Symmetry in Model Checking

Elena Petrova

April 30, 2003

1 Introduction

A difficulty in the automatic verification of finite-state systems is the state space explosion problem. The size of the state space to be verified may grow exponentially with the description size of such a system. Many strategies have been proposed to solve the state space explosion problem. In this report an approach using structural symmetry in the structure of a system will be considered.

The problem using symmetry in model checking is to reduce the model checking of an original state transition graph M to the model checking of a smaller quotient structure M'.

This works well for systems which are composed of many identical or isomorphic processes (components), the global state transition graph of such a system includes many symmetric states. Formally, the symmetry can be characterised by the group of graph automorphisms of the state transition graph.

The theoretical base and an example of using symmetry in model checking will be presented in this report.

2 Model of computation

We define a state transition graph of a concurrent system as a tuple $M = (S, R, S_0, L)$, where

- $S \subseteq D^n$ is the set of states with
  - D is the finite set of individual process locations (local states)
  - $I = \{1,\ \dots,\ n\}$ is the set of process indices
- $R \subseteq S \times S$ is the set of transition relations
- $S_0 \subseteq S$ is the set of initial states
- L is the labelling function.

Also we define the set of indexed atomic propositions $AP = \{d_i \mid\ d \in D, i \in I\}$.

Each state s can be written as an n-tuple $s = (d_1, d_2, ..., d_n)$ indicating that process i is in state d_i for all $i \in I$.

3 Some Theory

3.1 Groups

Group theory defines a group G as a set G together with a binary operation on G, called the group multiplication $\circ$. Let $a, b, c \in G$. The multiplication $\circ$ must satisfy the following properties:

- it is associative: $a \circ (b \circ c) = (a \circ b) \circ c$
- there exists an identity element $e \in G$ such that $e \circ a = a = a \circ e$ for any element $a \in G$
- for each element a in the group there exists an inverse element a^{-1} such that $a \circ a^{-1} = a^{-1} \circ a = e$

Also, we define H as a subgroup of G if H $\subseteq$ G and H is a group under the multiplication operation of G. We denote $H \leq G$ for any group H which is subgroup of the group G.

3.2 Permutations

A permutation π on a finite set I is a bijective mapping $\pi : I \rightarrow I$. The set of all permutations on I forms the full symmetric group under functional composition. This set is denoted as $Sym\ I$. Any subgroup of $Sym\ I$ is called a permutation group on I.

Let $I = \{1,\ 2,\ ..., n\}$ be a finite set of process indices and let J be a subset of I such that

$$J = \{j_1,\ ...,\ j_l\} \subseteq I \qquad \text{for } l \leq n$$

A permutation π on I is called a cycle iff for any $i \in I$

$$\begin{aligned}
\pi(j_i) &= j_{i+1} && \text{for } i \in J\ :\ i \neq l \\
\pi(j_l) &= j_1 \\
\pi(i) &= i && \text{for } i \notin J.
\end{aligned}$$

A cycle permutation is denoted as $(j_1\ ...\ j_l)$

A permutation π acting on the set of indices $I = \{1,\ 2,\ ...,\ n\}$ defines a new permutation acting on the set of local states. Let state $s = (d_1, d_2, ..., d_n) \in S$, then the permutation on this state is a simple renaming of indices using the permutation π, i.e. $\pi(s) = (d_{\pi(1)}, d_{\pi(2)}, ..., d_{\pi(n)})$. A permutation on the formula over indexed atomic proposition can be defined in the same way — it is just the renaming of proposition indices with permutation π.

The identity permutation Id on any state or formula is a mapping of this state or formula to itself. We define $Id : I \rightarrow I$ such that $Id(i) = i$ for all indices $i \in I$.

3.3 Automorphisms

Let $M = (S, R, L)$ be a Kripke structure, let G be a permutation group on S. A permutation $\pi \in G$ is an automorphism on M if it preserves the transition relation, i.e. any edge in M is also an edge in M after applying the permutation π to its ending nodes:

$$\forall s_1, s_2 \in S, \ \ (s_1, s_2) \in R \ \Rightarrow \ (\pi(s_1), \pi(s_2)) \in R$$

G is an automorphism group for the Kripke structure M if and only if every permutation $\pi \in G$ is an automorphism of M.

3.4 Automorphisms of formulas

For a CTL* formula f we define the formula automorphism as

$$Aut\ f = \{\pi \in Sym\ I \mid \ \pi(f) \equiv f\},$$

where $\equiv$ denotes logical equivalence.

The equivalence of two formulas f_1 and f_2 is defined as

$$f_1 \equiv f_2 \qquad \Leftrightarrow \qquad \forall\, M, x: \quad M, x \models f_1 \ \text{ iff } \ M, x \models f_2$$

For example, for the formula $f = P_1 \wedge P_2$ and the permutation $\pi = (1\ 2)$, we have

$$\pi(f) = P_{\pi(1)} \wedge P_{\pi(2)} = P_2 \wedge P_1 \equiv P_1 \wedge P_2 = f.$$

Hence, automorphism for the given formula f is $Aut\ f = \{Id,\ (1\ 2)\}$.

Let us show that $Aut\ f$ is a subgroup of the symmetric group $Sym\ I$.

Lemma 1: For all CTL* formula f we have $Aut\ f \leq Sym\ I$.
Proof:

First, $Aut\ f$ includes the identity permutation, since $Id(f) = f \equiv f$ (by definition).

Second, for each element π_1 of the group $Aut\,f$ the inverse element π_1^{-1} from $Sym\ I$ is also an automorphism:
By definition of the inverse element we have $\pi_1(f) \equiv f = Id(f) = \pi_1 \circ \pi_1^{-1}(f)$. If we multiply this equivalence with π_1^{-1} from the left on both sides, we get $f \equiv \pi_1^{-1}$.

Third, for each pair of elements π_1, π_2 from $Aut\ f$ the concatenation of these two permutations is also an automorphism.

Since π_2 is an automorphism then $\pi_2(f) \equiv f$. Therefore, we have $\pi_1 \circ \pi_2(f) \equiv \pi_1(f)$. It is clear, that $\pi_1 \circ \pi_2(f) \equiv f$ also holds.

$Aut\ f$ is used to capture the "top-level" symmetry of any formula. We also use $Auto\ f$ such that $Auto\ f \leq Aut\ f$, which is used to find the "internal" symmetry of f.

$Auto\ f$ is defined inductively in the following way:

- For a propositional formula f, we define $Auto\ f = Aut\ f$
- For a general CTL* formula f, we define $Auto\ f$ according to the following cases:
  - $f = \mathrm{X}g$ or $f = \mathrm{E}g$: in this case $Auto\ f = Auto\ g$
  - $f = g\ \mathrm{U}\ h$: in such case $Auto\ f = Auto\ g \cap Auto\ h$
  - If neither of the above conditions hold then f is a boolean combination of atomic propositions and subformulas of the form $\mathrm{X}g$, $\mathrm{E}g$, and $g\ \mathrm{U}\ h$. That is $f = b(e_1, e_2..., e_k, f_1, f_2, ..., f_l)$, where b is a boolean formula over atomic propositions $e_1, e_2, ..., e_k$ and subformulas $f_1, f_2, ..., f_l$, where each f_i is of the form $\mathrm{X}g$ or $\mathrm{E}g$ or $g\ \mathrm{U}\ h$. Now, we replace each of f_i in the formula b by a new unindexed proposition F_i and define

    $$\begin{aligned} Auto\ f = &\ Auto\ b(e_1, e_2, ..., e_k, F_1, F_2..., F_l) \\ &\cap Auto\ f_1 \cap Auto\ f_2 \cap \cdots \cap Auto\ f_l \end{aligned}$$

    It is clear, that $b(e_1, e_2, ..., e_k, F_1, F_2, ..., F_l)$ is a propositional formula.

Consider the following example:
The set of process indices $I = \{1, 2\}$ and the given formula is:

$$f = P_1 \wedge \mathrm{EX}(Q_1 \vee Q_2) \vee P_2 \wedge \mathrm{EX}(Q_1 \vee Q_2).$$

Then $Auto\ f = Auto\ (P_1 \wedge B \vee P_2 \wedge B) \cap Auto\ (\mathrm{EX}(Q_1 \vee Q_2))$, where B is a new unindexed atomic proposition replacing the subformula $\mathrm{EX}(Q_1 \vee Q_2)$.

Now, let us show that $Auto\ f$ is actually a subgroup of $Aut\ f$.

Lemma 2: For all CTL* formula f we have $Auto\ f \leq Aut\ f$
Proof:

We need to show, that for each permutation $\pi \in Auto\ f$ the statement $\pi(f) \equiv f$ holds. The proof proceeds by induction on the formula structure.

Base case:

Formula f is a propositional formula and $Auto\ f = Aut\ f$ by definition of $Auto\ f$. Hence, $\pi \in Aut\ f$ and $\pi(f) \equiv f$.

Induction step:

Now we have several cases depending on the formula form:

- $f = g\ \mathrm{U}\ h$
  By induction hypothesis for the subformulas g and h we have $Auto\ g \leq Auto\ f$ and $Auto\ h \leq Aut\ f$. Also, by definition of $Auto\ f$ for a formula of such form we have $Auto\ g \geq Auto\ f$ and $Auto\ h \geq Auto\ f$. Hence, if the permutation π is in $Auto\ f$ then it is also in $Auto\ g$ and $Auto\ h$.
  Let us show that permutation π on the formula f is equivalent to the

formula f.
By the definition of Until, $M, x \models f$ if and only if there exists $i \geq 0$ such that $M, x^{(i)} \models h$ and for all j such that $0 \leq j \leq i$ we have $M, x^{(j)} \models g$. So,

$$M, x \models f = g\mathrm{U}h \quad \Leftrightarrow$$
$$\exists i \geq 0 \;\; M, x^{(i)} \models h, \;\; \forall j \;\; 0 \leq j \leq i \quad M, x^{(j)} \models g$$

By induction hypothesis for g and h and since permutation π is in *Auto* g and also in *Auto* h we have that

$$\exists i \geq 0 \text{ s.t. } M, x^{(i)} \models h, \text{ and for all } 0 \leq j \leq i \quad M, x^{(j)} \models g \text{ holds } \Leftrightarrow$$
$$\exists i \geq 0 \text{ s.t. } M, x^{(i)} \models \pi(h), \text{ and for all } 0 \leq j \leq i \quad M, x^{(j)} \models \pi(g) \text{ holds}$$

Using the Until definition we have that

$$\exists i \geq 0 \text{ s.t. } M, x^{(i)} \models \pi(h), \text{ and for all } 0 \leq j \leq i \quad M, x^{(j)} \models \pi(g) \text{ holds } \Leftrightarrow$$
$$M, x \models \pi(g) \mathrm{~U~} \pi(h) \quad \Leftrightarrow$$
$$M, x \models \pi(g \mathrm{~U~} h) \; = \; \pi(f)$$

The last transformation holds, since permutation π does not affect the Until operator.

- $f = \mathrm{E}g$
  By definition of the formula automorphism *Auto* $f =$ *Auto* g, by induction hypothesis for subformula g we have *Auto* $g \leq$ *Aut* f. Hence, if permutation π is in *Auto* f then $\pi \in$ *Auto* g. Let us show the equivalence of the formulas f and $\pi(f)$.
  By the definition of Exists

$$M, x \models f = \mathrm{E}g \quad \Leftrightarrow$$
$$\exists \; x' \text{ starting } s_0 \text{ such that } M, x' \models g$$

Since $g \equiv \pi(g)$ (by induction hypothesis)

$$\exists \; x' \text{ starting } s_0 \text{ such that } M, x' \models g \Leftrightarrow$$
$$\exists \; x' \text{ starting } s_0 \text{ such that } M, x' \models \pi(g)$$

Using the definition of Exists we have

$$\exists \; x' \text{ starting } s_0 \text{ such that } M, x' \models \pi(g) \quad \Leftrightarrow$$
$$M, x \models \mathrm{E}\pi(g) \quad \Leftrightarrow$$
$$M, x \models \pi(\mathrm{E}g) = \pi(f)$$

As in the previous case permutation π does not affect Exists.

- $f = \mathrm{X}g$
  By definition of the formula automorphism $Auto\ f = Auto\ g$, and by induction hypothesis for subformula g we obtain $Auto\ g \le Aut\ f$ and, hence, permutation $\pi \in Auto\ g$. The equivalence of the formulas f and $\pi(f)$ can be shown in the following way:
  By the definition of neXt

$$M, x \models f = \mathrm{X}g \quad \Leftrightarrow \\ M, x^{(1)} \models g$$

  Since the equivalence $g \equiv \pi(g)$ we have

$$M, x^{(1)} \models g \quad \Leftrightarrow \\ M, x^{(1)} \models \pi(g)$$

  By the definition of neXt and using the fact that permutation π does not affect X we have

$$M, x^{(1)} \models \pi(g) \quad \Leftrightarrow \\ M, x \models \mathrm{X}\pi(g) \quad \Leftrightarrow \\ M, x \models \pi(\mathrm{X}g) = \pi(f)$$

- $f = b(e_1, e_2, ..., e_k, f_1, f_2, ..., f_l)$
  By induction hypothesis, we have for all $0 \le i \le l$ that $Auto\ f_i \le Aut\ f$. By the definition of the formula automorphism for a boolean combination of subformulas and atomic propositions,

$$Auto\ f = Auto\ b \cap Auto\ f_1 \cap Auto\ f_2 \cap \cdots \cap Auto\ f_l.$$

  If $\pi \in Auto\ f$ then $\pi \in Auto\ f_i$ for all i such that $0 \le i \le l$.
  Let us define boolean constants y_i for all i such that $0 \le i \le k$ and boolean constants z_j for all j such that $0 \le j \le l$ in the following way:

$$y_i = 1 \quad \Leftrightarrow \quad M, x \models f_i \ \Leftrightarrow M, x \models \pi(f_i) \\ z_j = 1 \quad \Leftrightarrow \quad M, x \models e_j \ \Leftrightarrow M, x \models \pi(e_j)$$

  Obviously, that

$$M, x \models b(e_1, e_2, ..., e_k, f_1, f_2, ..., f_l) \Leftrightarrow \\ M, x \models b(z_1, z_2, ..., z_k, y_1, y_2, ..., y_l) \Leftrightarrow \\ M, x \models b(\pi(e_1), \pi(e_2), ..., \pi(e_k), \pi(f_1), \pi(f_2), ..., \pi(f_l)) \Leftrightarrow \\ M, x \models \pi(b(e_1, e_2, ..., e_k, f_1, f_2, ..., f_l))$$

Putting all observations together we get the induction step.

3.5 Quotient structure

Now we consider how to define a smaller quotient structure for an original state transition graph M which consists of some identical components.

Let G be any subgroup of $Sym\ I$. We define the equivalence relation $\equiv_G$ on the states in S, where $s \equiv_G t$ iff there is a permutation $\pi \in G$ such that $t = \pi(s)$. We define the equivalence class for a state s such that $[s]_G = \{t \mid s \equiv_G t\}$.

Now, the idea is to collapse each equivalence class to one of its members. Therefore, for each equivalence class $[s]_G$ we choose a unique representative $\overline{s}$. Let $\overline{S} \subseteq S$ denote the set of all representatives.

The quotient of M modulo $\equiv_G$ is the Kripke structure $(\overline{S}, \overline{R}, \overline{L})$, where:

- we have $\overline{s} \to \overline{t} \in \overline{R}$ iff there exist states s' and t' such that $s' \equiv_G \overline{s},\ t' \equiv_G \overline{t}$ and there exists a transition relation $s' \to t' \in R$
- $\overline{L}(x) = L(x)$ for all representatives $x \in \overline{S}$

We use the notation M/G for the quotient of M modulo G.

4 Compression Theorem

In this section we show how to use quotient structures to check formulas in the original structures.

4.1 Correspondence Lemma

Now we consider the lemma about path correspondence in the original state transition graph and the quotient state transition graph.

Let M be an original structure and M/G be a quotient structure for M. For a sequence $x = (s_0, s_1, ...)$ of states in S, we define the sequence of the corresponding representatives in $\overline{S}$ as $\overline{x} = (\overline{s}_0, \overline{s}_1, ...)$.

Correspondence Lemma:
There is a bidirectional correspondence between paths of the original structure M and the quotient structure $\overline{M} = M/G$ for any $G \leq AutM$:

1. If $x = s_0, s_1, ...$ is a path in M, then $\overline{x}$ is a path in $\overline{M}$.
2. If $\overline{x} = \overline{s}_0, \overline{s}_1, ...$ is a path in $\overline{M}$, then for every state $s'_0 \equiv_G \overline{s}_0$ in M there exists a corresponding path $x' = s'_0, s'_1, ...$ in M of states such that $s_i \equiv_G \overline{s}_i$.

Proof:
The first part of the lemma follows immediately from the definition of the quotient structure, so $(s_0 \to s_1) \in R \ \Leftrightarrow\ (\overline{s}_0 \to \overline{s}_1) \in \overline{R}$. Applying this observation to each s_i from x, we get the first statement.

Now, let $\overline{x} = \overline{s}_0, \overline{s}_1, ...$ be a path in $\overline{M}$. Choose an arbitrary $s'_0 \equiv_G \overline{s}_0$. There exists $s''_0 \equiv_G \overline{s}_0$ and there exists $s''_1 \equiv_G \overline{s}_1$ such that $s''_0 \to s''_1 \in R$ (by definition of the quotient structure and since $\overline{s}_0 \to \overline{s}_1 \in \overline{R}$).

By transitivity, $s'_0 \equiv_G \overline{s}_0$ and $s''_0 \equiv_G \overline{s}_0$ imply $s'_0 \equiv s''_0$. For some permutation $\pi \in G$ we have that $s'_0 = \pi(s''_0)$. Let $s'_1 = \pi(s''_1)$. The transition $s'_0 \to s'_1 = \pi(s''_0) \to \pi(s''_1) \in M$ since $s''_0 \to s''_1 \in M$ and $\pi \in G \leq AutM$. The state s'_1 is equivalent to representative $\overline{s}_1$, namely $s'_1 = \pi(s''_1) \equiv_G \overline{s}_1$.

This define the first edge $s'_0 \to s'_1$ in the corresponding path x' in the quotient structure. Continuing with state s'_1, the same argument can be applied to obtain s'_2 such that $s'_1 \to s'_2 \in M$. Continuing in this way we see that there is a correspondence between $s'_i \to s'_{i+1} \in M$ and transition from $\overline{x}$, namely $\overline{s}_i \to \overline{s_{i+1}} \in \overline{M}$. This process continues for all i.

So, we get $x' = s'_0 \to s'_1 \to s'_2 \to ...$ a corresponding path to x in the quotient structure in $\overline{M}$. This proves the second part of the Correspondence Lemma.

4.2 Significant subformulas

Let f be any CTL* formula. A subset of subformulas of f, called significant subformulas, is defined as follows:

- f is a significant subformula of itself
- For every subformula g of f which is of the form $\mathrm{X}g'$ or $\mathrm{E}g'$, both g and g' are significant subformulas of f. Also, for every subformula g of the form g' U h', g, g' and h' are significant subformulas of f.

So, g is a significant subformula of f if either g is the same as f, or the outermost connective of g is a temporal operator or is a path quantifier, or g can be an immediate argument of a subformula whose outer most connective is a temporal operator or a path quantifier.

Consider an example:
For the formula $f = \mathrm{E}g$, where the subformula $g = (P_1 \vee P_2) \wedge ((Q_1 \vee Q_2)\mathrm{U}(R_1 \vee R_2))$ the set of significant subformulas is equal to

$$\{f, g, (Q_1 \vee Q_2)\mathrm{U}(R_1 \vee R_2), (Q_1 \vee Q_2), (R_1 \vee R_2)\}$$

.

Now let us prove that automorphism of a formula is a subset of a its significant subformula automorphism.

Lemma 3:
For every significant subformula h of f, *Auto* $f \subseteq$ *Auto* h.

Proof:
Let g be any significant subformula of f. We define the *immediate* significant subformulas of g as follows.

- If subformula g is of the form $\mathrm{X}g'$ or $\mathrm{E}g'$, then g' is an immediate significant subformula of g.

- If subformula g is of the form g' U h', then g' and h' are the immediate significant subformulas of g.
- If neither of these conditions hold then g is a boolean combination of atomic propositions and significant subformulas $f_1, f_2, ..., f_k$, where each of f_i is of the form Xg', Eg' or g' U h'. In this case $f_1, f_2, ..., f_k$ are the immediate significant subformulas of g.

For every immediate significant subformula g' of g we have $Auto g \subseteq Auto\ g'$ (by the definition of $Auto\ g$). Applying this inductively for significant subformulas of g, we have for every significant subformula g' of g that $Auto\ g \subseteq Auto\ g'$. The claim of the lemma follows by using f and h in place of g and g' as in the above observation.

4.3 Compression theorem

Now we show the central theorem allowing to use a smaller quotient structure instead the original structure for verification.

Compression theorem:

For all structures M, all CTL* formulas f, all subgroups $G \leq AutM \cap Autof$, and all paths x in M,

$$M, x \models f \text{ iff } M/G, \overline{x} \models f$$

Proof:

By induction on formula structure, we show that for every significant subformula g and every path x from M,

$$M, x \models g \text{ iff } M/G, \overline{x} \models g$$

Let the function *count(g)* return the number of symbols U, X or E in the formula g. The proof of the theorem is performed by induction on *count*(g).

Induction base:

For *count(g)* $= 0$ the subformula g is a propositional formula.
Hence, $M, x \models g \Leftrightarrow s_0$ satisfies g and $M/G, \overline{x} \models g \Leftrightarrow \overline{s_0}$ satisfies g.
There exists a permutation $\pi \in G$ such that $\overline{s_0} = \pi(s_0)$. By the theorem assumption $\pi \in G \Rightarrow \pi \in Auto\ f$ and $Auto\ f \subseteq Auto\ g$ since g is a significant subformula of f (Lemma 3), from this we see that $\pi \in Auto\ g$. Hence, g is equivalent to $\pi(g)$. State s_0 satisfies g iff $\pi(s_0)$ satisfies g iff $\overline{s_0}$ satisfies g. Hence, $M, x \models g$ iff $M/G, \overline{x} \models g$.

Induction step:

Assume that this lemma holds for all maximal subformulas g' such that *count*$(g') \leq k$ and for all paths in M. Let the formula g be any maximal subformula such that *count*$(g') = k+1$.
For the induction step we have the following cases:

- g =Eg'
$M, x \models g$ if and only if there exist a path x' in M starting from s_0 such that $M, x' \models g'$. Using the induction hypothesis ($count(g') = k$) we have

$$M, x' \models g' \;\Leftrightarrow\; M/G, \overline{x'} \models g' \quad (1)$$

Since paths $\overline{x'}$ and $\overline{x}$ start from the same state in $\overline{S}$, namely $\overline{s_0}$, using the definition of E, we have

$$M/G, \overline{x'} \models g' \;\Leftrightarrow\; M/G, \overline{x} \models \mathrm{E}g' \;\Leftrightarrow\; M/G, \overline{x} \models g \quad (2)$$

Induction step follows from the statements (1) and (2).

- g=Xg'
By definition of X,

$$M, x \models g \Leftrightarrow\; M, x^{(1)} \models g',$$

From the induction hypothesis follows

$$M, x^{(1)} \models g' \Leftrightarrow M/G, \overline{x^{(1)}} \quad (1).$$

Using the equality $\overline{x^{(1)}} = \overline{x}^{(1)}$ and the definition of X, we obtain

$$M/G, \overline{x^{(1)}} \Leftrightarrow M/G, \overline{x} \models g \quad (2).$$

The induction step follows from the observations (1) and (2).

- g=g'Uh' where $count(g')$, $count(h')$$\leq k$
By the definition of Until, $M, x \models g$ iff $M, x^{(i)} \models h'$ for some $i \geq 0$ and for all j such that $0 \leq j < i$ we have that $M, x^{(j)} \models g'$. By induction hypothesis, the later condition holds iff $M/G, \overline{x^{(j)}} \models h'$ and for all j such that $0 \leq j < i$ we have $M/G, \overline{x^{(j)}} \models g'$. By definition of Until the last condition holds iff

$$M/G, \overline{x} \models g'\mathrm{U}h' = M/G, \overline{x} \models g.$$

The induction step follows from these observations.

- Other cases:
In this case, a formula g is a boolean combination of atomic propositions and maximal subformulas whose outermost connective is from the set {X, U, E}, i.e. $g = b(e_1, e_2, ..., e_k, f_1, f_2, ..., f_l)$, where b is a boolean formula over indexed atomic propositions $e_1, e_2, ..., e_k$ and maximal subformulas $f_1, f_2, ..., f_l$ such that, for each i the outer most connectives of f_i belongs to {X, U, E} and $count(f_i) = k$+1.
Using the induction hypothesis for the previous three cases, we see that for each i such that $1 \leq i \leq l$,

$$M, x \models f_i \text{ iff } M/G, \overline{x} \models f_i.$$

From the definition of $Auto\ g$, we have

$$Auto\ g \leq Auto\ b(e_1, e_2, ..., e_k, , F_1, F_2, ..., F_l),$$

where $F_1, F_2, ..., F_l$ are some distinct unindexed atomic propositions. From Lemma 3 we have that $Autof \subseteq Auto\ g$. Hence,

$$G \leq Auto\ b(e_1, e_2, ..., e_k, F_1, F_2, ..., F_l).$$

For each i such that $1 \leq i \leq l$, let u_i be a boolean constant defined as follows:

$$\begin{array}{ll} u_i = true & \text{if } M, x \models f_i \\ u_i = true & \text{otherwise} \end{array}$$

It is easy to see that $M, x \models g$ iff the state s_0 satisfies Boolean formula $b(e_1, e_2, ..., e_k, u_1, u_2, ..., u_l)$.
Let $\pi \in G$ be such that $\overline{s_0} = \pi(s_0)$. Clearly,

$$\begin{array}{c} s_0 \text{ satisfies } b(e_1, e_2, ..., e_k, u_1, u_2, ..., u_l) \quad \Leftrightarrow \\ \overline{s_0} \text{ satisfies } \pi(b(e_1, e_2, ..., e_k, u_1, u_2, ..., u_l)). \end{array}$$

Since $G \leq Auto\ b(e_1, e_2, ..., e_k, F_1, F_2, ..., F_l)$, a permutation π is in $Auto\ b(e_1, e_2, ..., e_k, F_1, F_2, ..., F_l)$.
Hence, a formula $\pi(\, b(e_1, e_2, ..., e_k, u_1, u_2, ..., u_l))$ is equivalent to a formula $b(e_1, e_2, ..., e_k u_1, u_2, ..., u_l)$ and also

$$\begin{array}{c} \text{state } s_0 \text{ satisfies } b(e_1, e_2, ..., e_k, u_1, u_2, ..., u_l) \Leftrightarrow \\ \text{state } \overline{s_0} \text{ satisfies } b(e_1, e_2, ..., e_k, u_1, u_2, ..., u_l). \end{array}$$

Therefore, $\overline{s_0}$ satisfies $b(e_1, e_2, ..., e_k, u_1, u_2, ..., u_l)$ iff $M/G, \overline{x} \models g$.
Putting all the above observation together, we get the induction step.

5 Example

Consider a token ring algorithm with n concurrent processes.

Each process can be in one of three states: the process does not have the token (n), process has the token (t) or the process is in the critical section (c). The idea of a token-ring is that process i can only enter the critical section if it has the token. Process i in state n can receive token only from previous process (with index i-1), otherwise it must stay in n state. If the process has token, in the next step it can go to the critical section, stay with the token or send the token to the next process. If the process is in the critical section, only two variants are possible: to stay in the critical section or to return in the t state. The state transition graph for one process in the token-ring is depicted in Figure 1.

Now, let us formalise this protocol.
The state space of this system is $S = D^I$, where

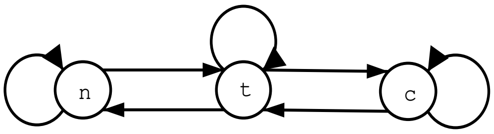

Figure 1: The transition state graph for one process

- $I = \{1, ..., n\}$ is the set of process indices
- $D = \{t, n, c\}$ is the set of local states of each process.

The set of indexed atomic propositions for this system is $AP = \{t_i\} \cup \{n_i\} \cup \{c_i\}$

The set of transition relation is defined as follows:

$$\begin{array}{rlll} R = \{(s, s') \mid \forall i : & d_i = t & \Rightarrow & d'_i \in \{t, n, c\} \\ & d_i = c & \Rightarrow & d'_i \in \{t, c\} \\ & d_i = n & \Rightarrow & d'_i = t \text{ if } send(i-1) \\ & & & d'_i = n \text{ otherwise, } \} \end{array}$$

where $send(i-1) = (d_{i-1} = t) \wedge (d'_{i-1} = n)$. d' means a state for d at the next time.

In this system only $2n$ states are reachable (since initially we only have one token):
$C_i = \ (n, ..., n, c, n, ..., n)$
$T_i = \ (n, ..., n, t, n, ..., n)$

The labels c and t are at positions with number i in the states C_i and T_i correspondingly.

The given system can be described with the transition graph M presented in Figure 2.

In this case an automorphism group for the system M is induced by the cycle permutation $\pi_c = (1\ 2\ ...\ \text{n})$, which corresponds to rotations of the ring. There are two equivalence classes of the states: T=$\{T_i\}$ and C=$\{C_i\}$ for all i. We choose representatives for each equivalence class: let $T_1 = \overline{T}$ be the representative of class T and $C_n = \overline{C}$ be the representative of class C.

The corresponding quotient structure for system M is $M/G = (\overline{S}, \overline{R}, \overline{S_0}, \overline{L})$, where:

- $\overline{S} = \{T_1, C_n\}$ is the set of states. It is union of representatives.
- $\overline{R} = \{(T_1, T_1), (T_1, C_n), (C_n, T_1), (C_n, C_n)\}$ is the set of all transition relation in this system.

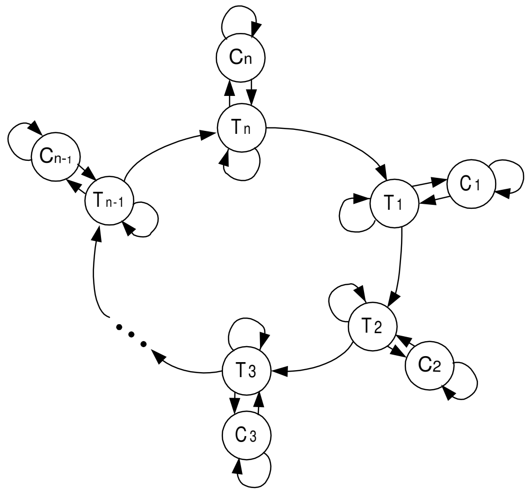

Figure 2: The initial transition state graph of the system

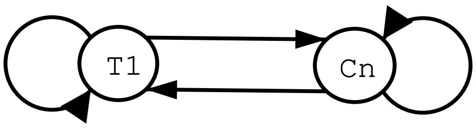

Figure 3: The quotient structure M/G

- $\overline{S_0} = \{\overline{s_0}\} = \{\overline{(t, n, ..., n)}\} = \{T_1\}$ is an initial state since only one process has token.
- $L(\overline{T}) = L(T_1) = \{t_1, n_2, n_3, ..., n_n\}$ and
  $L(\overline{C}) = L(C_n) = \{n_1, n_2, n_3, ..., c_n\}$ are labels for each state

The transition graph for the quotient structure is presented in Figure 3.

We want to verify the mutual exclusion property of the system M using the compression theorem. By the mutual exclusion property no two processes i and j may be in the critical section at the same time.

Verification of the mutual exclusion property of the initial system can be replaced with verification of quotient model (by the Compression theorem). Hence, we need to show the correctness of following formula:
$f = AG(\wedge_i (c_i \Rightarrow \neg \vee_{j \neq i} \ c_j))$, where

As easy to see $Auto\ f = Auto\ (\wedge_i (c_i \Rightarrow \neg \vee_{j \neq i}\ c_j)) = Sym\ I$ and

$$G(\pi_c) \leq Auto\ f \cap Auto\ M = Auto\ M$$

The formula correctness is provided by two following observations in M/G:

1. in state T_1 there is no local state with label c_i;
2. in state C_n c_i=1 only for case $i = n$.

Hence, the formula holds for the quotient structure and accordingly for the initial structure M.

Bibliography

1. Emerson, Sistla — Symmetry and Model Checking, Formal Methods in System Design, 9(1/2), 1996
2. Clarke, Grumberg, Peled — Model Checking, 1999