Symbolic Model Checking
$10^{20}$ States and Beyond

Burch  Clarke  McMillan  Dill  Hwang

Seminal Papers in Verification

March 23, 2012
Outline

1. The Mu-Calculus
2. Model Checking
3. Example
4. Results
The Mu-Calculus

The Mu-Calculus is similar to standard first-order logic.

- Does not include relational symbols or constant symbols.
- Relational symbols are replaced by relational variables.
- $\mu P[R]$ denotes the least fixed point of an $n$-ary relational term $R$ and $P$ is an $n$-ary relational variable.
Symbolic Model Checking

- Use BDDs as internal representation
- Recursively translate formula to BDD
- CTL expressions can be translated into efficient BDD operations.
- FalseBDD and TrueBDD correspond to trees with only one terminal node, 0 or 1 respectively.
Translating formulas

- Over the structure of formulas & terms

### BDD\_f: Formulas

- \( f \) is individual var: \( \text{BDDAtom}(f) \)
- \( f = f_1 \land f_2 \): \( \text{BDDAnd}(\text{BDD}_f(f_1), \text{BDD}_f(f_2)) \)
- \( f = \neg f_1 \): \( \text{BDDNegate}(\text{BDD}_f(f_1)) \)
- \( f = \exists x. f \): \( \text{BDDExists}(x, \text{BDD}_f(f_1)) \)
- \( f = R(x_1, \ldots, x_n) \): \( \text{BDD}_R(R)(d_1 \leftarrow x_1, \ldots, d_n \leftarrow x_n) \)

### BDD\_R: Terms

- \( R \) is relational var: \( l_R(R) \)
- \( R = \lambda x_1, \ldots, x_n.f \): \( \text{BDD}_f(f)(x_1 \leftarrow d_1, \ldots, x_n \leftarrow d_n) \)
- \( R = \mu P[R'] \): \( \text{FixedPoint}(P, R', \text{FalseBDD}) \)
The Mu-Calculus
Model Checking
Example
Results

- $\text{AF } f_1 = \mu Z . f_1 \lor AX Z$
- $\text{EF } f_1 = \mu Z . f_1 \land EX Z$
- $A[f_1 U f_2] = \mu Z . f_2 \lor (f_1 \land AX Z)$
- $E[f_1 U f_2] = \mu Z . f_2 \lor (f_1 \land EX Z)$
The set of atomic prepositions
\( AP = \{ a, b, c \} \)

The set of states
\( S = \{ s_0, s_1, s_2 \} \)

The set of transitions
\( T = \{ (s_0, s_1), (s_1, s_0), (s_0, s_2), (s_2, s_1) \} \)

The labelling function
\( L = \{ (s_0, \{ a, b \}), (s_1, \{ b, c \}), (s_2, \{ a, c \}) \} \)
CTL formulae:
\[ f = EX \, c \]

Mu-Calculus:
\[ R = \lambda s[\exists t[c(t) \land T(s, t)]] \]
States are described by means of a vector of boolean variables

\[ s_i = (x_1, x_2) \]

Boolean vectors can be represented as formulas

\[ s_0 = \neg e_1 \land e_2, \ s_1 = \neg e_1 \land e_2, \ s_2 = e_1 \land e_2 \]

Transitions, described by the pairs \((s_i, s'_i)\), can be represented as

\[ s_i \land s'_i \]
The Mu-Calculus
Model Checking
Example
Results

c(t)

\[ T(s, t) \]

Andreína Francisco
Symbolic Model Checking
\( c(t) \land T(s, t) \)
\[ \exists t [ c(t) \land T(s, t) ] \]

\[
e'_{2} = 0 \land e'_{1} = 0
\]

\[ 0 \]

\[
e'_{2} = 0 \land e'_{1} = 1
\]

\[ 0 \]

\[
e'_{2} = 1 \land e'_{1} = 0
\]

\[ 0 \rightarrow 1 \]

\[
e'_{2} = 1 \land e'_{1} = 1
\]

\[ 0 \rightarrow 1 \]
\[ \exists t [c(t) \land T(s, t)] = [c(t) \land T(s, t)] e_2^0, e_1^0 \lor [c(t) \land T(s, t)] e_2^0, e_1^1 \lor \ldots \]
Symbolic model checking allows larger models (many magnitudes).

- Interesting result: BDDs grow linearly
- State space very large
- Execution time still rises quickly
Symbolic Model Checkers

- Most hardware design companies have their own Symbolic Model Checker(s)
  - Intel, IBM, Motorola, Siemens, ST, Cadence, ...
  - very advanced tools
  - proprietary technology!

- On the academic side
  - CMU SMV [McMillan]
  - VIS [Berkeley, Colorado]
  - Bwolen Yang's SMV [CMU]
  - NuSMV [CMU, IRST, UNITN, UNIGE]
  - ...

Alessandro Artale
Formal Methods Lecture VII  Symbolic Model Checking