Liveness Analysis for Parameterised Boolean Equation Systems

Jeroen Keiren (VU)
Joint work with Wieger Wesselink and Tim Willemse (TU/e)

Exact Sciences
Theoretical Computer Science

ATVA 2014, Sydney, 4 November 2014

VRIJE UNIVERSITEIT AMSTERDAM
Faculteit der Exacte Wetenschappen

Parameterised Boolean Equation Systems (PBESs)

Example

$$\begin{aligned}
\nu X(i,j,k,l\colon N) \quad &= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2,j,k,l+1)) \\
&\wedge (j = 1 \implies Y(i,1,1,k)) \\
&\wedge \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}$$

- System of mutually dependent fixpoint equations
- Parameterised with data
- Formulas over data and recursion variables
- Typically interested in solution to specific instance

Parameterised Boolean Equation Systems (PBESs)

Example

$$
\begin{aligned}
\nu X(i, j, k, l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2, j, k, l+1)) \\
&\land (j = 1 \implies Y(i, 1, 1, k)) \\
&\land \forall m\colon N.(j = 2 \implies Y(i, j, m+k, k)); \\
\nu Y(i, j, k, l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1, 1, l+1, 1)) \land X(2, 2, 1, l);
\end{aligned}
$$

- System of mutually dependent fixpoint equations
- Parameterised with data
- Formulas over data and recursion variables
- Typically interested in solution to specific instance

Parameterised Boolean Equation Systems (PBESs)

Example

$$\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\quad \land (j = 1 \implies Y(i,1,1,k)) \\
&\quad \land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\quad \land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}$$

- System of mutually dependent fixpoint equations
- Parameterised with data
- Formulas over data and recursion variables
- Typically interested in solution to specific instance

Parameterised Boolean Equation Systems (PBESs)

Example

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) \quad &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\land (j = 1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}
$$

init $X(1,1,1,1)$;

- System of mutually dependent fixpoint equations
- Parameterised with data
- Formulas over data and recursion variables
- Typically interested in solution to specific instance

Parameterised Boolean Equation Systems (PBESs)

Applications

- μ-calculus model checking (e.g. in mCRL2, CADP)
- Equivalence checking
- Solving Datalog queries

Parameterised Boolean Equation Systems (PBESs)

Applications

- μ-calculus model checking (e.g. in mCRL2, CADP)
- Equivalence checking
- Solving Datalog queries

Solving methods

- Fixed point iteration
- Tableaux
- Instantiation (like state space exploration) + solving resulting Boolean Equation System/Parity game

Motivating example

Instantiation

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) \quad &= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2,j,k,l+1)) \\
&\wedge (j = 1 \implies Y(i,1,1,k)) \\
&\wedge \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}
$$

Motivating example
Instantiation

$$
\begin{aligned}
\nu X(i, j, k, l\colon N) \quad &= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2, j, k, l+1)) \\
&\wedge (j = 1 \implies Y(i, 1, 1, k)) \\
&\wedge \forall m\colon N. (j = 2 \implies Y(i, j, m+k, k)); \\
\nu Y(i, j, k, l\colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1, 1, l+1, 1)) \wedge X(2, 2, 1, l);
\end{aligned}
$$

Solution for $X(1, 1, 1, 1)$?

Motivating example
Instantiation

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i=1 \land j=1 \land k=1 \implies X(2,j,k,l+1)) \\
&\land (j=1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j=2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j=2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}
$$

Solution for $X(1,1,1,1)$?

$$\nu X_{1,1,1,1} =$$

Motivating example

Instantiation

$$\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i=1 \land j=1 \land k=1 \implies X(2,j,k,l+1)) \\
&\land (j=1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j=2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j=2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}$$

Solution for $X(1,1,1,1)$?

$$\begin{aligned}
\nu X_{1,1,1,1} = {} & (1=1 \land 1=1 \land 1=1 \implies X(2,1,1,2)) \\
& \land (1=1 \implies Y(1,1,1,1))
\end{aligned}$$

Motivating example
Instantiation

$$\begin{aligned}
\nu X(i,j,k,l\colon N) \quad &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\land (j = 1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}$$

Solution for $X(1,1,1,1)$?

$$\begin{aligned}
\nu X_{1,1,1,1} = X_{2,1,1,2} &\land Y_{1,1,1,1} \\
&\land \forall m\colon N.(1 = 2 \implies Y(1,1,m+1,1))
\end{aligned}$$

Motivating example
Instantiation

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) \quad &= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2,j,k,l+1)) \\
&\wedge (j = 1 \implies Y(i,1,1,k)) \\
&\wedge \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}
$$

Solution for $X(1,1,1,1)$?

$$\nu X_{1,1,1,1} = X_{2,1,1,2} \wedge Y_{1,1,1,1} \wedge \mathit{true}$$

Motivating example

Instantiation

$$\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\land (j = 1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N. (j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}$$

Solution for $X(1,1,1,1)$?

$$\nu X_{1,1,1,1} = X_{2,1,1,2} \land Y_{1,1,1,1}$$

$$\vdots$$

$$\nu Y_{1,1,1,1} = (1 \neq 2 \implies 1 < 10) \land (1 = 2 \implies X(1,1,2,1)) \land X(2,2,1,1$$

Motivating example

Instantiation

$$\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2,j,k,l+1)) \\
&\wedge (j = 1 \implies Y(i,1,1,k)) \\
&\wedge \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}$$

Solution for $X(1,1,1,1)$?

$$\nu X_{1,1,1,1} = X_{2,1,1,2} \wedge Y_{1,1,1,1}$$

$$\vdots$$

$$\nu Y_{1,1,1,1} = X_{2,2,1,1}$$

Motivating example

Instantiation

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) \quad &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\land (j = 1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}
$$

Solution for $X(1,1,1,1)$?

$$\nu X_{1,1,1,1} = X_{2,1,1,2} \land Y_{1,1,1,1}$$

$$\nu X_{2,2,1,1} = true \land true \land \forall m\colon N.(2 = 2 \implies Y(2,2,m+1,1))$$

$$\vdots$$

$$\nu Y_{1,1,1,1} = X_{2,2,1,1}$$

Motivating example

Instantiation

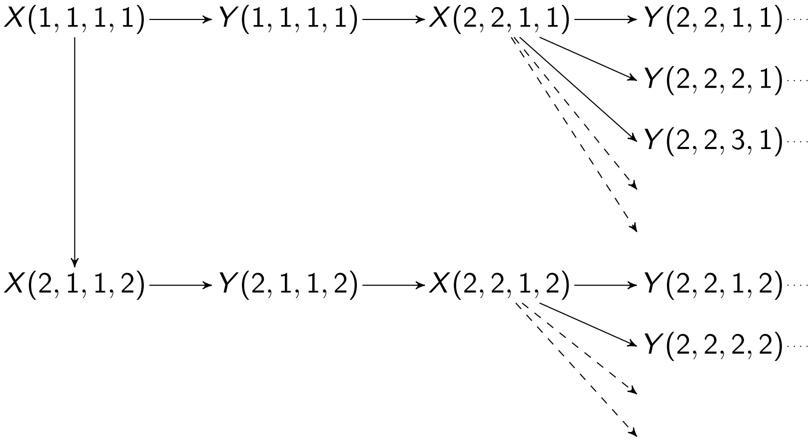
$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\quad \land (j = 1 \implies Y(i,1,1,k)) \\
&\quad \land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\quad \land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}
$$

Solution for $X(1,1,1,1)$?

$$
\begin{aligned}
\nu X_{1,1,1,1} &= X_{2,1,1,2} \land Y_{1,1,1,1} \\
\nu X_{2,2,1,1} &= Y_{2,2,1,1} \land Y_{2,2,2,1} \land Y_{2,2,3,1} \land \cdots \\
&\vdots \\
\nu Y_{1,1,1,1} &= X_{2,2,1,1} \\
&\vdots
\end{aligned}
$$

Motivating example

Dependency graph



Problem analysis

$$
\begin{aligned}
\nu X(i, j, k, l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2, j, k, l+1)) \\
&\land (j = 1 \implies Y(i, 1, 1, k)) \\
&\land \forall m\colon N.(j = 2 \implies Y(i, j, m+k, k)); \\
\nu Y(i, j, k, l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1, 1, l+1, 1)) \land X(2, 2, 1, l);
\end{aligned}
$$

Problem analysis

$$\begin{aligned}
\nu X(i,j,k,l\colon N) \;&= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2,j,k,l+1)) \\
&\wedge (j = 1 \implies Y(i,1,1,k)) \\
&\wedge \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \;&= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}$$

> $X(2,2,1,1)$ depends on infinitely many $Y(2,2,v+1,1)$;

Problem analysis

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) \quad &= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2,j,k,l+1)) \\
&\wedge (j = 1 \implies Y(i,1,1,k)) \\
&\wedge \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}
$$

- $X(2,2,1,1)$ depends on infinitely many $Y(2,2,v+1,1)$;
- Parameter k of Y only relevant when $j \neq 2$.

Problem analysis

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) \;&= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2,j,k,l+1)) \\
&\wedge (j = 1 \implies Y(i,1,1,k)) \\
&\wedge \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \;&= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}
$$

- $X(2,2,1,1)$ depends on infinitely many $Y(2,2,v+1,1)$;
- Parameter k of Y only relevant when $j \neq 2$.

Observation

Replacing $Y(i,j,m+k,k)$ by e.g. $Y(i,j,1,k)$ is sound!

Liveness Analysis

Problem

Modify PBES to avoid state space explosion

Liveness Analysis

Problem

Modify PBES to avoid state space explosion

Solution

1. Construct control flow graph
2. Detect dead variables
3. Simplify PBES

Construct control flow graph

What are control flow parameters?

Definition (Control flow parameter)

Parameter whose precise value is known throughout the PBES

Construct control flow graph

What are control flow parameters?

Definition (Control flow parameter)

Parameter whose precise value is known throughout the PBES

Example

$$\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\land (j = 1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}$$

Construct control flow graph

What are control flow parameters?

Definition (Control flow parameter)

Parameter whose precise value is known throughout the PBES

Example

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2,j,k,l+1)) \\
&\wedge (j = 1 \implies Y(i,1,1,k)) \\
&\wedge \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}
$$

Construct control flow graph

What are control flow parameters?

Definition (Control flow parameter)

Parameter whose precise value is known throughout the PBES

Example

$$
\begin{aligned}
\nu X(i, j, k, l \colon N) \quad &= (i = 1 \land j = 1 \land k = 1 \implies X(2, j, k, l + 1)) \\
&\land (j = 1 \implies Y(i, 1, 1, k)) \\
&\land \forall m \colon N. (j = 2 \implies Y(i, j, m + k, k)); \\
\nu Y(i, j, k, l \colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1, 1, l + 1, 1)) \land X(2, 2, 1, l);
\end{aligned}
$$

Construct control flow graph

What are control flow parameters?

Definition (Control flow parameter)

Parameter whose precise value is known throughout the PBES

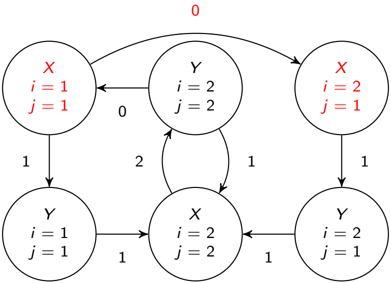
Example

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) \quad &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\land (j = 1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}
$$

> i^X, i^Y and j^X, j^Y are CFPs

Control flow graph

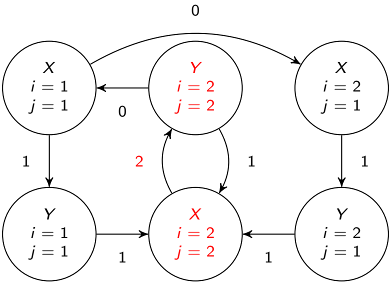
Running example



$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i=1 \land j=1 \land k=1 \implies X(2,j,k,l+1)) \\
&\land (j=1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j=2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j=2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}
$$

Control flow graph

Running example



$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\land (j = 1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}
$$

Analysis

How can we use the CFG to detect dead variables?

Analysis

How can we use the CFG to detect dead variables?

1. For each node in the CFG, identify possibly relevant parameters

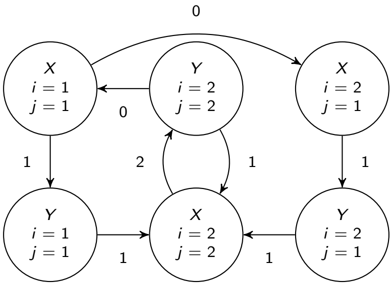
Analysis

How can we use the CFG to detect dead variables?

1. For each node in the CFG, identify possibly relevant parameters
2. Backwards reachability (fixed point computation)

Analysis

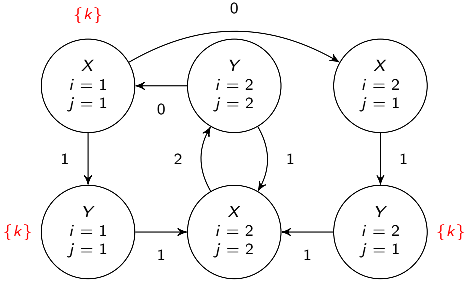
1. Identify possibly relevant parameters



$$
\begin{aligned}
\nu X(i,j,k,l\colon N) \quad &= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2,j,k,l+1)) \\
&\wedge (j = 1 \implies Y(i,1,1,k)) \\
&\wedge \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) \quad &= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}
$$

Analysis

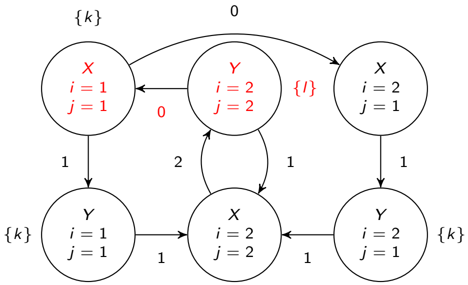
1. Identify possibly relevant parameters



$$\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\quad \land (j = 1 \implies Y(i,1,1,k)) \\
&\quad \land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\quad \land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}$$

Analysis

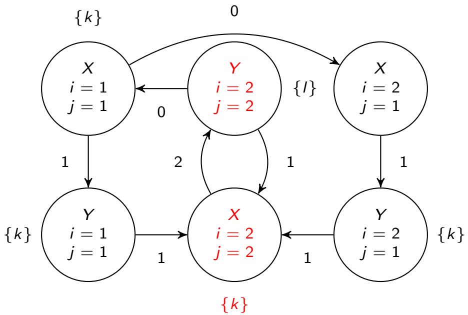
2. Backwards reachability



$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2,j,k,l+1)) \\
&\quad \land (j = 1 \implies Y(i,1,1,k)) \\
&\quad \land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\quad \land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}
$$

Analysis

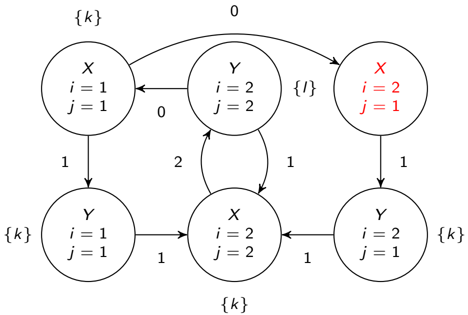
2. Backwards reachability



$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i=1 \land j=1 \land k=1 \implies X(2,j,k,l+1)) \\
&\land (j=1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j=2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j=2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}
$$

Reduction

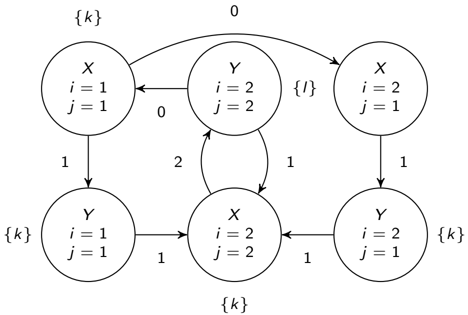
Set parameters to default value if not labelled.



$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \wedge j = 1 \wedge k = 1 \implies X(2,j,k,l+1)) \\
&\wedge (j = 1 \implies Y(i,1,1,k)) \\
&\wedge \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\wedge (j = 2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}
$$

Reduction

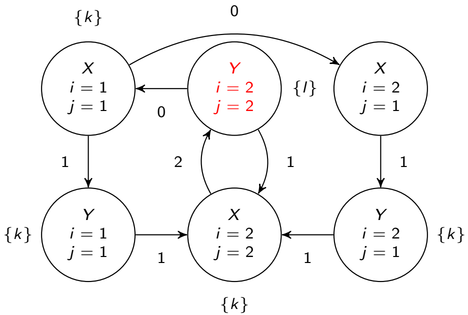
Set parameters to default value if not labelled.



$$\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2,1,1,1)) \\
&\land (j = 1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j = 2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}$$

Reduction

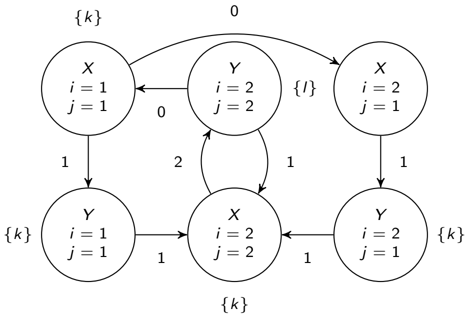
Set parameters to default value if not labelled.



$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i=1 \land j=1 \land k=1 \implies X(2,1,1,1)) \\
&\land (j=1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N.(j=2 \implies Y(i,j,m+k,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j=2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}
$$

Reduction

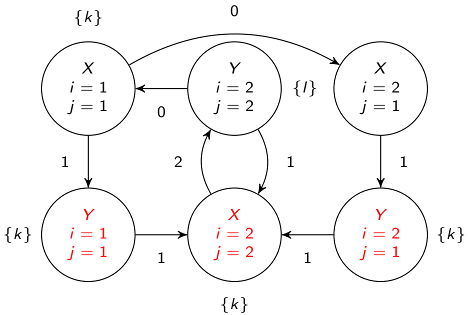
Set parameters to default value if not labelled.



$$\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2,1,1,1)) \\
&\land (j = 1 \implies Y(i,1,1,k)) \\
&\land \forall m\colon N. (j = 2 \implies Y(i,2,1,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,l);
\end{aligned}$$

Reduction

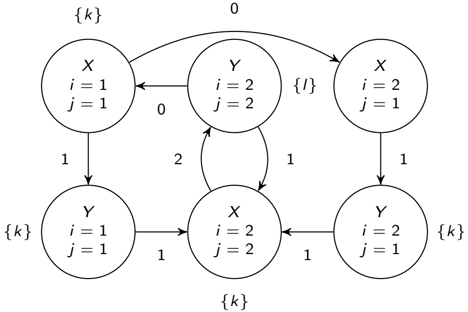
Set parameters to default value if not labelled.



$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i=1 \wedge j=1 \wedge k=1 \implies X(2,1,1,1)) \\
&\wedge (j=1 \implies Y(i,1,1,k)) \\
&\wedge (j=2 \implies Y(i,2,1,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k<10) \\
&\wedge (j=2 \implies X(1,1,l+1,1)) \wedge X(2,2,1,l);
\end{aligned}
$$

Reduction

Set parameters to default value if not labelled.



$$
\begin{aligned}
\nu X(i,j,k,l\colon N) &= (i = 1 \land j = 1 \land k = 1 \implies X(2,1,1,1)) \\
&\land (j = 1 \implies Y(i,1,1,1)) \\
&\land (j = 2 \implies Y(i,2,1,k)); \\
\nu Y(i,j,k,l\colon N) &= (j \neq 2 \implies k < 10) \\
&\land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,1);
\end{aligned}
$$

Reduction

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) \quad & = (i = 1 \lor j = 1 \lor k = 1 \implies X(2,1,1,1)) \\
& \land (j = 1 \implies Y(i,1,1,1)) \\
& \land (j = 2 \implies Y(i,2,1,k)); \\
\nu Y(i,j,k,l\colon N) \quad & = (j \neq 2 \implies k < 10) \\
& \land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,1); \\
\textbf{init } X(1,1,1,1);
\end{aligned}
$$

Finite instantiation!

Reduction

$$
\begin{aligned}
\nu X(i,j,k,l\colon N) \quad & = (i = 1 \lor j = 1 \lor k = 1 \implies X(2,1,1,1)) \\
& \land (j = 1 \implies Y(i,1,1,1)) \\
& \land (j = 2 \implies Y(i,2,1,k)); \\
\nu Y(i,j,k,l\colon N) \quad & = (j \neq 2 \implies k < 10) \\
& \land (j = 2 \implies X(1,1,l+1,1)) \land X(2,2,1,1); \\
\textbf{init } & X(1,1,1,1);
\end{aligned}
$$

Finite instantiation! (8 equations)

Optimisation

Problem

Size of CFG exponential in number of CFPs:

$$\nu X(i_1, \ldots, i_n \colon B) = (i_1 \wedge X(\mathit{false}, \ldots, i_n)) \vee (\neg i_1 \wedge X(\mathit{true}, \ldots, i_n)) \vee \\ \cdots \vee (i_n \wedge X(i_1, \ldots, \mathit{false})) \vee (\neg i_n \wedge X(i_1, \ldots, \mathit{true}))$$

Optimisation

Problem

Size of CFG exponential in number of CFPs:

$$\nu X(i_1, \ldots, i_n \colon B) = (i_1 \wedge X(\mathit{false}, \ldots, i_n)) \vee (\neg i_1 \wedge X(\mathit{true}, \ldots, i_n)) \vee \\ \cdots \vee (i_n \wedge X(i_1, \ldots, \mathit{false})) \vee (\neg i_n \wedge X(i_1, \ldots, \mathit{true}))$$

Resolution

> Local CFG for each CFP
> Link data parameters to CFPs
> Perform local analysis with limited global transfer of information

This approximates global analysis

Check out paper for details

Results

Reductions: Size

	$\|D\|$	Original	parelm	st.graph (global)	st.graph (local)	
Model Checking Problems						
No deadlock						
Hesselink	2	540,737	100%	100%	100%	√
	3	13,834,801	100%	100%	100%	√
Messages can overtake one another						
Onebit	2	164,353	63%	73%	70%	×
	4	1,735,681	88%	92%	90%	×
Values written to the register can be read						
Hesselink	2	1,093,761	1%	92%	92%	√
	3	27,876,961	1%	98%	98%	√
Equivalence Checking Problems						
Branching bisimulation equivalence						
ABP-CABP	2	31,265	0%	3%	0%	√
	4	73,665	0%	5%	0%	√
Buf-Onebit	2	844,033	16%	23%	23%	√
	4	8,754,689	32%	44%	44%	√
Hesselink I-S	2	21,062,529	0%	93%	93%	×

Results

Times

	$\|D\|$	Original	parelm	st.graph (global)	st.graph (local)	
Model Checking Problems						
No deadlock						
Hesselink	2	64.9	99%	95%	99%	√
	3	2776.3	100%	100%	100%	√
Messages can overtake one another						
Onebit	2	36.4	70%	67%	79%	×
	4	332.0	88%	88%	90%	×
Values written to the register can be read						
Hesselink	2	132.8	-3%	90%	91%	√
	3	5362.9	25%	98%	99%	√
Equivalence Checking Problems						
Branching bisimulation equivalence						
ABP-CABP	2	3.9	-4%	-1880%	-167%	√
	4	8.7	-7%	-1410%	-72%	√
Buf-Onebit	2	112.1	30%	28%	31%	√
	4	1344.6	35%	44%	37%	√
Hesselink I-S	2	4133.6	0%	74%	91%	×

Conclusions

Presented analysis that. . .

- Identifies CFPs in a PBES
- Detects dead variables
- Reduces PBES

Additionally:

- Local analysis to avoid exponential blow up
- Examples infinite to finite

Future work

- Use CFG to detect invariants
- Use CFG to improve existing heuristics
- Optimise implementation