Statistical Model Checking of Simulink Models

Edmund M. Clarke
School of Computer Science
Carnegie Mellon University

```cpp
++CDatabase::_stats.mem_used_u
_params.max_unrelevance = (int
  if (_params.max_unrelevance <
    _params.max_unrelevance =
    _params.min_num_clause_lits_fd
    if (_params.min_num_clause_lit
      _params.lit_num_clause_lit
    params.max_num_clause_lit
    params.max_numClause.clause_le
  if (_params.max_numClause.clause
    params.max_numClause.clause
CHECK(
  cout << "Forced to reduce unre
  cout << "MaxUnrel: " << _params
  " MinLenDel: " << _pa
  " MaxLenCL : " << _pa
);
The State Explosion Problem

My 27 Year Quest:

- Symmetry Reduction
- Parametric Model Checking
- Partial Order Reduction
- Symbolic Model Checking
- Induction in Model Checking
- SAT based Bounded Model Checking
- Predicate Abstraction
- Counterexample Guided Abstraction Refinement
- Compositional Reasoning

... 

- **Statistical Model Checking!**
Executive Summary

- **State Space Exploration** is infeasible for large systems.
  - Often easier to simulate a system
- Our Goal: Provide **probabilistic guarantees of correctness** using a small number of simulations
  - How to generate each simulation run?
  - How many simulation runs to generate?
- Applications: Stateflow / Simulink, Biological Models.

**Statistical Model Checking of Mixed-Analog Circuits with an Application to a Third Order Delta - Sigma Modulator.**

E. M. Clarke, A. Donzé, and A. Legay. **Best Paper Award** at Haifa Verification Conference 2008.

Bayesian Statistical Model Checking

- **Bayesian Approach** to Statistical Model Checking
  - Faster than state-of-the-art Statistical Model Checking.
  - Generally requires fewer simulations.

- Can use **prior knowledge** about the model
  - Represented by the *prior probability* distribution of the model satisfying the specification.

- Can revise **prior knowledge** in light of experimental data
  - Compute *posterior probability* of the model satisfying the specification.

Bayesian Statistical Model Checking
Motivation - Scalability

- **State Space Exploration** infeasible for large systems.
  - Symbolic MC with OBDDs scales to $10^{300}$ states.
  - Scalability depends on the structure of the system.
- **Simulation** is feasible for many more systems.
- Target Applications include:
  - Stateflow Simulink Models
  - Analog Circuits
  - Verilog Models
  - Biological Models
Motivation – Parallel Model Checking

- Some success with explicit state Model Checking
  - Parallel Murphi
- More difficult to distribute Symbolic MC using BDDs.
- Learned Clauses in SAT solving are not easy to distribute for Bounded Model Checking.
- Simulation can be easily parallelized.
- Statistical Model Checking should be able to exploit
  - multiple cores
  - commodity clusters
Probabilistic Model Checking

- Given a stochastic model $\mathcal{M}$ such as
  - a Markov Chain, or
  - the solution to a stochastic differential equation
- a Bounded Linear Temporal Logic property $\phi$ and a probability threshold $\theta \in (0, 1)$.
- Does $\mathcal{M}$ satisfy $\phi$ with probability at least $\theta$?
  $$\mathcal{M} \models P_{\geq \theta}(\phi)$$
- Example: Is every request acknowledged within 10 time units with 99.999999% probability?
- Numerical techniques compute the precise probability of $\mathcal{M}$ satisfying $\phi$:
  - Does NOT scale to large systems.
Statistical Probabilistic Model Checking

- Decides between two mutually exclusive composite hypotheses:
  - Null Hypothesis \( H_0 : \mathcal{M} \models P_{\geq \theta}(\phi) \)
  - Alternate Hypothesis \( H_1 : \mathcal{M} \models P_{< \theta}(\phi) \)

- Statistical tests can determine the true hypothesis:
  - based on sampling the traces of system \( \mathcal{M} \)
  - answer may be wrong, but error probability is bounded.

- Statistical Hypothesis Testing \( \rightarrow \) Model Checking!

Carnegie Mellon
Challenges

- Each simulation trace is expensive to generate
  - Computation time: few minutes to several days.

- Given an upper bound on the probability of making incorrect decisions:
  - Sample as many traces as needed, but no more.

- Nondeterministic Systems:
  - Nondeterminism due to incompletely specified inputs
  - Model Checking Markov Decision Processes (PRISM)
  - Statistical Model Checking not yet adapted to MDPs
Existing Work

- [Younes and Simmons 06] use Wald’s SPRT
  - SPRT: Sequential Probability Ratio Test

- The SPRT decides between
  - the simple null hypothesis $H_0' : \mathcal{M} \models P_{=\theta_0}(\phi)$
  vs
  - the simple alternate hypothesis $H_1' : \mathcal{M} \models P_{=\theta_1}(\phi)$

- SPRT is asymptotically optimal (when $H_0'$ or $H_1'$ is true)
  - Minimizes the expected number of samples
  - Among all tests with equal or smaller error probability.
Existing Work

- MC chooses between two composite hypotheses
  \[ H_1 : \mathcal{M} \models P_{<\theta}(\phi) \quad H_0 : \mathcal{M} \models P_{\geq \theta}(\phi) \]
- Existing works use SPRT for hypothesis testing with an indifference region:
  \[ \mathcal{M} \models P_{=\theta-\delta}(\phi) \quad \mathcal{M} \models P_{=\theta+\delta}(\phi) \]
But MC chooses between two mutually exclusive composite hypotheses

Null Hypothesis $H_0 : \mathcal{M} \models P_{\geq \theta}(\phi)$

vs

Alternate Hypothesis $H_1 : \mathcal{M} \models P_{< \theta}(\phi)$

We have developed a new MC algorithm
- Statistical Model Checking Algorithm
- Performs Composite Hypothesis Testing
- Based on Bayes Theorem and the Bayes Factor.
Faster Statistical Model Checking II

- Model Checking: $H_0 : \mathcal{M} \models P_{\geq \theta}(\phi)$
- Suppose $\mathcal{M}$ satisfies $\phi$ with (unknown) probability $u$.
  - $u$ is given by a random variable $U$ with density $g$.
  - $g$ represents the prior belief that $\mathcal{M}$ satisfies $\phi$.
- Generate independent and identically distributed (iid) sample traces.
- $x_i$: the $i^{th}$ sample trace $\sigma$ satisfies $\phi$.
  - $x_i = 1$ iff $\sigma_i \models \phi$
  - $x_i = 0$ iff $\sigma_i \not\models \phi$
- Then, $x_i$ will be a Bernoulli trial with density
  \[ f(x_i|u) = u^x_i(1 - u)^{1-x_i} \]
- $X = (x_1, \ldots, x_n)$ a sample of Bernoulli random variables.
- Bayes Theorem (Posterior Probability):
  \[
P(H_0 \mid X) = \frac{P(X \mid H_0)P(H_0)}{P(X)}
  \]
- Prior Probability of $H_0$ being true:
  \[
P(H_0) = \int_\theta^1 g(u)du
  \]
- Ratio of Posterior Probabilities:
  \[
  \frac{P(H_0 \mid X)}{P(H_1 \mid X)} = \frac{P(X \mid H_0)}{P(X \mid H_1)} \frac{P(H_0)}{P(H_1)}
  \]
  Bayes Factor
Faster Statistical Model Checking IV

- **Bayes Factor**: Measure of confidence in $H_0$ vs $H_1$
  - provided by the data $X = (x_1, \ldots, x_n)$
  - weighted by the prior $g$.

- **Bayes Factor** \#Threshold: **Accept** Null Hypothesis $H_0$.
- **Bayes Factor** \? \#Threshold: **Reject** Null Hypothesis $H_0$.

**Definition**: Bayes Factor $\mathcal{B}$ of sample $X$ and hypotheses $H_0$, $H_1$

$$\mathcal{B} = \frac{P(X \mid H_0)}{P(X \mid H_1)} = \frac{\int_{\theta}^{1} f(x_1 \mid u) \cdots f(x_n \mid u) \cdot g(u) \, du}{\int_{0}^{\theta} f(x_1 \mid u) \cdots f(x_n \mid u) \cdot g(u) \, du}$$
**Faster Statistical Model Checking V**

**Require:** Property $P_{\geq \theta}(\Phi)$, Threshold $T > 1$, Prior density $g$

$n := 0$ \hspace{1cm} \{number of traces drawn so far\}

$x := 0$ \hspace{1cm} \{number of traces satisfying so far\}

repeat

\[ \sigma := \text{draw a sample trace of the system (iid)} \]

\[ n := n + 1 \]

if $\sigma \models \Phi$ then

\[ x := x + 1 \]

end if

$B := \text{BayesFactor}(n, x)$

until $(B > T \lor B < 1/T)$

if $(B > T)$ then

return $H_0$ accepted

else

return $H_1$ accepted

end if
Bounded Linear Temporal Logic (BLTL): Extension of LTL with time bounds on temporal operators.

Let \( \sigma = (s_0, t_0), (s_1, t_1), \ldots \) be an execution of the model
- along states \( s_0, s_1, \ldots \)
- the system stays in state \( s_i \) for time \( t_i \)

\( \sigma^i \): Execution trace starting at state \( i \).

\( V(\sigma, i, x) \): Value of the variable \( x \) at the state \( s_i \) in.

A natural model for Simulink traces
- Simulink has discrete time semantics
Semantics of BLTL

The semantics of BLTL for a trace $\sigma^k$:

- $\sigma^k \models x \sim c$  iff  $V(\sigma, k, x) \sim c$, where $\sim$ is in $\{\leq, \geq, =\}$
- $\sigma^k \models \Phi_1 \lor \Phi_2$  iff  $\sigma^k \models \Phi_1$ or $\sigma^k \models \Phi_2$
- $\sigma^k \models \neg \Phi$  iff  $\sigma^k \models \Phi$ does not hold
- $\sigma^k \models \Phi_1 \mathcal{U}^t \Phi_2$  iff  there exists natural $i$ such that
  1) $\sigma^{k+i} \models \Phi_2$
  2) $\sum_{j<i} t_j \leq t$
  3) for each $0 \leq j < i$, $\sigma^{k+j} \models \Phi_1$
Fuel System Controller

The Simulink model:
Fuel System Controller

- We Model Check the formula (Null hypothesis)
  \( M, \text{FaultRate} \models P_{\geq \theta} (\neg F^{100} G^{1}(\text{FuelFlowRate} = 0)) \)
  for \( \theta = 0.5, 0.7, 0.8, 0.9, 0.99 \).
- “It is not the case that within 100 seconds, FuelFlowRate is zero for 1 second”.
- We use various values of FaultRate for each of the three sensors in the model.
- We use uniform priors over \([0,1)\); both hypotheses equally likely a priori.
- We choose Bayes threshold \( T @ \#1000 \), i.e., stop when one hypothesis is 1000 times more likely than the other.
Recall the Null hypothesis:

\[ \mathcal{M}, \text{FaultRate} \models P_{\geq \theta}(\neg F^{100} G^{1}(\text{FuelFlowRate} = 0)) \]

Number of samples and test decision:
- blue numbers: test accepted Null hypothesis.
- red numbers: test rejected Null hypothesis.

<table>
<thead>
<tr>
<th>Fault rates</th>
<th>Probability threshold ( \theta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>[3 7 8]</td>
<td>.5</td>
</tr>
<tr>
<td></td>
<td>63</td>
</tr>
<tr>
<td>[10 8 9]</td>
<td>29</td>
</tr>
<tr>
<td>[20 10 20]</td>
<td>9</td>
</tr>
<tr>
<td>[30 30 30]</td>
<td>9</td>
</tr>
</tbody>
</table>
Δ – Σ Modulators for Dummies

- Widely used family of Analog Digital Converters
- Efficient control of quantization error, i.e., the difference between the analog input and the digital output
- **Saturation** is a critical issue:
  - Internal state variable of the integrator may reach the maximum value.
  - The output does not respond linearly to the input.
  - Saturation compromises the quality of A-D conversion.
Simple Discrete-Time $\Delta - \Sigma$ Modulator

- **Quantization error** is the difference between the input and the output.
- **Integrator** stores the summation of $\delta$'s in a state variable $x$.
- **Quantizer** produces output based on the sign of $x$. 

\[
\begin{align*}
\delta(k) &= u(k) - v(k) \\
x(k) &= x(k-1) + \delta(k) \\
v(k) &= \text{sign}(x(k))
\end{align*}
\]
Higher Order $\Delta - \Sigma$ Modulators

- More complex designs use more than one integrator.
- The order of a $\Delta - \Sigma$ modulator is the number of integrators used.
- Integrator’s state variables can become saturated.
  - We study the property $P_{\geq \theta} F_{\text{Satur}}$.
  - “circuit eventually saturates with probability at least $\theta$”.
- We simulate the system using input signals sampled from a uniform distribution.

  ➡️ Statistical MC for inputs of bounded amplitude.
### Experimental Results

<table>
<thead>
<tr>
<th>Maximum Input Amplitude</th>
<th>Estimated Saturation Probability</th>
<th>Number of samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.15</td>
<td>0.0938</td>
<td>4967</td>
</tr>
<tr>
<td>0.2</td>
<td>0.6406</td>
<td>17815</td>
</tr>
<tr>
<td>0.25</td>
<td>0.9843</td>
<td>416</td>
</tr>
</tbody>
</table>

- Estimated probability of \( F \text{ Satur} \) being true for a 3\(^{rd}\) order \( \Delta - \Sigma \) modulator.
- Consistent with results obtained in [Dang et al 04] with reachability techniques.
- Our approach needed **seconds** while [Dang et al 04] needed **hours** of computation time.
- Experiments with 5\(^{th}\) and 7\(^{th}\) order \( \Delta - \Sigma \) modulators showed higher likelihoods of saturation.
Model Checking of Simulink stochastic models: $M \models P_{\geq \theta}(\Phi)$ ?
Future Work: Cost-Based Bayesian MC

- Model Checking query: $\mathcal{M} \models P_{\geq \theta}(\Phi)$, for $0 < \theta < 1$.
- $C(N)$: Cost of generating the $N^{th}$ sample.
- $R(u, \theta)$: Cost of incorrectly deciding the MC query
  - $u$ is the (unknown) probability that $\mathcal{M}$ satisfies $\Phi$
  - $\theta$ is the probability threshold in the specification
- Then, the key problem is to compute $E[R(u, \theta) \mid X_N]$
  - expected cost of a wrong decision after observing $N$ samples $X_N = (x_1, \ldots, x_N)$
- Stopping Criterion:
  - Stop when cost exceeds the reduction in the expected cost of making a wrong decision.

$$C(N+1) \geq E[R(u, \theta) \mid X_{N+1}] - E[R(u, \theta) \mid X_N]$$
Conclusions

- Some evidence that Statistical MC scales to large systems
  - Simulink Models
  - Delta-Sigma Modulator

- We have developed a Bayesian MC algorithm which
  - is faster than state-of-the-art approaches,
  - can use prior knowledge about the system.

- Initial experiments on Simulink are encouraging.

- Plan:
  - More Simulink examples.
  - Extend our implementation to Verilog and analog circuit models.