Verification – Lecture 21
Quotienting Algorithms for Bisimulation

Bernd Finkbeiner – Sven Schewe
Rayna Dimitrova – Lars Kuhtz – Anne Proetzsch

Wintersemester 2007/2008

REVIEW

Bisimulation equivalence

Let $S_i = (Q_i, Q_{0,i}, E_i, L_i)$, $i{=}1,2$, be two state graphs over *AP*.

A *bisimulation* for (S_1, S_2) is a binary relation $\mathcal{R} \subseteq Q_1 \times Q_2$ such that:

1. $\forall q_1 \in Q_{0,1}\, \exists q_2 \in Q_{0,2}.\ (q_1, q_2) \in \mathcal{R}$ and
   $\forall q_2 \in Q_{0,2}\, \exists q_1 \in Q_{0,1}.\ (q_1, q_2) \in \mathcal{R}$

2. for all states $q_1 \in Q_1$, $q_2 \in Q_2$ with $(q_1, q_2) \in \mathcal{R}$ it holds:

   (a) $L_1(q_1) = L_2(q_2)$

   (b) if $q_1' \in$ *Successors*(q_1) then there exists $q_2' \in$ *Successors*(q_2) with $(q_1', q_2') \in \mathcal{R}$

   (c) if $q_2' \in$ *Successors*(q_2) then there exists $q_1' \in$ *Successors*(q_1) with $(q_1', q_2') \in \mathcal{R}$

S_1 and S_2 are bisimilar, denoted $S_1 \sim S_2$, if there exists a bisimulation for (S_1, S_2)

Coarsest bisimulation

$\sim_S$ is an equivalence and the coarsest bisimulation for S

Quotient state graph

For $S = (Q, Q_0, E, L)$ and bisimulation $\sim_S \subseteq S \times S$ on S let

$S/\sim_S = (Q', Q'_0, E', L')$ be the *quotient* of S under $\sim_S$

where

- $Q' = S/\sim_S = \{ [q]_\sim \mid q \in Q \}$ with $[q]_\sim = \{ q' \in Q \mid q \sim_S q' \}$
- $Q'_0 = \{ [q]_\sim \mid q \in Q_0 \}$
- $E' = \{([q]_\sim, [q']_\sim) \mid (q, q') \in E\}$
- $L'([q]_\sim) = L(q)$

note that $S \sim S/\sim_S$ Why?

Bisimulation vs. CTL* and CTL equivalence

Let S be a *finite* state graph and s, s' states in S

The following statements are equivalent:

(1) $s \sim_S s'$

(2) s and s' are CTL-equivalent, i.e., $s \equiv_{CTL} s'$

(3) s and s' are CTL*-equivalent, i.e., $s \equiv_{CTL^*} s'$

this is proven in three steps: $\equiv_{CTL} \subseteq \sim \subseteq \equiv_{CTL^*} \subseteq \equiv_{CTL}$

important: equivalence is also obtained for any sub-logic containing $\neg$, $\wedge$, and $\exists\bigcirc$

The importance of this result

- CTL and CTL* equivalence coincide
  - despite the fact that CTL* is more expressive than CTL
- Bisimilar transition systems preserve the same CTL* formulas
  - and thus the same LTL formulas (and LT properties)
- Non-bisimilarity can be shown by a single CTL (or CTL*) formula
  - $S_1 \models \Phi$ and $S_2 \not\models \Phi$ implies $S_1 \not\sim S_2$
- You even do not need to use an until-operator!
- To check $S \models \Phi$, it suffices to check $S/\sim \models \Phi$

Bisimulation quotient state graph

For $S = (Q, Q_0, E, L)$ and bisimulation $\sim_S \subseteq Q \times Q$ on S let

$$S/\sim_S = (Q', Q'_0, E', L') \quad \text{be the } \textit{quotient} \text{ of } S \text{ under } \sim_S$$

where

- $Q' = Q/\sim_S = \{ [q]_\sim \mid q \in Q \}$ with $[q]_\sim = \{ q' \in Q \mid q \sim_S q' \}$
- $Q'_0 = \{ [q]_\sim \mid q \in Q_0 \}$
- $E' = \{([q]_\sim, [q']_\sim) \mid (q, q') \in E\}$
- $L'([q]_\sim) = L(q)$

note that $S \sim S/\sim_S$

Quotient state graph / Partitioning

For $S = (Q, Q_0, E, L)$ and an *equivalence relation* $\sim \subseteq Q \times Q$ on S let

$$S/\sim = (Q', Q'_0, E', L') \quad \text{be the } \textit{quotient} \text{ of } S \text{ under } \sim\text{, where}$$

- $Q' = Q/\sim = \{ [q]_\sim \mid q \in Q \}$ with $[q]_\sim = \{ q' \in Q \mid q \sim q' \}$
- $Q'_0 = \{ [q]_\sim \mid q \in Q_0 \}$
- $E' = \{([q]_\sim, [q']_\sim) \mid (q, q') \in E\}$
- $L'([q]_\sim) = L(q)$

A *partition* $\Pi = \{B_1, \ldots, B_k\}$ of Q is a set of nonempty ($B_i \neq \emptyset$) and pairwise disjoint *blocks* B_i that decompose Q ($Q = \biguplus_{i=1,\ldots k} B_i$).

A partition defines an equivalence relation $\sim$ ($(q, q') \in \sim \Leftrightarrow \exists Q_i \in \Pi.\, q, q' \in B_i$).

Likewise, an equivalence relation $\sim$ defines a partition $\Pi = Q/\sim$.

Blocks, Superblocks, and Stability

A *partition* $\Pi = \{B_1, \ldots, B_k\}$ of Q is a set of nonempty ($B_i \neq \emptyset$) and pairwise disjoint *blocks* B_i that decompose Q ($Q = \biguplus_{i=1,\ldots,k} B_i$).

A nonempty union $C = \biguplus_{i \in I} B_i$ of blocks is called a *superblock*.

A block B_i of a partition Π is called *stable* w.r.t. a set B if either $B_i \cap \textit{Pre}(B) = \emptyset$, or $B_i \subseteq \textit{Pre}(B)$.

($\textit{Pre}(B) = \{q \in Q \mid \textit{Successors}(q) \cap B \neq \emptyset\}$)

A partition Π is called *stable* w.r.t. a set B if all blocks of Π are.

Lemma 1. A partition Π with consistently labeled blocks is stable with respect to all of its (super)blocks if, and only if, it is the quotient of a bisimulation relation ($\Pi = Q/\sim$).

Partition refinement

For two partitions $\Pi = \{B_1, \ldots, B_k\}$ and $\Pi' = \{B'_1, \ldots, B'_j\}$ of Q, we say that Π is finer than Π' iff every block of Π' is a superblock of Π.

For a given partition $\Pi = \{B_1, \ldots, B_k\}$, we call a (super)block C of Π a *splitter* of a block B_i / the partition Π if B_i / Π is not stable w.r.t. C.

Refine(B_i, C) denotes $\{B_i\}$ if B_i is stable w.r.t. C, and $\{B_i \cap \textit{Pre}(C), B_i \smallsetminus \textit{Pre}(C)\}$ if C is a splitter of C.

Refine$(\Pi, C) = \biguplus_{i=1,\ldots,k}$Refine$(B_i, C)$.

Lemma 2. Refine(Π, C) is finer than Π.

Lemma 3. If Π is finer than Π' then Refine(Π, C) is finer than Refine(Π', C).

Algorithms for bisimulation quotienting

Input: Transition system $S = (Q, Q_0, E, L)$

Output: Bisimulation quotient state graph

1. $\Pi = Q/\sim_{AP}$ $\qquad (q, q') \in \sim_{AP} \Leftrightarrow L(q) = L(q')$
2. while some block $B \in \Pi$ is a splitter of Π loop invariant: Π is coarser than $Q/\sim_S$
   (a) pick a block B that is a splitter of Π
   (b) $\Pi =$ Refine(Π, B)
3. return Π

Correctness and termination

1. $\Pi = Q/\sim_{AP}$ $\qquad (q, q') \in \sim_{AP} \Leftrightarrow L(q) = L(q')$
2. while some block $B \in \Pi$ is a splitter of Π loop invariant: Π is coarser than $Q/\sim_S$
   (a) pick a block B that is a splitter of Π
   (b) $\Pi =$ Refine(Π, B)
3. return Π

Lemma 4. The algorithm terminates.

Lemma 5. The loop invariant holds initially.

Lemma 6. The loop invariant is preserved.

Theorem 7. The algorithm returns the quotient $Q/\sim_S$ of the coarsest bisimulation $\sim_S$.

Complexity

1. $\Pi = Q/{\sim_{AP}}$
2. while some block $B \in \Pi$ is a splitter of Π
   (a) pick a block B that is a splitter of Π
   (b) $\Pi =$ Refine(Π, B)
3. return Π

$(q,q') \in {\sim_{AP}} \Leftrightarrow L(q) = L(q')$

loop invariant: Π is coarser than $Q/{\sim_S}$

Lemma 8. $Q/{\sim_{AP}}$ can be constructed in time $\mathcal{O}(|Q| \cdot |AP|)$.

Proof Idea. Build tree that branches by the atomic propositions. The leafs are labeled with the elements of $Q/{\sim_{AP}}$.

The complexity of each refinement step depends on the strategy how B is picked.

Refinement complexity

2. while some block $B \in \Pi$ is a splitter of Π
   (a) pick a block B that is a splitter of Π
   (b) $\Pi =$ Refine(Π, B)

Trying all $B \in \Pi$ takes $\mathcal{O}(|E|)$ time.

– There may be $\mathcal{O}(|Q|)$ splits.

Corollary 9. The overall algorithm takes $\mathcal{O}(|Q| \cdot (|AP| + |E|))$ time.

Refinement complexity

2. while some block $B \in \Pi$ is a splitter of Π
   (a) pick a block B that is a splitter of Π
   (b) $\Pi =$ Refine(Π, B)

Trying all $B \in \Pi$ takes $\mathcal{O}(|E|)$ time.

– There may be $\mathcal{O}(|Q|)$ splits.

Corollary 9. The overall algorithm takes $\mathcal{O}(|Q| \cdot (|AP| + |E|))$ time.

– but we can do better –

An improved algorithm for bisimulation quotienting

Input: Transition system S $= (Q, Q_0, E, L)$

Output: Bisimulation quotient state graph

1. $\Xi = \{Q\}$
2. $\Pi = Q/\sim_{AP}$
3. while $\Xi \neq \Pi$
   (a) Pick $B \in \Xi \setminus \Pi$
   (b) Pick $B' \in \Pi$ such that $B' \subseteq B$ and $|B'| \leqslant \frac{1}{2}|B|$
   (c) $\Xi = (\Xi \setminus \{B\}) \cup \{B'\} \cup \{B \setminus B'\}$
   (d) $\Pi =$ Refine$\Big($Refine$(\Pi, B'), B \setminus B'\Big)$
4. return Π

Extra Challenge Question: Prove that the algorithm in the script is wrong. (31.5 Pts)

Termination

1. $\Xi = \{Q\}$
2. $\Pi = Q/{\sim_{AP}}$
3. while $\Xi \neq \Pi$
   (a) Pick $B \in \Xi \setminus \Pi$
   (b) Pick $B' \in \Pi$ such that $B' \subseteq B$ and $|B'| \leqslant \frac{1}{2}|B|$
   (c) $\Xi = (\Xi \setminus \{B\}) \cup \{B'\} \cup \{B \setminus B'\}$
   (d) $\Pi = \text{Refine}\Big(\text{Refine}(\Pi, B'), B \setminus B'\Big)$
4. return Π

Lemma 10. The loop invariant Ξ is coarser than Π is coarser than $Q/{\sim_S}$ holds.

Lemma 11. Ξ is strictly refined in every step of the while loop.

Correctness

1. $\Xi = \{Q\}$
2. $\Pi = Q/{\sim_{AP}}$
3. while $\Xi \neq \Pi$
   (a) Pick $B \in \Xi \setminus \Pi$
   (b) Pick $B' \in \Pi$ such that $B' \subseteq B$ and $|B'| \leqslant \frac{1}{2}|B|$
   (c) $\Xi = (\Xi \setminus \{B\}) \cup \{B'\} \cup \{B \setminus B'\}$
   (d) $\Pi = \text{Refine}\Big(\text{Refine}(\Pi, B'), B \setminus B'\Big)$

   until $\Xi = \Pi$
4. return Π

Lemma 12. If Π is finer than Π' and Π' is stable w.r.t. a set $C \subseteq Q$ than Π is stable w.r.t. C.

Proof Sketch. If $A \in \Pi$ is splitted and $\Pi' \ni A' \supseteq A$ than A' is splitted.

Theorem 13. The algorithm returns the partition $Q/{\sim_S}$ of the coarsest bisimulation $\sim_S$.

Proof Idea. Loop invariant: Π is stable w.r.t. every block in Ξ.
$\Rightarrow$ Π is stable w.r.t. every block in $\Pi = \Xi$