Algorithmic Verification of Channel Machines Using Small Models

Jonathan Sharyari

Department of Information Technology
Uppsala University

Supervisor: Parosh Abdulla

Reviewer: Mohamed Faouzi Atig

April 20^{th}, 2016

1. **General Verification**
2. **All for the Price of Few**
   - Parameterized Systems
   - Small Models
   - View Abstraction
   - Verification Algorithm
3. **Objective**
4. **Method**
   - Channel Systems
   - Transition System
   - α and γ
5. **Results**

Verification

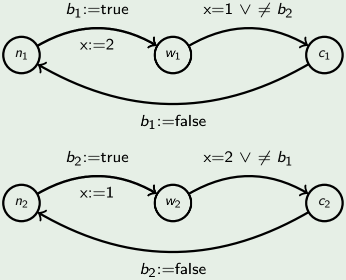
- Verification is the *"process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase"*

Peterson's Mutual Exclusion – Pseudo Code

- **while** true **do**
  $\langle b_1 := true, x = true \rangle$;
  wait while $(x = true \wedge b_2 = true)$;
  CRITICAL SECTION
  b_1 := false
- **while** true **do**
  $\langle b_2 := true, x = false \rangle$;
  wait while $(x = false \wedge b_1 = true)$;
  CRITICAL SECTION
  b_1 := false

Petersons's Mutual Exclusion – Program Graph

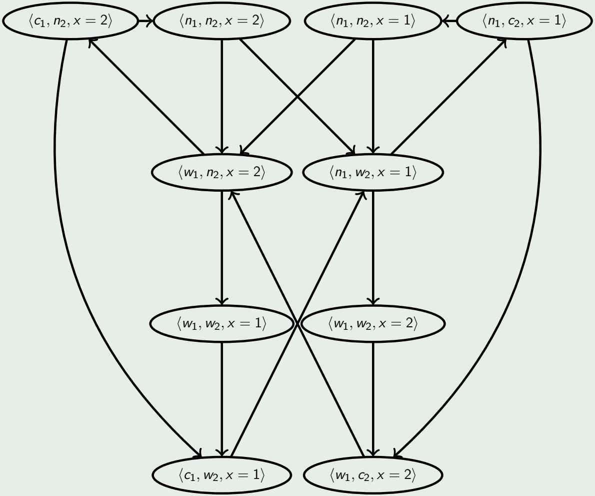
Example (Program Graphs)



- $\langle\ c_1, c_2\ \rangle$ is a *bad global state*

Peterson's Mutual Exclusion – Transition System

Example (Transition System)



All for the Price of Few

- Builds upon work by Parosh Abdulla, Frédéric Haziza and Lukáš Holík, *All for the Price of Few*, 2013
- Parameterized Systems
- Small models
- View abstraction

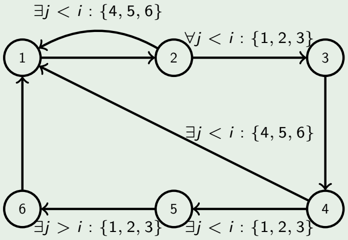
Parameterized Systems

- The size of the system is a parameter of the system
- Results in the verification of an infinite system
- Example: unbounded number of participants, unbounded integers, unbounded channels

Parameterized Systems – Burns' Protocol

Example (Burns' Protocol)

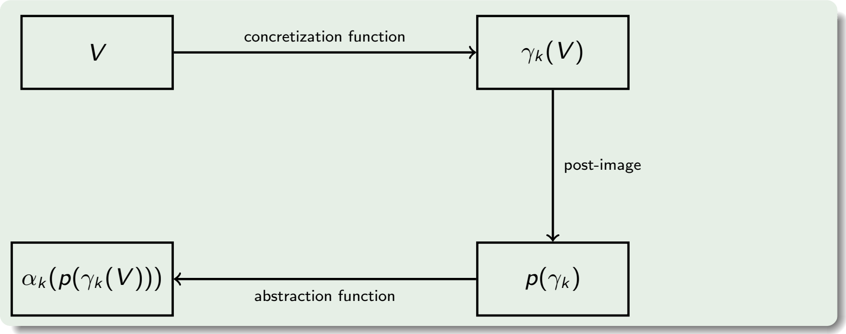
flags[i] :=0;
if $\exists j < i : \mathit{flag}[i] = 1$ **then**
    goto 1;
flags[i] :=1;
if $\exists j < i : i : \mathit{flag}[i] = 1$ **then**
    goto 1;
await $\forall$ j>i: flag[j] $\neq$ 1;
flag[i] :=0; **goto** 1;



Small Models

- In some cases, a "small" model of a system may exhibit all the relevant behaviour larger systems.
- It suffices to prove the correctness of a small model

View Abstraction



View Abstraction – Example

- Abstraction function α: The *subwords* of configurations

Example (Abstraction)

$\alpha_2(\{1,2,3\}) = \{\{1,2\},\{2,3\},\{1,3\},\{1\},\{2\},\{3\}\}$

- Concretization function γ: The "inverse" of the abstraction function

Example (Concretization)

$\gamma_3(\{\{1,2\},\{2,3\}, \{1,3\}, \{1\}, \{2\}, \{3\}\}) = \{1,2,3\}$

Verification algorithm

while *True* **do**
    if $\mathcal{R}_k \cap Bad \neq \emptyset$ **then** ▷ True if unsafe
        return Unsafe
    V $:= \mu X.\alpha_k(I) \cup Apost_k(X)$ ▷ fixpoint overapproximation
    if $\gamma_k(V) \cap Bad = \emptyset$ **then** ▷ True if safe
        return Safe
    k := k+1 ▷ Safe but not a small model

Goal

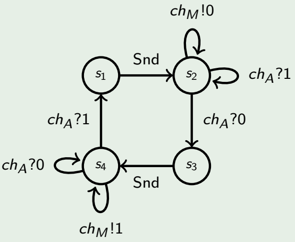
- Adapting the verification method to verify *channel systems*
- Implementing the verification method

Channel Systems

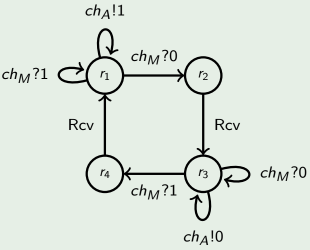
- A channel system is a system that relies on channels for its operation, e.g. communication protocols
- If channels are unbounded, the model checking of such protocols corresponds to searching an infinite graph

Alternating Bit Protocol – Program Graphs

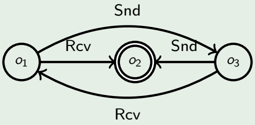
Example (ABP Program Graphs)



(a) ABP sender



(b) ABP receiver



(c) ABP observer

Channel Transition System

- Tuples $\langle S, \xi \rangle$, where S is a global state, and ξ is the *evaluation* of channel states.
- Alternating bit protocol: $\langle [s, r, o], [ch_M, ch_A] \rangle$

View Abstraction

$$\begin{tikzcd}
V \arrow[r, "\text{concretization function}"] & \gamma_k(V) \arrow[d, "\text{post-image}"] \\
\alpha_k(p(\gamma_k(V))) & p(\gamma_k) \arrow[l, "\text{abstraction function}"']
\end{tikzcd}$$

Abstraction Function

- Creates views of size k from configurations of size $k+1$
- The cross product of the subwords of all channels.

Example (α for ABP with $k = 2$)

$$\langle S, [110, 0]\rangle \in \gamma_{k+1}(V) \Rightarrow$$
$$\{\langle S, [11, 0]\rangle, \langle S, [11, \varepsilon]\rangle, \langle S, [10, 0]\rangle, \langle S, [10, \varepsilon]\rangle, \langle S, [1, 0]\rangle, \langle S, [1, \varepsilon]\rangle, \langle S, [0, 0]\rangle, \langle S, [0, \varepsilon]\rangle, \langle S, [\varepsilon, 0]\rangle, \langle S, [\varepsilon, \varepsilon]\rangle\} \subseteq V$$

Concretization function

- Creates configurations of size $k+1$ from views of size k
- The "inverse" of the abstraction function

Example (γ for ABP with $k = 2$)

$\{\langle S, [11, 0]\rangle, \langle S, [11, \varepsilon]\rangle, \langle S, [10, 0]\rangle, \langle S, [10, \varepsilon]\rangle, \langle S, [1, 0]\rangle,$
$\langle S, [1, \varepsilon]\rangle, \langle S, [0, 0]\rangle, \langle S, [0, \varepsilon]\rangle, \langle S, [\varepsilon, 0]\rangle, \langle S, [\varepsilon, \varepsilon]\rangle\} \subseteq V$
$\Rightarrow \{$
$\langle S, [110, 0]\rangle, \langle S, [110, \varepsilon]\rangle, \langle S, [111, 0]\rangle, \langle S, [111, \varepsilon]\rangle\} \in \gamma_{k+1}(V)$

Concretization function

- Creates configurations of size $k+1$ from views of size k
- The "inverse" of the abstraction function

Example (γ for ABP with $k=2$)

$\{\ \langle S,[11,0]\rangle,\ \langle S,[11,\varepsilon]\rangle,\ \langle S,[10,0]\rangle,\ \langle S,[10,\varepsilon]\rangle,\ \langle S,[1,0]\rangle,$
$\langle S,[1,\varepsilon]\rangle,\ \langle S,[0,0]\rangle,\ \langle S,[0,\varepsilon]\rangle,\ \langle S,[\varepsilon,0]\rangle,\ \langle S,[\varepsilon,\varepsilon]\rangle\ \} \subseteq V$
$\Rightarrow$
$\{\ \langle S,[110,0]\rangle,\ \langle S,[110,\varepsilon]\rangle,\ \langle S,[111,0]\rangle,\ \langle S,[111,\varepsilon]\rangle\} \in \gamma_{k+1}(V)$

Concretization function

- Creates configurations of size $k+1$ from views of size k
- The "inverse" of the abstraction function

Example (γ for ABP with $k=2$)

$\{ \langle S,[11,0]\rangle, \langle S,[11,\varepsilon]\rangle, \langle S,[10,0]\rangle, \langle S,[10,\varepsilon]\rangle, \langle S,[1,0]\rangle,$
$\langle S,[1,\varepsilon]\rangle, \langle S,[0,0]\rangle, \langle S,[0,\varepsilon]\rangle, \langle S,[\varepsilon,0]\rangle, \langle S,[\varepsilon,\varepsilon]\rangle \} \subseteq V$
$\Rightarrow$
$\{\langle S,[110,0]\rangle, \langle S,[110,\varepsilon]\rangle, \langle S,[111,0]\rangle, \langle S,[111,\varepsilon]\rangle\} \in \gamma_{k+1}(V)$

Concretization function

- Creates configurations of size $k+1$ from views of size k
- The "inverse" of the abstraction function

Example (γ for ABP with $k = 2$)

$\{\ \langle S,[11,0]\rangle,\ \langle S,[11,\varepsilon]\rangle,\ \langle S,[10,0]\rangle,\ \langle S,[10,\varepsilon]\rangle,\ \langle S,[1,0]\rangle,$
$\langle S,[1,\varepsilon]\rangle,\ \langle S,[0,0]\rangle,\ \langle S,[0,\varepsilon]\rangle,\ \langle S,[\varepsilon,0]\rangle,\ \langle S,[\varepsilon,\varepsilon]\rangle\ \} \subseteq V$
$\Rightarrow$
$\{\ \langle S,[110,0]\rangle,\ \langle S,[110,\varepsilon]\rangle,\ \langle S,[111,0]\rangle,\ \langle S,[111,\varepsilon]\rangle\} \in \gamma_{k+1}(V)$

Statistics

	k	size(V)	Result	Time	Mem	Backward Size V	Result	Time	MPass Result	Time
ABP	2	108	Safe	0.00s	1MB	56	Safe	0.01s	Safe	1.04s
SW3	3	4247	Safe	0.10s	3MB	270	Safe	0.17s	–	–
SW4	4	98629	Safe	3.64s	36MB	840	Safe	2.03s	–	–
SW5	5	1834345	Safe	120.20s	924MB	2028	Safe	24.31s	–	–
BRP	2	45	Safe	0.02s	3MB	–	??	Timeout	Safe	1.23s
ABP_F	1	–	Fail	0.00s	1MB	–	Fail	0.01s	Fail	6.04s
SW3_F	1	–	Fail	0.00s	1MB	–	Fail	0.10s	Fail	26.08s
BRP_F	1	–	Fail	0.00	2	–	Fail	0.15	–	–

References