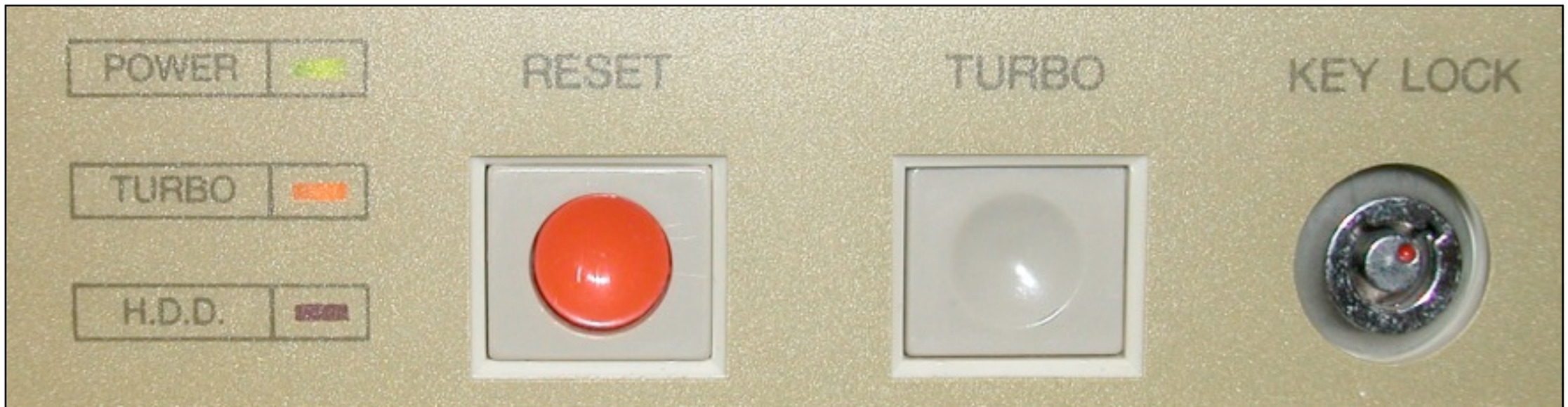
STURBO
Speeding up Conflicting Transactions

Jons-Tobias Wamhoff
Christof Fetzer
Technische Universität Dresden, Germany

Pascal Felber
Université de Neuchâtel, Switzerland

Motivation

- Concurrent data accesses may conflict
- Transactions must abort and retry
  - Defer retry until winner committed
- Boost speed of winner to reduce wait



AMD Turbo CORE

- *Asymmetric* multi-core processor
  - Deterministically provide *maximum performance* within power envelop (TDP)
- *P-states* characterize unique frequency and voltage with P0 being the highest
  - Allow different P-states for distinct CPUs
- Application power management allows *boosted* P-states
  - Software requests P0 and processor below TDP
  - Processor can run above base operating frequency

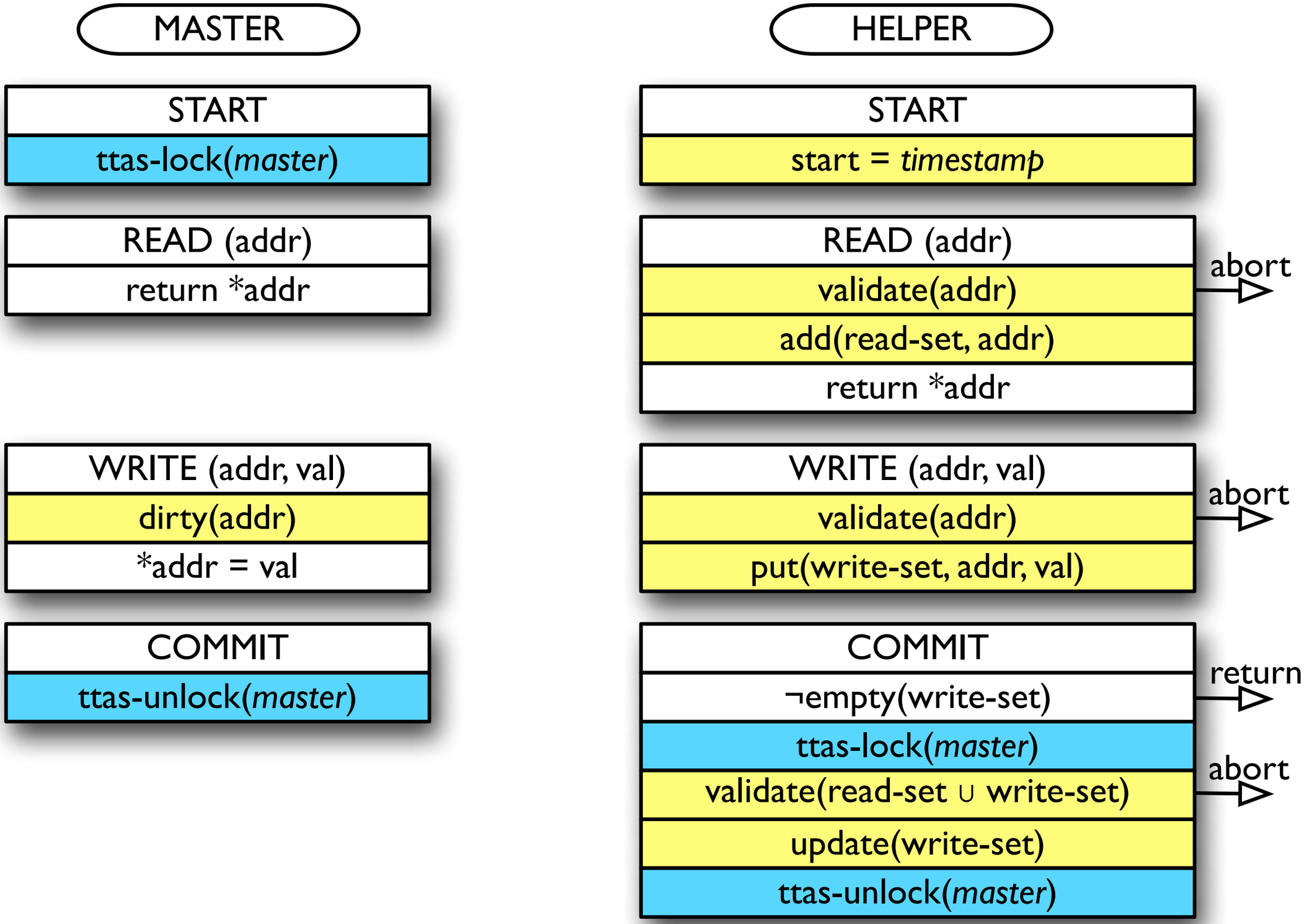
Possible Boosting Targets

- *FastLane* (FL) already has one fast thread
- *Transactional Mutex Locks* (TML) allows only single writer
- Boosting only commit periods might not compensate p-state switching latency

FASTLANE: General Idea

- 1 *pessimistic* **master** thread
  - Commits transactions without aborting
  - Minimal instrumentation and bookkeeping
  - Runs almost at sequential non-instrumented speed
- N *speculative* **helper** threads
  - Commit transactions only when not in conflict
  - Contribute progress without impairing on the performance of the master
  - Typically slower than STM threads

FASTLANE: Algorithm



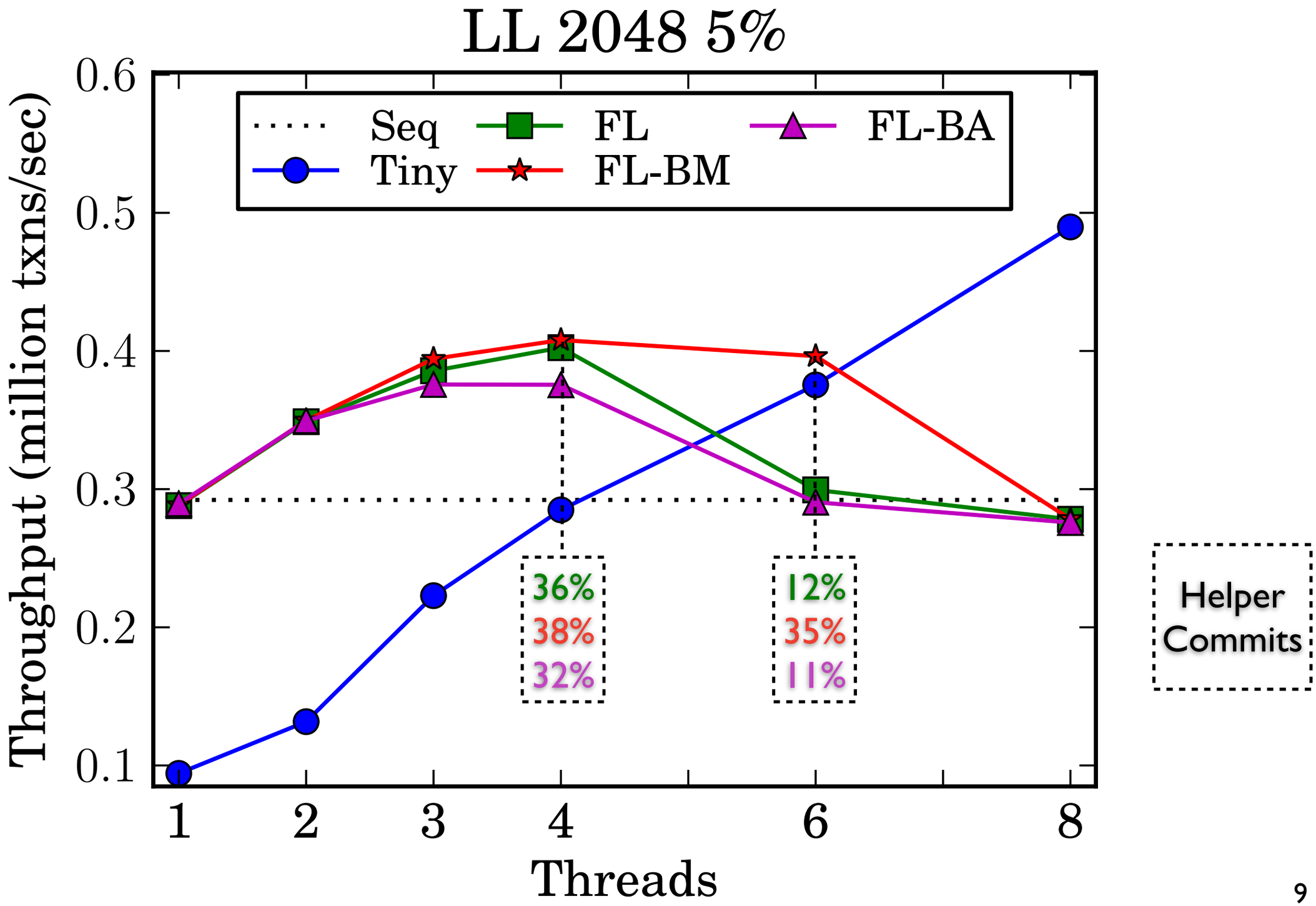
FASTLANE: Boosting

- Multiple options to apply boosting:
  - Boost *master* - slow helpers
  - Boost *owner of master lock*
    - Fast master and helpers commit
    - Slow helpers transaction processing
- Boost *all*
  - Slow down during waiting for master lock

Hardware Configuration

- AMD Bulldozer
  - 4 packages with 2 integer cores each (total 8)
  - 6 possible P-states
    - 2 boosted P-states (controlled by CPU)
    - 4 adjustable P-states (controlled by software)
- Frequencies min / base / max:
  - 1.9 / 3.1 / 4.0 GHz

Intset Benchmark



Challenges

- Implement boosting for TML, ...
- Investigate power efficiency
- Reduce latency of p-state switching

Thank you!