

# Enhancing Real-Time Behaviour of Parallel Applications using Intel TSX

Florian Haas, Stefan Metzloff, Sebastian Weis,  
and Theo Ungerer

Department of Computer Science,  
University of Augsburg,  
Germany

January 22, 2014



# Outline

Motivation

Approach

Evaluation

Conclusion & Outlook

- ▶ Increasing parallelism in safety-critical systems
  - ▶ Interferences at shared resources
  - ▶ Interferences at shared data structures
- ↪ Indeterministic execution behaviour:
  - ▶ Accesses have to be serialized
  - ▶ Mutual detention may result
- ↪ **Real-Time capable synchronization model needed**



A380, [1]



Google Driverless Car, [2]

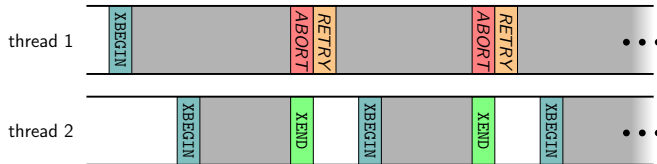
- ▶ Hardware Transactional Memory
  - ▶ Simplifies parallel programming
  - ▶ Optimistic lock-free parallel access can improve performance
- ▶ Transactional Synchronization Extension (TSX) implemented in current Intel Haswell micro architecture
  - ▶ No congestion in conflict-free case
  - ▶ Does not guarantee transaction commit
  - ▶ Transaction aborts and conflicts can happen at any time
  - ▶ Conflicting transactions must be scheduled to meet real-time requirements

↪ **Congestion controller for fair transaction execution is needed**

# Example

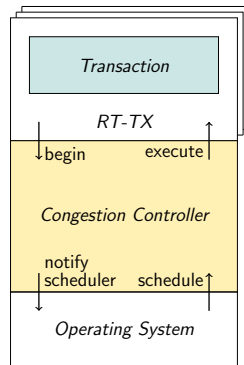
## Without congestion control

- ▶ Thread 2 starts its transaction after thread 1
- ▶ Transaction 2 leads to a conflict and abort of transaction 1



- ▶ Transaction 2 always aborts transaction 1 subsequently  
↳ Starvation of thread 1

- ▶ RT-TX library wraps TSX instructions
  - ▶ Global congestion controller monitors transaction execution
  - ▶ Schedule of transactions depends on statistics:
    - ▶ TX with less commits are preferred
    - ▶ TX with more commits are delayed
- ↪ Conflicts between TXs are resolved
- ↪ Progress of transactions is more deterministic



TSX instructions are wrapped:

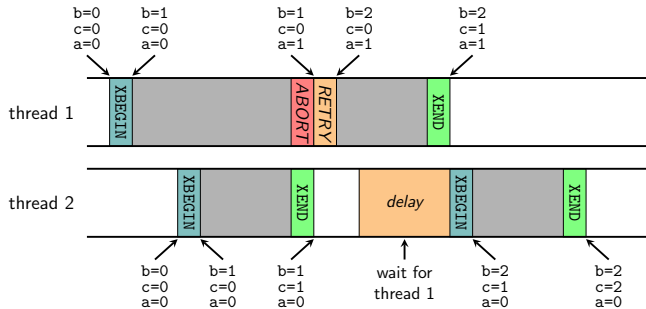
- ▶ `RTTX_begin`
  - ▶ registers transaction at congestion controller
  - ▶ waits for execution allowance
  - ▶ update TX status and begin counter
- ▶ `RTTX_end`
  - ▶ update TX status and commit counter
  - ▶ removes TX from congestion controller
- ▶ Abort handler
  - ▶ part of `RTTX_begin`
  - ▶ in case of abort, TX is restarted
  - ▶ because TX can always fail, a global lock is used for fallback
  - ▶ update TX status and abort counter (→ commit ratio decreases)

- ▶ Pausing transactions with high commit rate before executing:
    - ▶ Either: sleep for a specific amount of time
    - ▶ Or: wait for other transactions to commit
  - ▶ Prioritising transactions is difficult:
    - ▶ Scheduler must be aware of threads with transactions
    - ▶ Transaction commit rate is required for scheduling decision
    - ▶ Dependencies between transactions must be known
- ↪ **A new syscall is needed to notify the scheduler**

# Example

## With congestion control

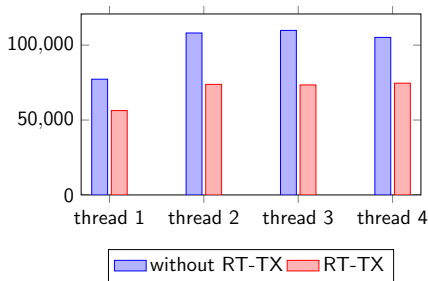
b: nr. of begins    c: nr. of commits    a: nr. of aborts



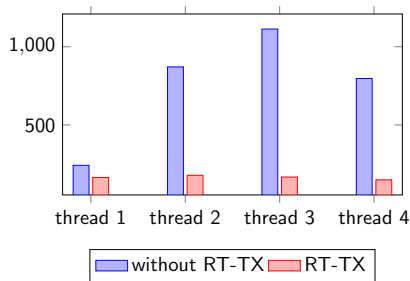
↪ Suspending thread 2 ensures progress of thread 1

- ▶ 4 threads:
  - ▶ 1 transaction per thread
  - ▶ All threads read from shared array
  - ▶ 1 thread also writes occasionally shared array
- ▶ Conflicts occur due to read and write operations
- ▶ Congestion controller ensures a fair execution
- ▶ Thread with highest completion rate is delayed
- ▶ Each threads is executed until TX has 10,000 commits
- ▶ Transaction aborts are measured:
  - ▶ Without RT-TX: congestion controller does not delay TXs
  - ▶ With RT-TX: congestion controller activated

Total abortions:



Max. subsequent abortions:



↪ RT-TX congestion controller leads to  
less total abortions and less subsequent abortions

## Conclusion & Outlook

- ▶ TSX eases parallel programming, but congestion still possible
- ▶ RT-TX provides congestion manager and scheduler integration
- ↪ RT-TX is real-time capable:
  - ▶ Fair transaction scheduling
  - ▶ Ensures transaction progress
- ▶ Future work:
  - ▶ Integration in Linux scheduler
  - ▶ Evaluate with MARSSX86 simulator
  - ▶ Consider different measures for scheduling decisions

# Questions?

- [1] <http://www.flickr.com/photos/8313254@N08/496320750/>
- [2] <http://www.flickr.com/photos/jurvetson/5499949739/>