

# PVESTA: A Parallel Statistical Model Checking and Quantitative Analysis Tool

Musab AlTurki   José Meseguer

University of Illinois at Urbana-Champaign

August 31, 2011

# Statistical Model Checking

- ▶ Applying **statistical** methods to **formal** models of probabilistic systems
- ▶ **Attractive** formal analysis method for probabilistic systems
  - ▶ Scalability
  - ▶ Acceptable approximations within user-specified bounds
- ▶ Very useful for analysis of **complex** systems, e.g. distributed cyber-physical systems
  - ▶ Probabilistic algorithms
  - ▶ Stochastic nature of the environment
  - ▶ Quantitative vs. Boolean properties

# Main Contributions

- ▶ Drastically increase the scalability of statistical model checking through parallelization
- ▶ Make such scalability of analysis available to tools such as Maude
  - ▶ Probabilistic rewrite theories
- ▶ An implementation in an extension of VESTA, called **PVESTA**
- ▶ Maintaining an up-to-date version of of the VESTA tool

# Statistical Model Checking Algorithms

- ▶ Statistical model checking of **PCTL/CSL** formulas [Sen, Viswanathan, Agha 2005]
  - ▶ Statistical hypothesis testing based on  $n$  random samples
- ▶ Statistical quantitative analysis of **QUATEX** expressions [Agha, Meseguer, Sen 2005]
  - ▶ Confidence interval estimation based on  $n$  random samples
- ▶ Both implemented in Java in **VESTA**

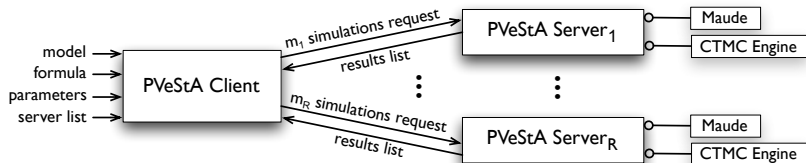
# Statistical Model Checking Algorithms

- ▶ Both based on multiple random samples, computed from **independent** Monte Carlo simulations
- ▶ Number of samples must be **large enough** to guarantee the desired statistical strength of the result
- ▶ Simulations tend to **dominate** other computations as systems get more complex
- ▶ **Significant** speedups and improved scalability by parallelization

# Parallel Statistical Model Checking

- ▶ Introduce **parallel**, map-reduce versions of both algorithms
- ▶ Samples computed in parallel by performing simulations in parallel
- ▶ Simulations task mapped onto a list of computing resources given as input
- ▶ Implemented in **PVESTA**, based on VESTA's implementation
  - ▶ CTMCs and probabilistic rewrite theories
  - ▶ PCTL/CSL and QUATEX

# Architecture of PVEStA

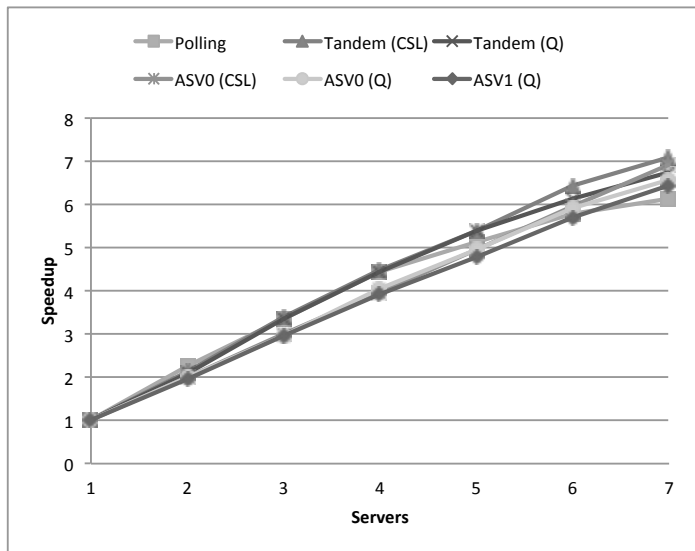


# Experimental Evaluation

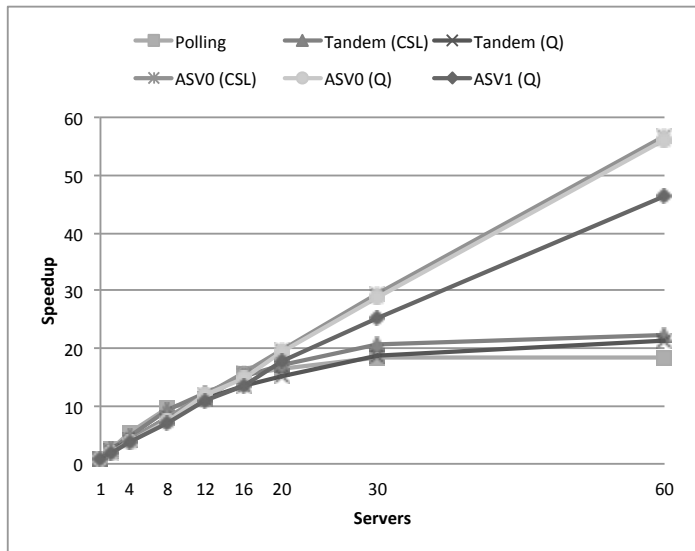
- ▶ Conducted two sets of experiments to evaluate parallelization performance gains:
  - ▶ **HPC** architecture: PC cluster of 256 nodes, AMD Opteron 2.2GHz CPUs with 2GB of RAM
  - ▶ **Multi-core** server: PC with dual quad-core 2.66GHz Intel Xeon processors with 16GB of RAM
- ▶ Four models: *Polling*, *Tandem*,  $ASV_0$ ,  $ASV_1$
- ▶ Example formulas in CSL and QUATEX



# PVESTA on a Multi-core Machine



# PVESTA on a HPC Cluster



# PVeSTA Web Site

PVeSTA: A Parallel Statistical Model Checking and Quantitative Analysis Tool

## PVeSTA

[Home](#) [Download](#) [Usage](#) [Feedback](#)

### A Parallel Statistical Model Checking and Quantitative Analysis Tool

---

#### An Overview of PVeSTA

PVeSTA is a client-server-based parallelization of the [VeSTA](#) tool, originally developed by [Koushik Sen](#), for statistical model checking and quantitative analysis of probabilistic systems. PVeSTA implements parallelized versions of both [Sen et. al.](#)'s statistical model checking algorithm and [Agha et. al.](#)'s statistical quantitative analysis algorithm, in which random samples are computed in parallel by distributing, as evenly as possible, the task of performing Monte Carlo simulations across different computing elements.

As in the original VeSTA, PVeSTA supports statistical verification of properties expressed in Probabilistic Computational Tree Logic (PCTL), Continuous Stochastic Logic (CSL) or the Quantitative Temporal Expressions language (QuaTE<sub>x</sub>) against probabilistic real-time models specified as probabilistic rewrite theories in [Maude](#) or continuous- or discrete-time Markov Chains.

---

#### About PVeSTA's Implementation

PVeSTA is implemented in Java 1.6, based on the original implementation of VeSTA. It consists of two command-line-based executable programs: (1) a client program, which essentially implements the sequential parts of the

<http://musabalturki.net/pvesta>

# Conclusion and Future Work

- ▶ **Parallelizations** of statistical model checking and quantitative analysis algorithms
- ▶ A client-server **implementation**, and experimental demonstration of increased scalability
- ▶ Further **refinements and extensions**
  - ▶ Improve performance and efficiency
  - ▶ Extend parallel statistical model checking to nested probabilistic formulas

Thank you!