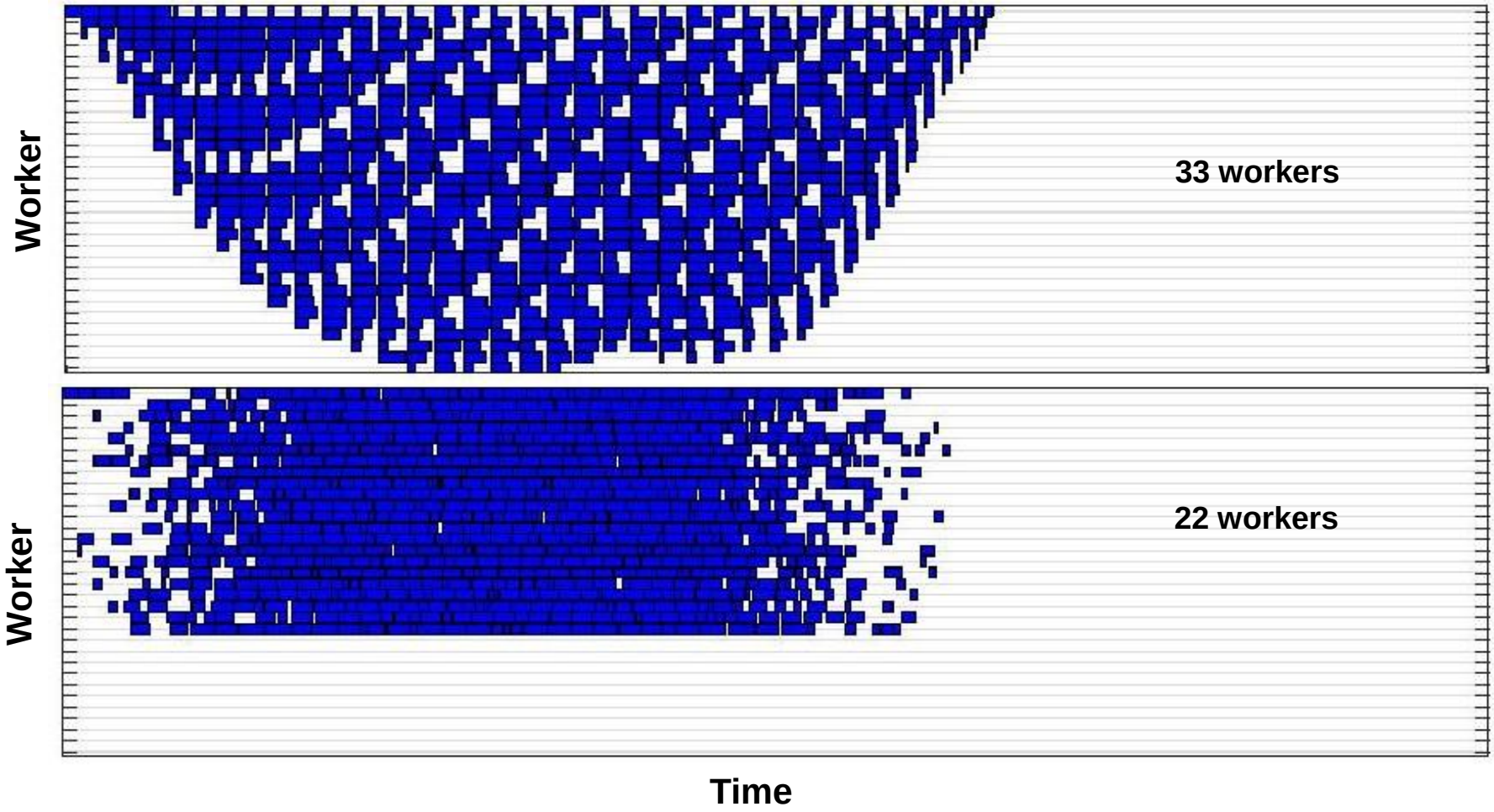
Experimental Evaluation

FDTD 2D – Execution Timelines



Experimental Evaluation

ADI – Execution Timelines



Conclusions

Effective task-parallelism with on-the-fly dependence resolution

Single persistent GPU kernel prevents global synchronization overhead

Evaluated against wavefront parallelism on stencil computations

The End

Questions ?