# A Study on the Parallelization of Terrain-Covering Ant Robots Simulations


SAPIENZA
UNIVERSITÀ DI ROMA

Alessandro Pellegrini
Francesco Quaglia

High Performance and Dependable Computing Systems Group
Sapienza, University of Rome

PADABS 2013

# Motivations

- Agent-based models reliably express interactions between different objects/entities in real world phenomena
- In some application domains, simulation timeliness is critical
  - Time-critical decision via what-if analysis (e.g., agent-based models in disaster-rescue contexts)

- Agent-based simulations are useful to study steady state or equilibrium properties of a system
- What if models are used to determine the exact simulated-time when a given global predicate becomes true?

# Motivations (2)

- Traditional sequential simulation is perfect for fine grain inspection of predicates ☺

# Motivations (2)

- Traditional sequential simulation is perfect for fine grain inspection of predicates ☺
- . . . yet it doesn't scale! 😐

# Motivations (2)

- Traditional sequential simulation is perfect for fine grain inspection of predicates ☺
- . . . yet it doesn't scale! 😐
- . . . and models are larger everyday!! ☹

# Motivations (2)

- Traditional sequential simulation is perfect for fine grain inspection of predicates ☺
- . . . yet it doesn't scale! 😐
- . . . and models are larger everyday!! ☹

- Parallel/Distributed simulation provides great performance ☺
- Fine grain inspection is not viable ☹
  - Process coordination is required
  - This hampers the achievable speedup

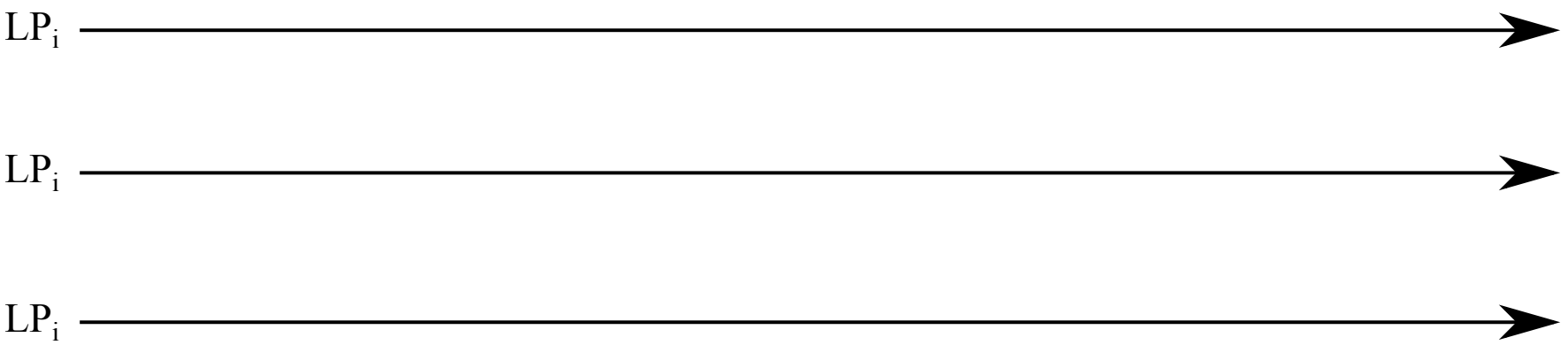
# Motivations (2)

- Traditional sequential simulation is perfect for fine grain inspection of predicates ☺
- . . . yet it doesn't scale! 😐
- . . . and models are larger everyday!! ☹

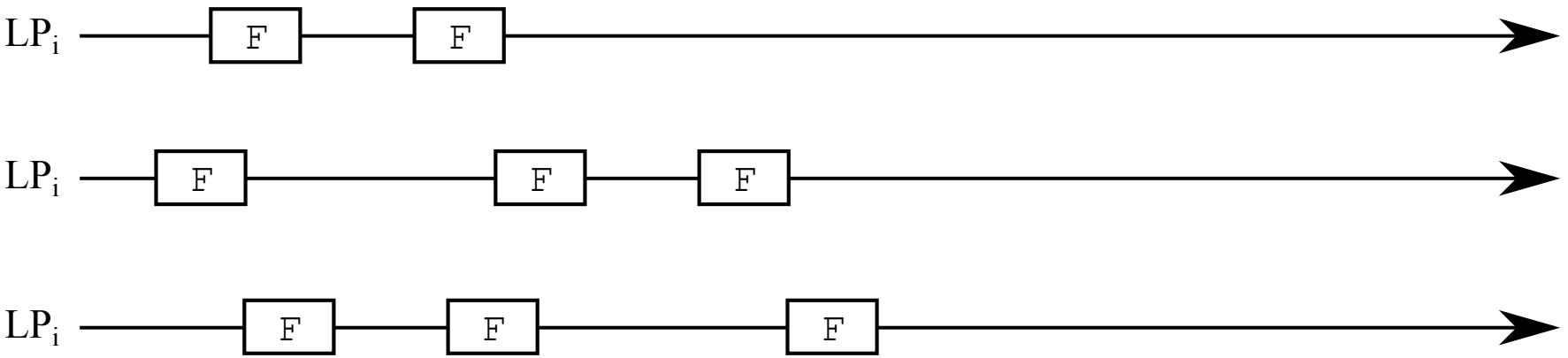
- Parallel/Distributed simulation provides great performance ☺
- Fine grain inspection is not viable ☹
  - Process coordination is required
  - This hampers the achievable speedup

- Speculative (optimistic) simulation inserts an additional delay
  - Inspection is delayed until a portion of the computation becomes committed

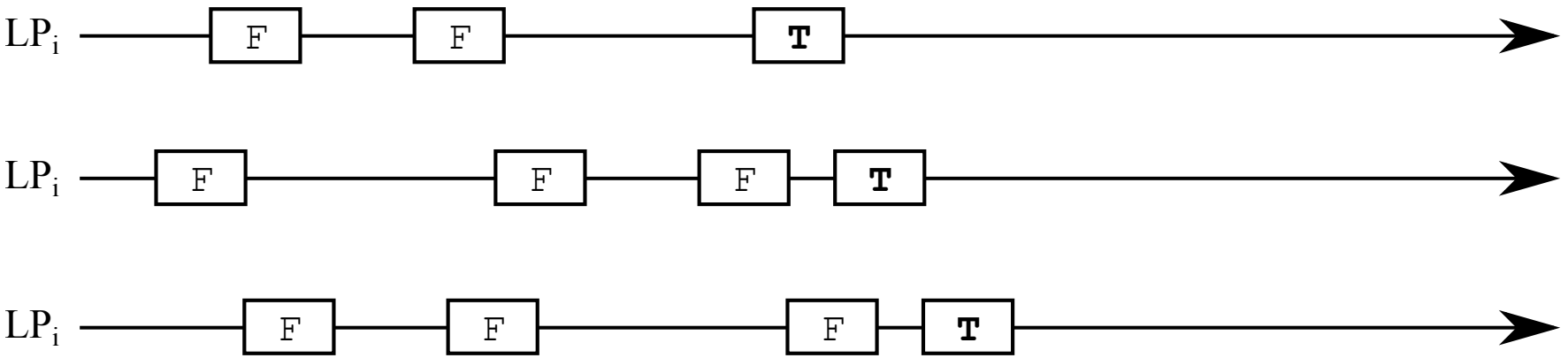
# The Completion-Shift Problem



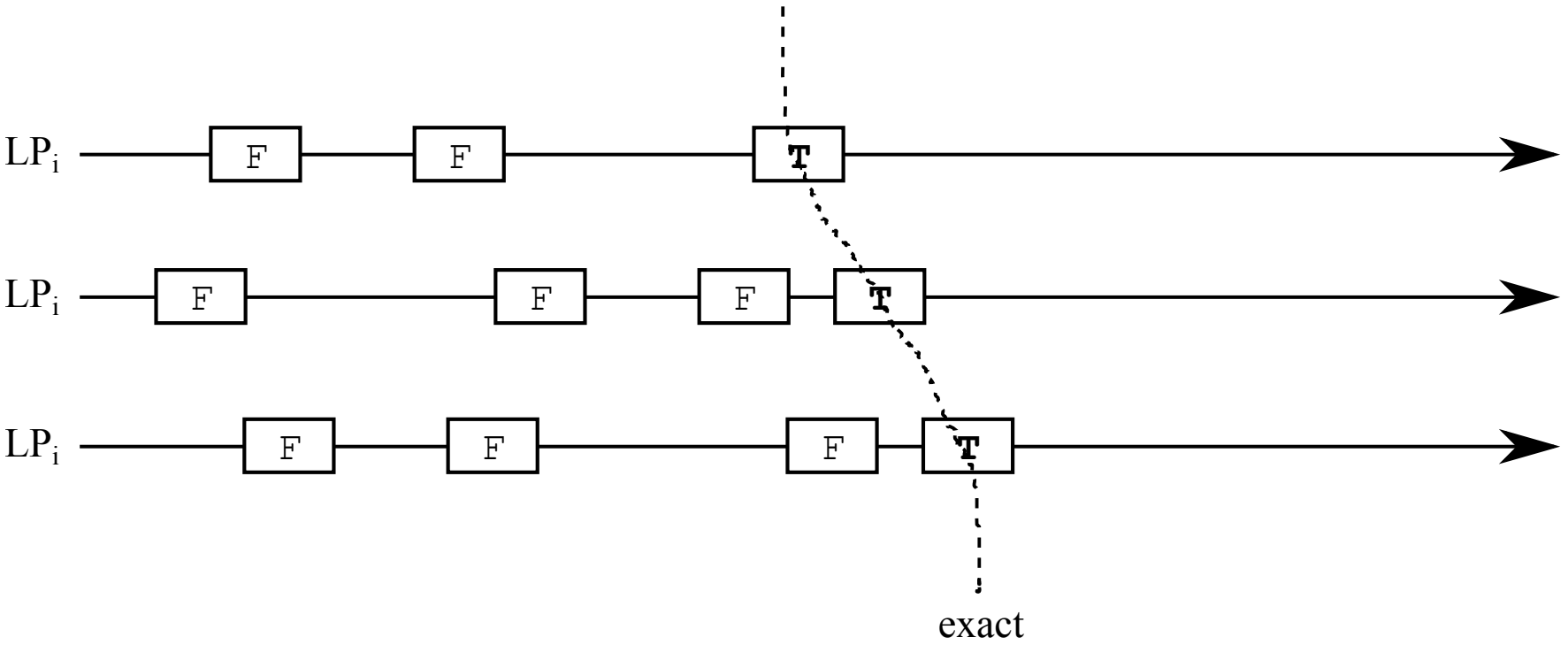
# The Completion-Shift Problem



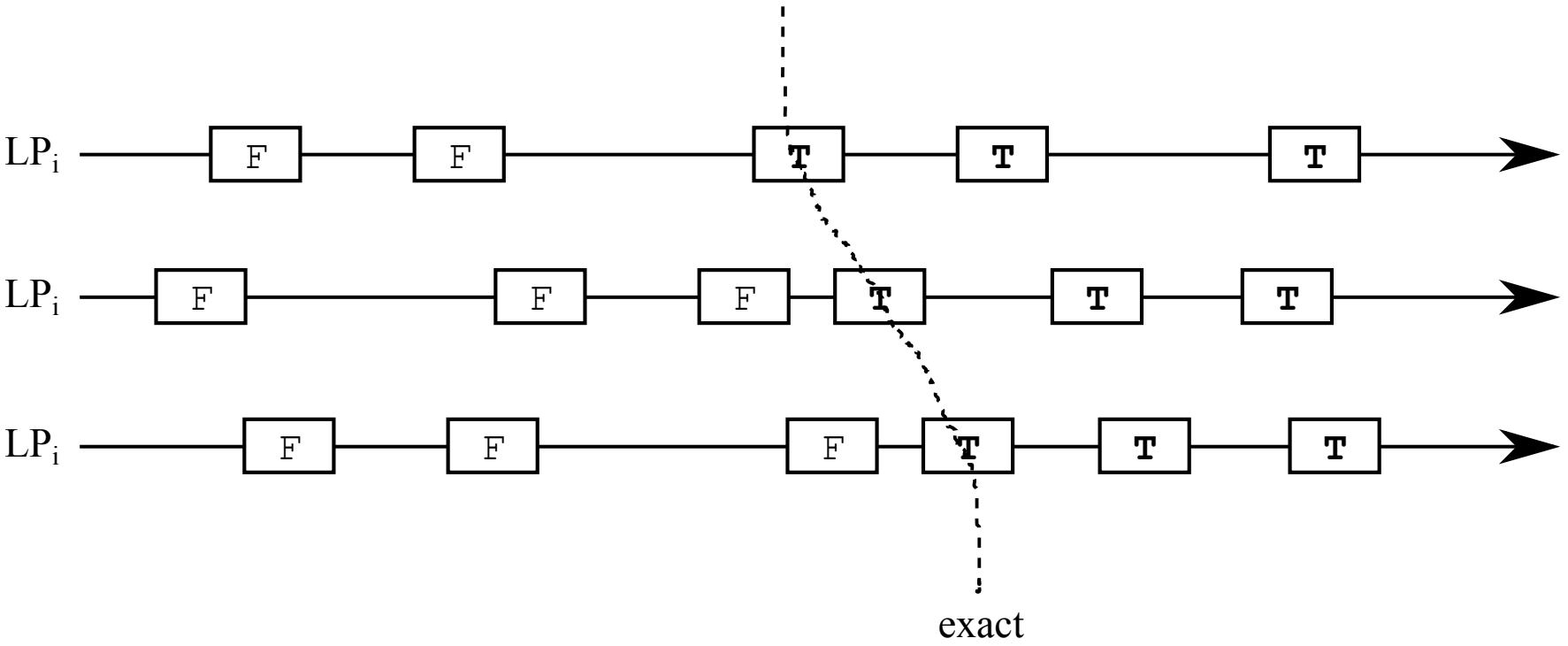
# The Completion-Shift Problem



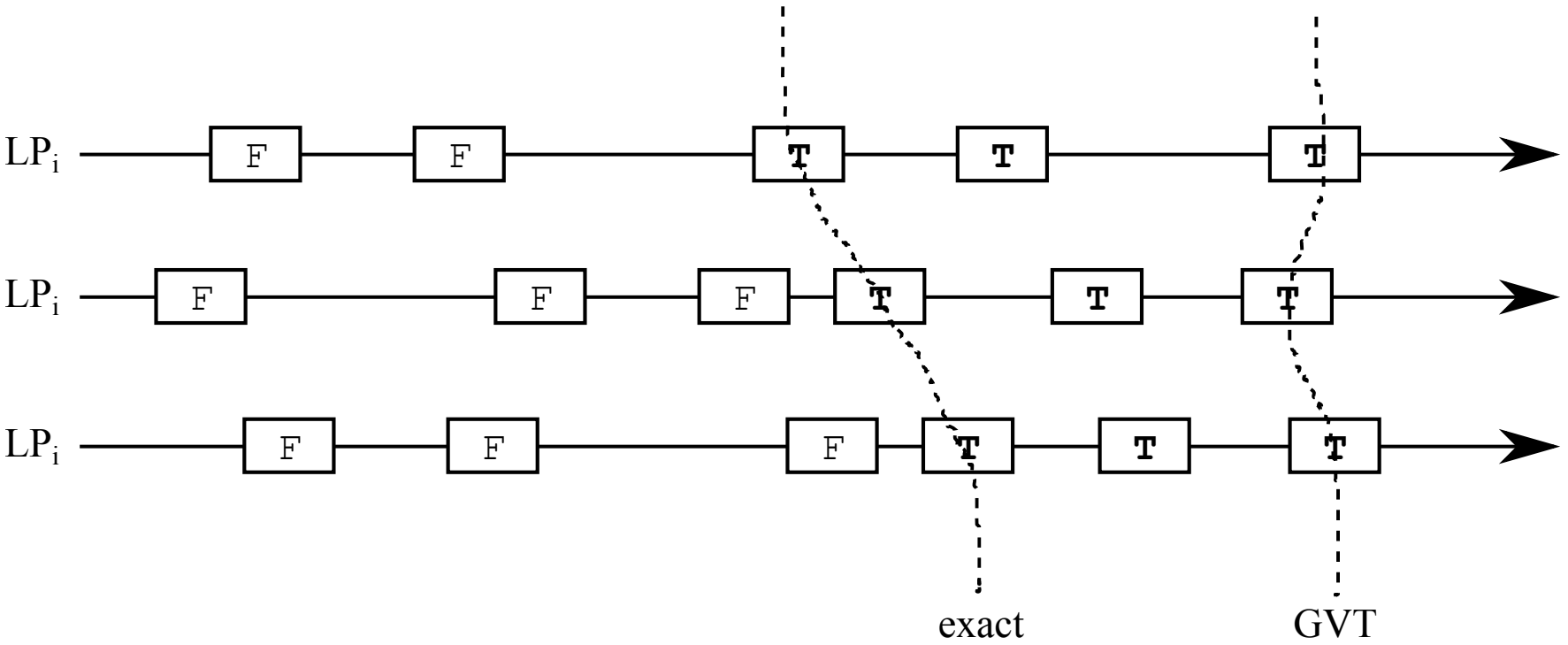
# The Completion-Shift Problem



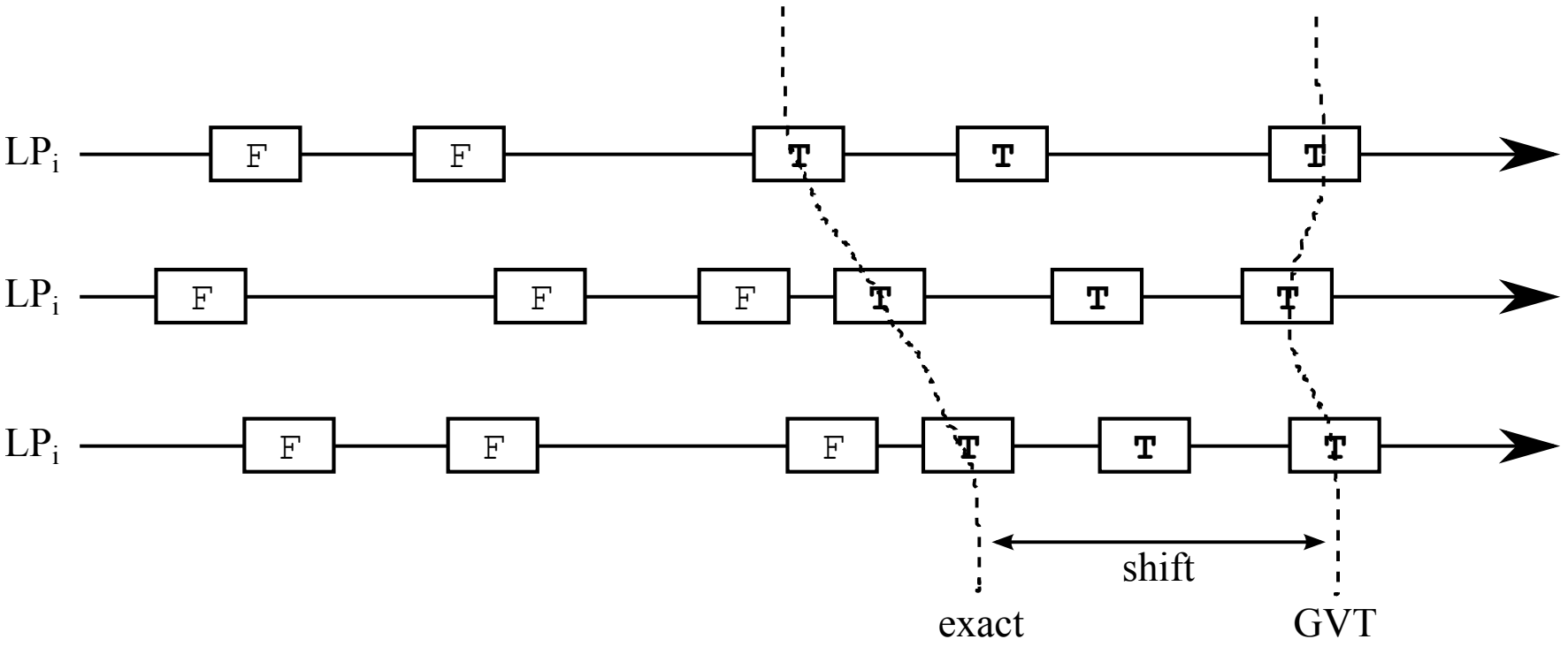
# The Completion-Shift Problem



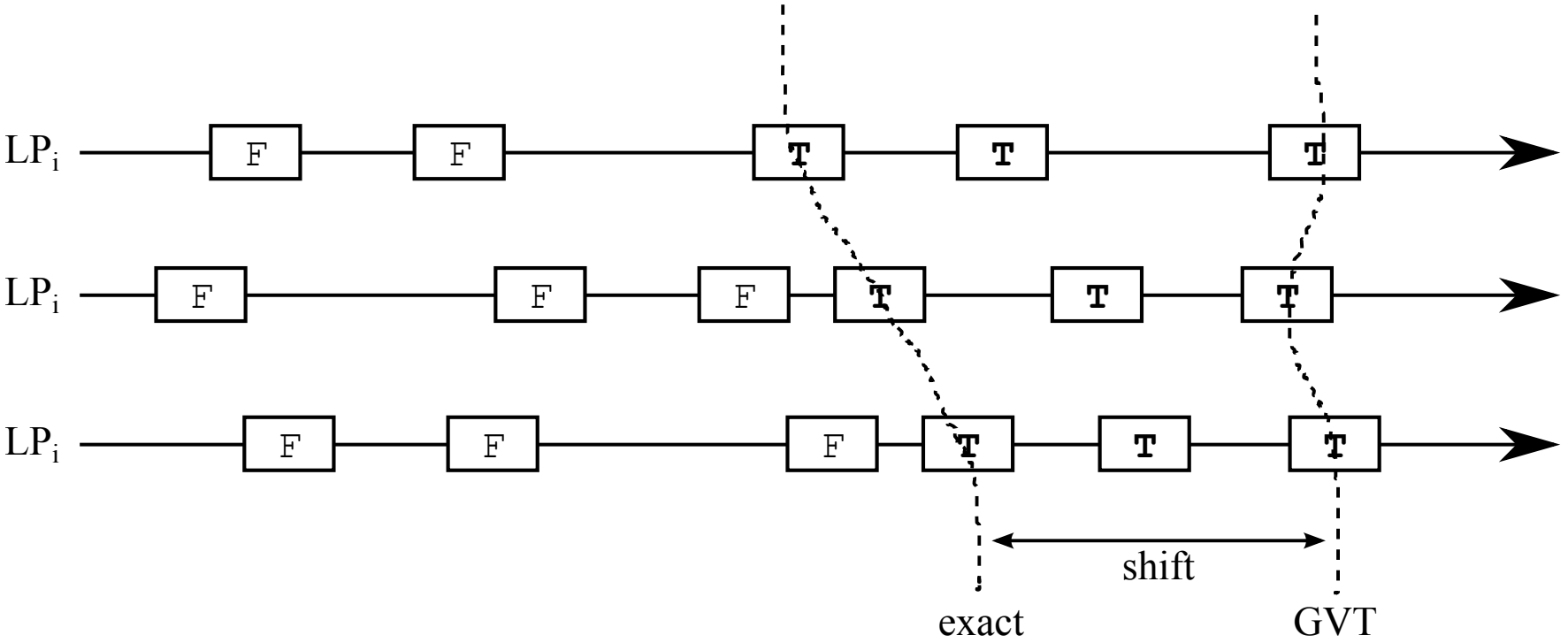
# The Completion-Shift Problem



# The Completion-Shift Problem



# The Completion-Shift Problem




How does this shift affects the results?

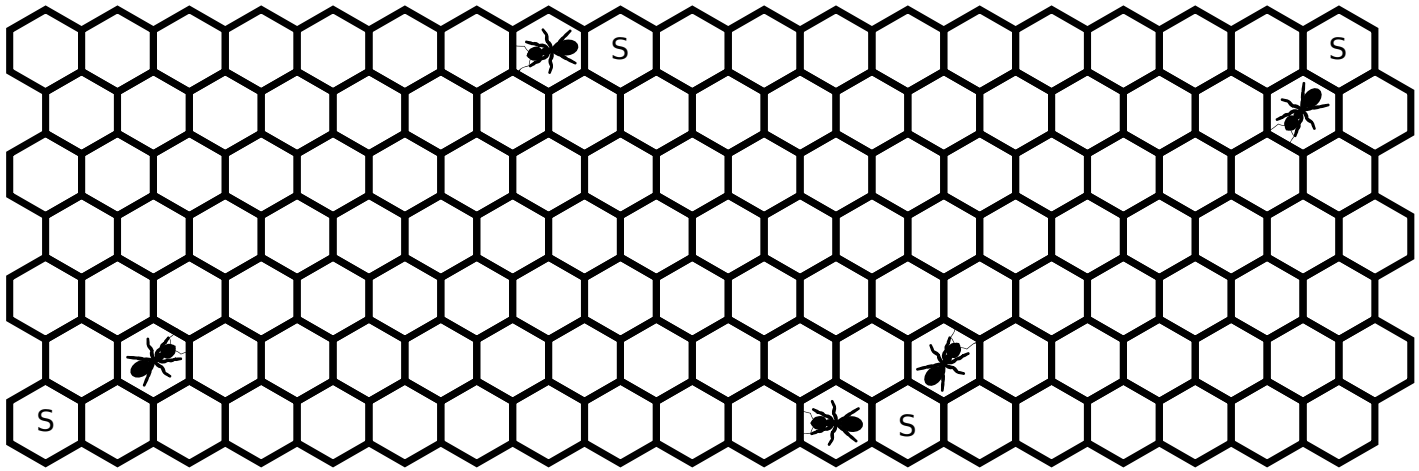
What is the tradeoff between performance and results' reliability?

# Simulation Model: Terrain-Covering Ant Robots

- Original model by Koenig–Liu [KL01], we propose a parallel version
- Interesting model for the assessment of rescue scenarios
- The terrain is modeled as an undirected graph
- Space is divided into hexagonal cells
- An ant robot can move to adjacent cells, accounting for its speed (50 cm/s at most)

# Simulation Model: Terrain-Covering Ant Robots

- Original model by Koenig–Liu [KL01], we propose a parallel version
- Interesting model for the assessment of rescue scenarios
- The terrain is modeled as an undirected graph
- Space is divided into hexagonal cells
- An ant robot can move to adjacent cells, accounting for its speed (50 cm/s at most)



# Simulation Model: Terrain-Covering Ant Robots (2)

- Ant robots leave *pheromones* when passing through a cell
- Pheromones notify other robots of their visit to a cell
- A node-counting algorithm allows to select the least-visited node when choosing direction

- Model's events are:
  - `REGION_IN`: an ant robot enters a given cell, trail counter is increased
  - `UPDATE_NEIGHBORS`: adjacent LPs are notified of new trail counter value
  - `REGION_OUT`: an ant robot is leaving a given cell

# Simulation Model's Configuration

- Square region, 12 $Km^2$
- 4900 hexagonal cells (each on a LP) on 32 ROOT-Sim kernel instances
- 4 source points (at region corners)
- Variable number of robots $[4, 32]$ per source point
- Variable GVT computation interval $[1, 5]$ seconds
- Simulation is run until full region coverage, visit factor of 20
- Comparison with a sequential run relying on a Calendar Queue

- 32-core HP ProLiant server, NUMA architecture
- 64 Gb RAM
- Linux Kernel 2.6.32-5-amd64 – Debian 6

# Reference Simulation Platform: ROOT-Sim


- Parallel simulation has been run on top of the ROme OpTimistic Simulator (ROOT-Sim)
- An Optimistic (Distributed) Parallel Discrete Simulation Platform
- Supports ANSI-C for simulation models' development
- Transparently supports rollback via log/restore facilities

# Experimental Results: Simulation Completion Time

- Results averaged over 20 runs with different initial random seeds
- The simulation completion time allows to assess the shift problem
- Results represent simulated hours!

| Configuration | | Sequential | GVT=1 | GVT=2 | GVT=3 | GVT=4 | GVT=5 |
|---|---|---|---|---|---|---|---|
| **16 Robots** | Mean | 211.86 | 216.31 | 218.27 | 218.69 | 234.99 | 221.81 |
| | Std. Dev. | 1,56 | 15.11 | 13.28 | 11.07 | 12.46 | 15.64 |
| **128 Robots** | Mean | 26.56 | 27.37 | 28.41 | 28.29 | 32.61 | 29.24 |
| | Std. Dev. | 0.16 | 1,08 | 1,37 | 3,25 | 1.83 | 1.01 |

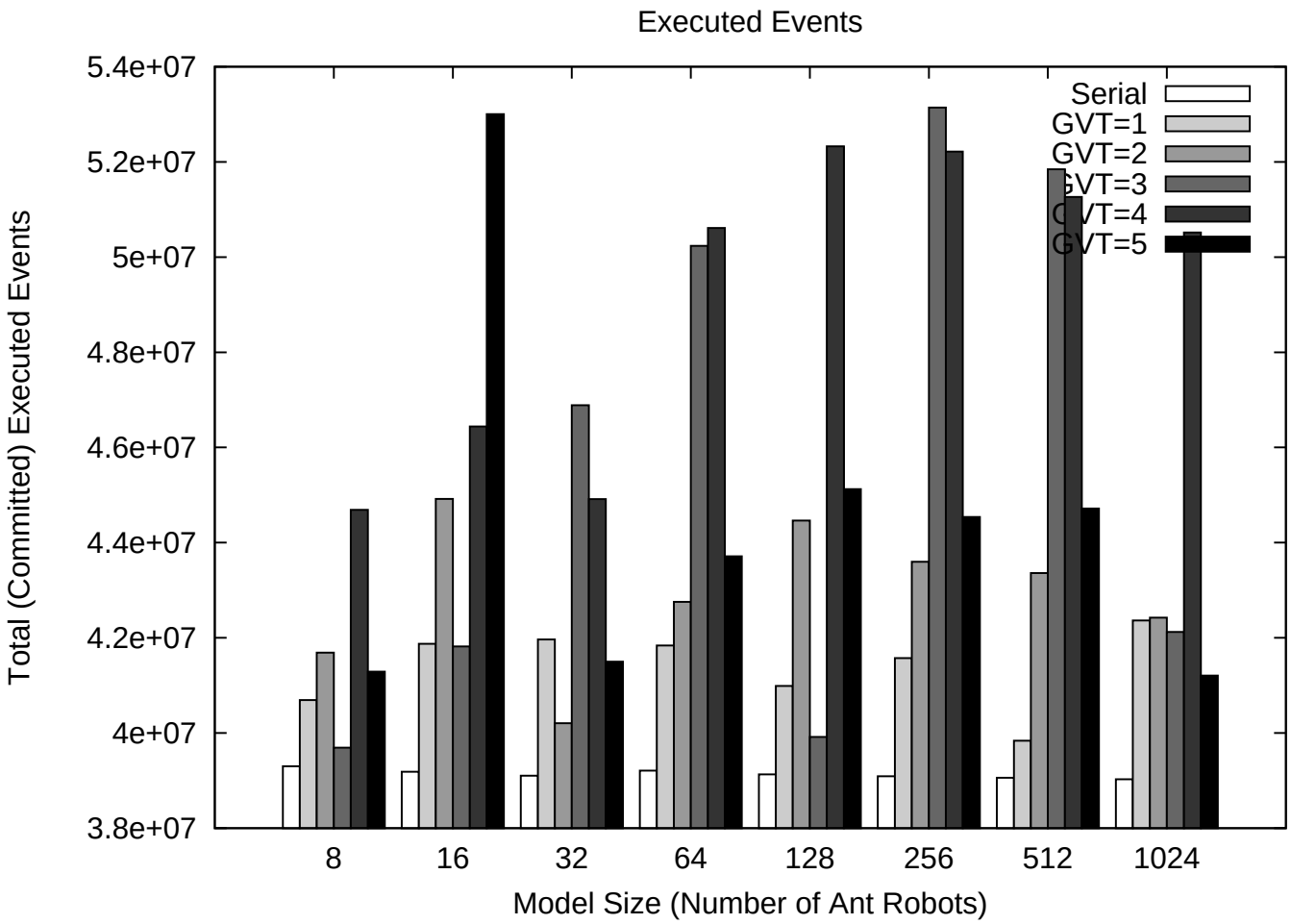
# Experimental Results: Simulation Completion Time

- Results averaged over 20 runs with different initial random seeds
- The simulation completion time allows to assess the shift problem
- Results represent simulated hours!

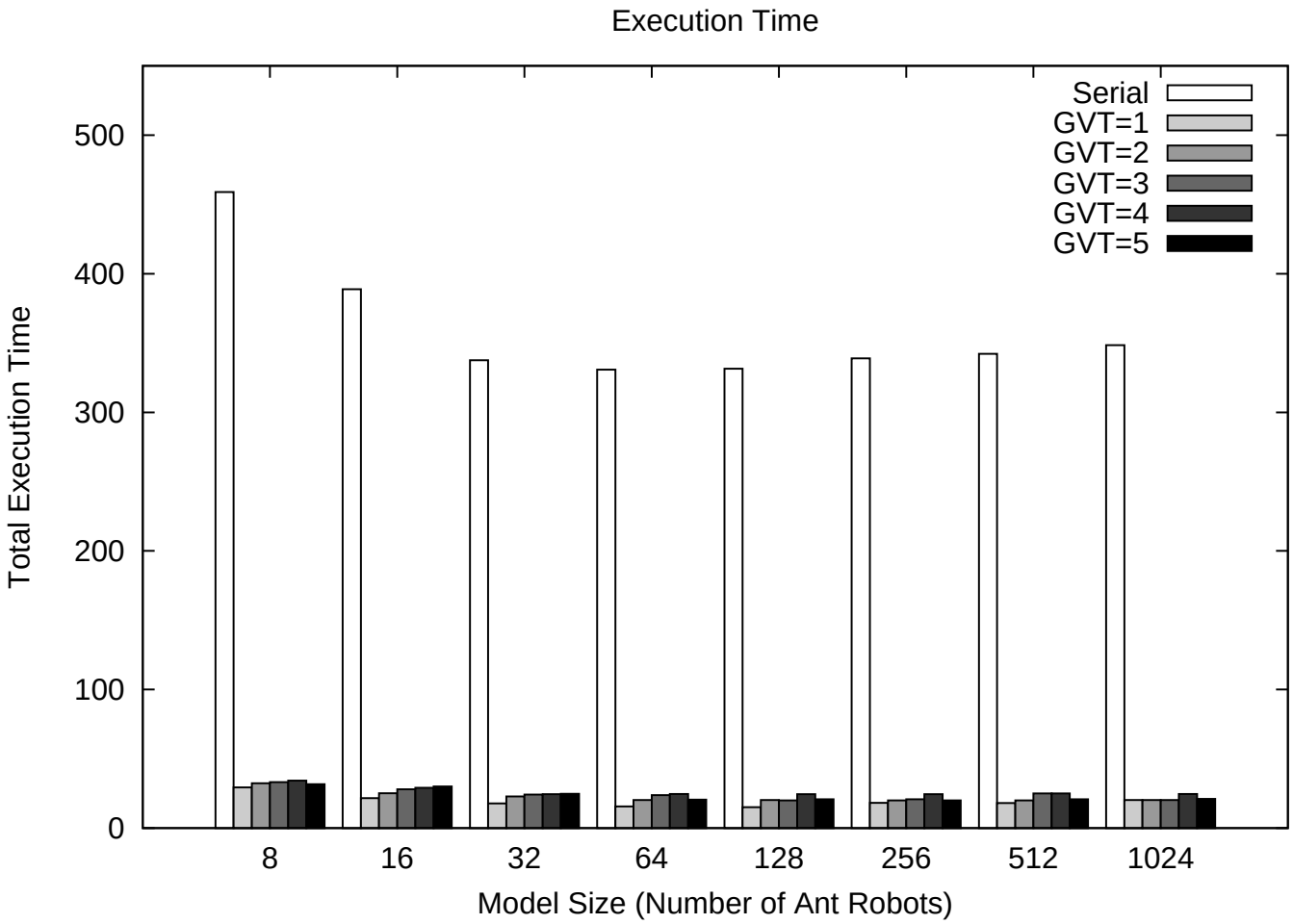
| Configuration | | Sequential | GVT=1 | GVT=2 | GVT=3 | GVT=4 | GVT=5 |
|---|---|---|---|---|---|---|---|
| **16 Robots** | Mean | 211.86 | 216.31 | 218.27 | 218.69 | 234.99 | 221.81 |
| | Std. Dev. | 1,56 | 15.11 | 13.28 | 11.07 | 12.46 | 15.64 |
| **128 Robots** | Mean | 26.56 | 27.37 | 28.41 | 28.29 | 32.61 | 29.24 |
| | Std. Dev. | 0.16 | 1,08 | 1,37 | 3,25 | 1.83 | 1.01 |

- Optimistic Simulation results are just upper bounds!

# Experimental Results: Executed Events



# Experimental Results: Execution Time



# Thanks for your attention

Questions?

pellegrini@dis.uniroma1.it
http://www.dis.uniroma1.it/∼pellegrini
http://www.dis.uniroma1.it/∼ROOT-Sim
http://www.dis.uniroma1.it/∼ROOT-Sim/tcar.tbz