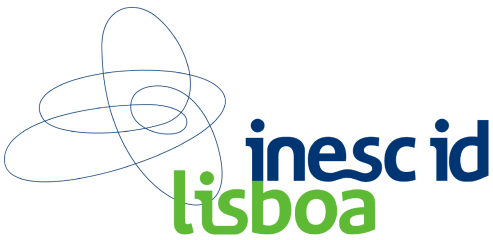
The moment of truth: are we done with STM?

Nuno Diegues, Paolo Romano, Luís Rodrigues

ndiegues@gsd.inesc-id.pt



Over 20 years of Transactional Memory

Over 20 years of Transactional Memory

Commodity processors with hardware support

Over 20 years of Transactional Memory

Processors by IBM (BG/Q and zEC12) and **Intel (Haswell)**

The question

Raise the question: **are we done with STM**?

The question

Raise the question: **are we done with STM**?

+ Hardware ought to be faster
+ Transparency and ease of use

The question

Raise the question: **are we done with STM**?

+ Hardware ought to be faster
+ Transparency and ease of use

- Research in STMs has evolved into a mature state
- Limited nature of hardware

The question

Raise the question: **are we done with STM**?

+ Hardware ought to be faster
+ Transparency and ease of use

- Research in STMs has evolved into a mature state
- Limited nature of hardware

What else is there to find?

Outline

1. (Quick) Motivation
2. Study Description
3. Compared Techniques
4. Results and Insights
5. Summary of Conclusions

Outline

Study

- Commodity hardware in Intel TSX
  - IBM processors target high performance computing

Study

- Commodity hardware in Intel TSX
  - IBM processors target high performance computing
  - Intel Haswell Xeon E3-1275v3 3.5GHz (3.9GHz Turbo)
  - 4 cores, 8 hardware threads (via hyper-threading)
  - 4x32KB L1 caches, 4x256KB L2 caches, 8MB L3 cache

Study

- Commodity hardware in Intel TSX
  - IBM processors target high performance computing
  - Intel Haswell Xeon E3-1275v3 3.5GHz (3.9GHz Turbo)
  - 4 cores, 8 hardware threads (via hyper-threading)
  - 4x32KB L1 caches, 4x256KB L2 caches, 8MB L3 cache
- Standard metrics for evaluation
  - Time to complete benchmarks
  - Power consumed (collected via Intel RAPL)
  - Relative to sequential, non-instrumented executions

Study

- Commodity hardware in Intel TSX
  - IBM processors target high performance computing
  - Intel Haswell Xeon E3-1275v3 3.5GHz (3.9GHz Turbo)
  - 4 cores, 8 hardware threads (via hyper-threading)
  - 4x32KB L1 caches, 4x256KB L2 caches, 8MB L3 cache
- Standard metrics for evaluation
  - Time to complete benchmarks
  - Power consumed (collected via Intel RAPL)
  - Relative to sequential, non-instrumented executions
  - **Combined metric:** Speedup / KJoules

Study

- Commodity hardware in Intel TSX
  - IBM processors target high performance computing
  - Intel Haswell Xeon E3-1275v3 3.5GHz (3.9GHz Turbo)
  - 4 cores, 8 hardware threads (via hyper-threading)
  - 4x32KB L1 caches, 4x256KB L2 caches, 8MB L3 cache
- Standard metrics for evaluation
  - Time to complete benchmarks
  - Power consumed (collected via Intel RAPL)
  - Relative to sequential, non-instrumented executions
  - **Combined metric:** Speedup / KJoules
- STAMP benchmarks (excluded Bayes) with standard parameters

Outline

Compared Techniques

- Locks
- STM
- HTM
- Hybrid TM

Compared Techniques - **Locks**

All benchmarks used an interface with the *atomic* construct:

- **GL**: single global lock
- **FL**: fine-grained locks — *per-application* effort

Compared Techniques - **STM**

Compared Techniques - **STM**

- **TL2**: commit-time locking
- **NOrec**: aimed at low thread count (single commit lock)
- **TinySTM**: encounter-time locking
- **SwissTM**: mixed encounter-time and commit-time locking

Compared Techniques - **HTM**

Intel TSX is single version, ensures strong isolation and allows nesting.

Most important it is **best-effort**:

Compared Techniques - **HTM**

Intel TSX is single version, ensures strong isolation and allows nesting.

Most important it is **best-effort**:

- No transaction is guaranteed to commit
- Exhausting cache lines with transactional footprint
- Architectural states, instructions, traps

Compared Techniques - **HTM**

Intel TSX is single version, ensures strong isolation and allows nesting.

Most important it is **best-effort**:

- No transaction is guaranteed to commit
- Exhausting cache lines with transactional footprint
- Architectural states, instructions, traps
- Fallback path must be provided in software
  - address to routine provided on XBEGIN

Compared Techniques - **HTM**

Intel TSX is single version, ensures strong isolation and allows nesting.

Most important it is **best-effort**:

- No transaction is guaranteed to commit
- Exhausting cache lines with transactional footprint
- Architectural states, instructions, traps
- Fallback path must be provided in software
  - address to routine provided on XBEGIN
- **TSX-GL** and **TSX-FL**

Compared Techniques - **HyTM**

Use an STM in the fallback path of TSX:

- **TSX-TL2** with reduced hardware transactions
- **TSX-NOrec** simpler, since NOrec has a single lock

Outline

STAMP results

Workload Characterization:

	Time in Tx (%)	Contention
kmeans	low (7)	low
ssca2	low (17)	low
intruder	medium (33)	high
vacation	high (89)	low
genome	high (97)	low
yada	high (99)	medium
labyrinth	high (100)	high

STAMP results

Workload Characterization:

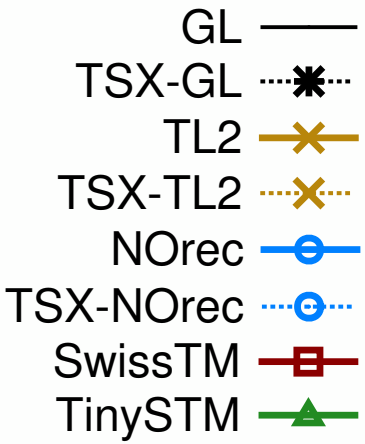
		Time in Tx (%)	Contention
L	kmeans	low (7)	low
L	ssca2	low (17)	low
M	intruder	medium (33)	high
M	vacation	high (89)	low
H	genome	high (97)	low
H	yada	high (99)	medium
H	labyrinth	high (100)	high

STAMP results

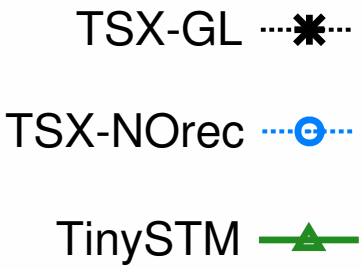
Characterization of the Techniques

		Most Performant	Least Power Consumption
L	kmeans		
L	ssca2		
M	intruder		
M	vacation		
H	genome		
H	yada		
H	labyrinth		

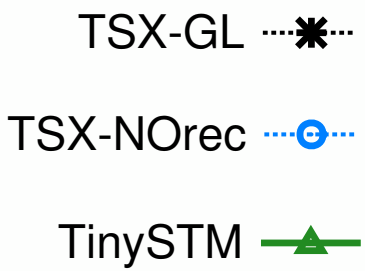
Plot labels



Plot labels

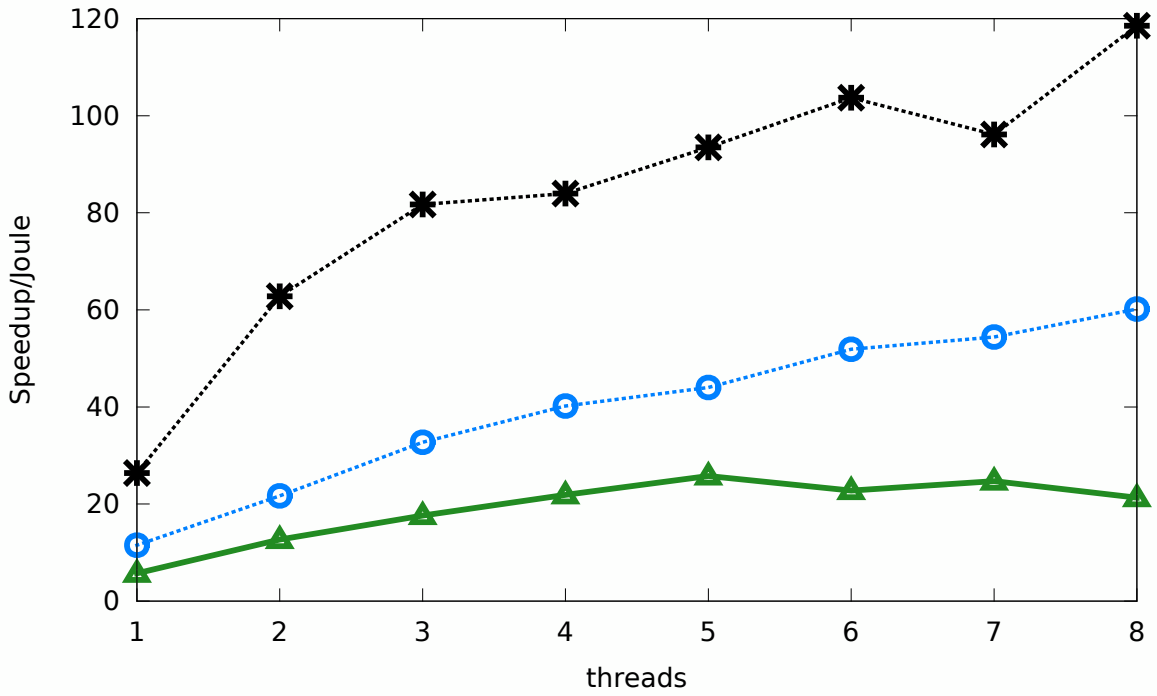


Plot labels



Speedup / **KJoule** along increasing threads

kmeans - low intensity



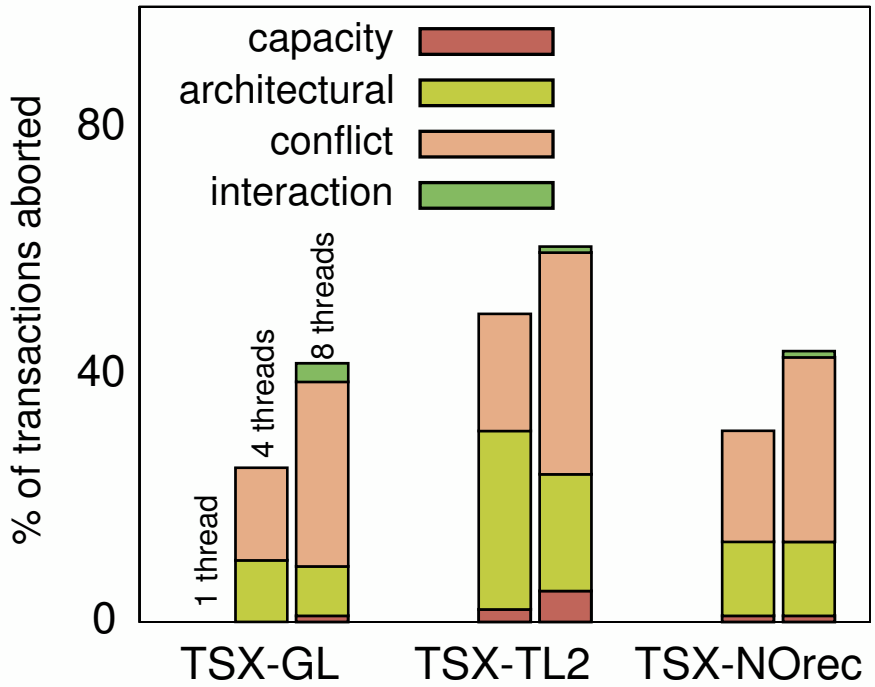
TSX-GL

TSX-NOrec

TinySTM

- Sequential overhead is noticeable
- **GL** allows some concurrency due to **L** workload
- HyTMs lag behind due to the STMs poor performance

STAMP results - **low intensity of transactions**



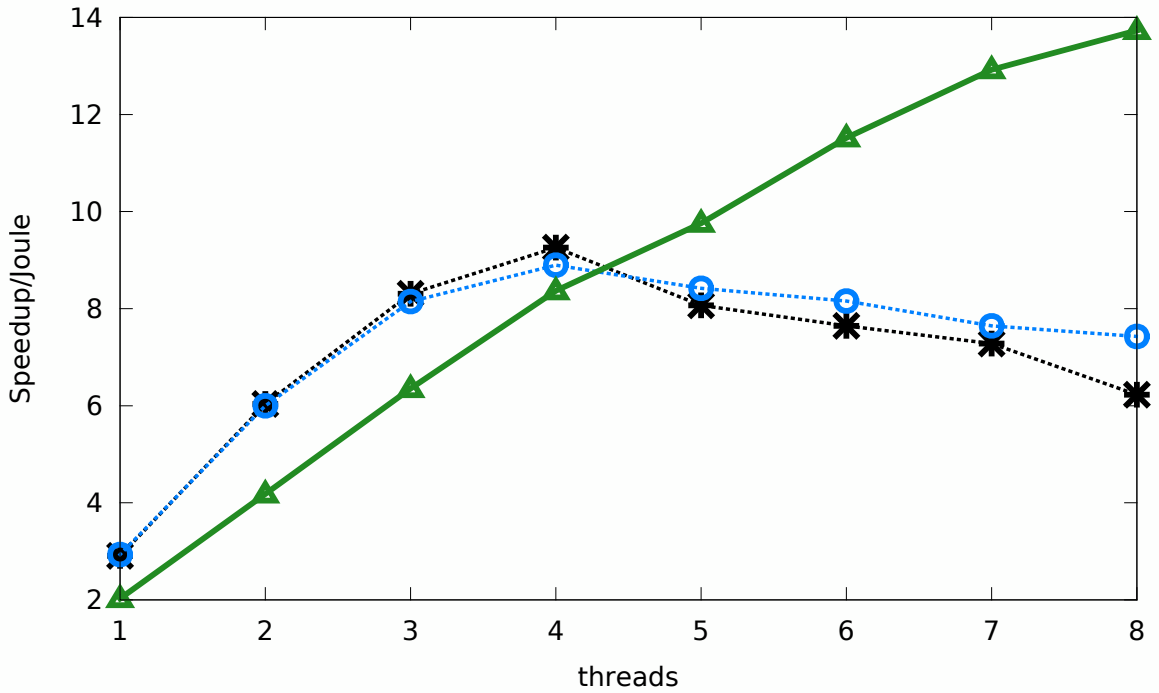
- 1 thread has negligible aborts
- STMs have 15% abort rate

STAMP results - **low intensity of transactions**

Characterization of the Techniques

		Most Performant	Least Power Consumption
L	kmeans	TSX-GL	TSX-GL
L	ssca2	TSX-GL	TSX-GL
M	intruder		
M	vacation		
H	genome		
H	yada		
H	labyrinth		

intruder - medium intensity



TSX-GL

TSX-NOrec

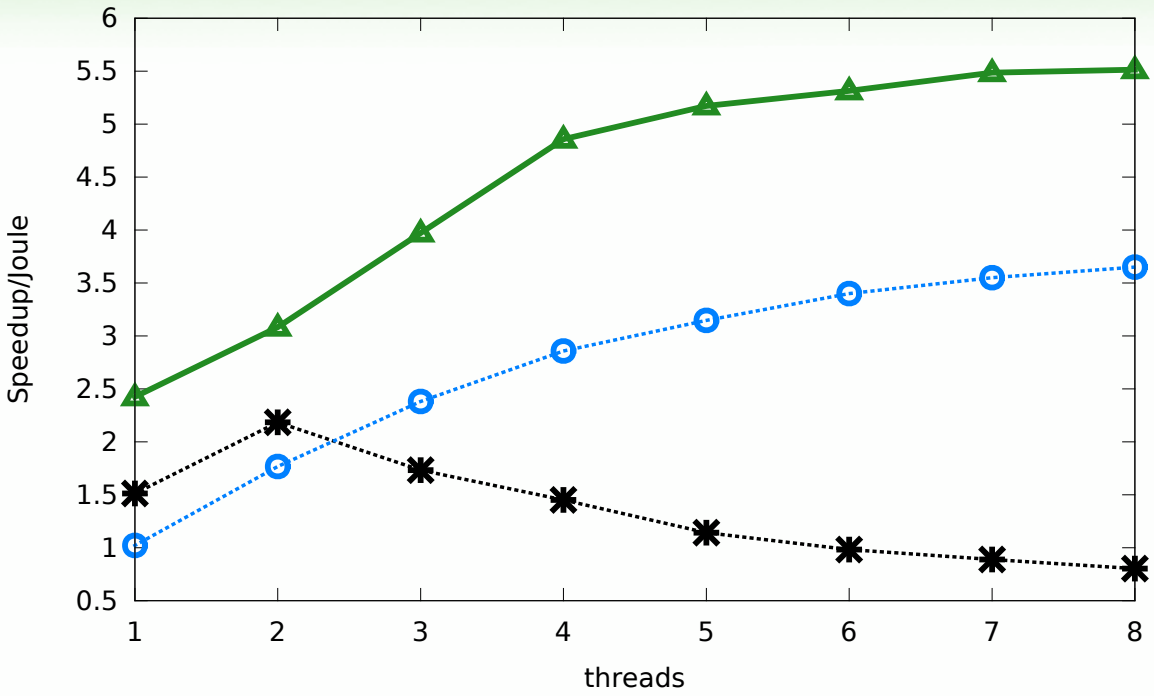
TinySTM

- Binding threads round-robin: $>$ 4t uses hyper-threading
- **TSX**-based approaches suffer from pressure on caches
- Best STMs (not **TL2**) scale regardless

STAMP results - **medium intensity of transactions**

		Most Performant	Least Power Consumption
L	kmeans	TSX-GL	TSX-GL
L	ssca2	TSX-GL	TSX-GL
M	intruder	TSX-GL $\leq$ 4t; TinySTM $\geq$ 5t	TSX-GL $\leq$ 5t; TinySTM $\geq$ 6t
M	vacation	TSX-GL $\leq$ 2t; TinySTM $\geq$ 3t	TSX-GL $\leq$ 4t; TinySTM $\geq$ 5t
H	genome		
H	yada		
H	labyrinth		

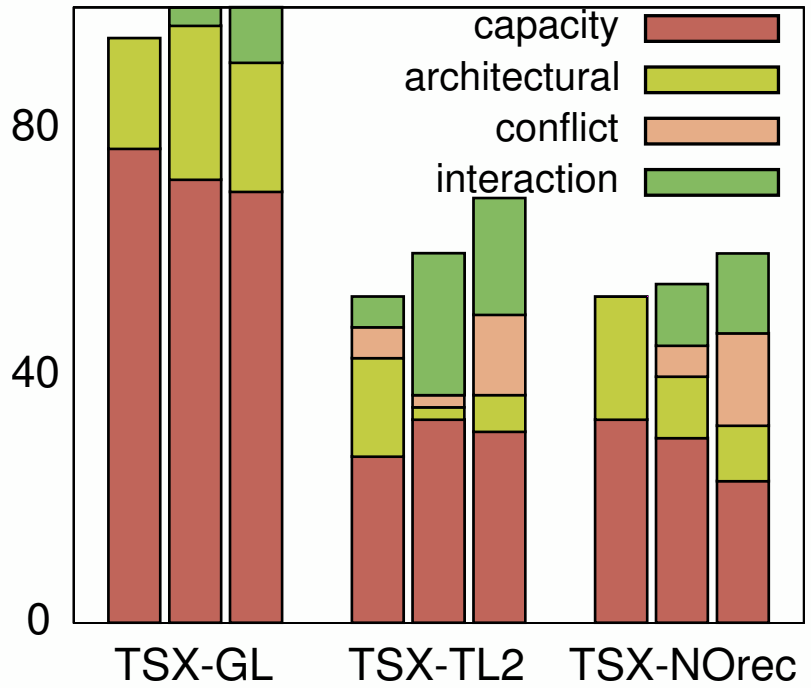
yada - high intensity



- **TSX-GL** does not scale
- HyTMs follow the trend of the STM counter-part
- When time to complete stagnates, power consumption stagnates
  - Logical cores of hyper-threading
  - Allow for additional hardware parallelism
  - Do not consume as much additional power

TSX-GL
TSX-NOrec
TinySTM

STAMP results - **high intensity of transactions**



Most conflicts are not due to data accesses

STAMP results

		Most Performant	Least Power Consumption
L	kmeans	TSX-GL	TSX-GL
L	ssca2	TSX-GL	TSX-GL
M	intruder	TSX-GL $\leq$ 4t; TinySTM $\geq$ 5t	TSX-GL $\leq$ 5t; TinySTM $\geq$ 6t
M	vacation	TSX-GL $\leq$ 2t; TinySTM $\geq$ 3t	TSX-GL $\leq$ 4t; TinySTM $\geq$ 5t
H	genome	TinySTM	TinySTM
H	yada	SwissTM	TinySTM
H	labyrinth	STMs (except TL2)	STMs (except TL2)

STAMP - fine-grained locking

- Requires a *per-application* effort
- Reasoning with transactions is meant to simplify programming

STAMP - fine-grained locking

- Requires a *per-application* effort
- Reasoning with transactions is meant to simplify programming
- Does not change the landscape of performance and power consumption

STAMP - fine-grained locking

- Requires a *per-application* effort
- Reasoning with transactions is meant to simplify programming

- Does not change the landscape of performance and power consumption
  - Additional lock acquisitions are noticeable in **L** workloads
  - An efficient fine-grained lock scheme was not found
  - TinySTM was competitive in **H** workloads

Lessons Learnt

- TSX is only worth in workloads with low intensity in transaction.

Lessons Learnt

- TSX is only worth in workloads with low intensity in transaction.
- STMs are the all-around champion, consistently performing good.

Lessons Learnt

- TSX is only worth in workloads with low intensity in transaction.
- STMs are the all-around champion, consistently performing good.
- The choice of fallback, when and how to do so, can impact performance and power consumption in 75%.

Lessons Learnt

- TSX is only worth in workloads with low intensity in transaction.
- STMs are the all-around champion, consistently performing good.
- The choice of fallback, when and how to do so, can impact performance and power consumption in 75%.
- Existing HyTMs do not justify the complexity to use them.

Thank You

Questions?