Symbolic

Reachability Analysis

Gianpiero Cabodi *Stefano Quer*

Politecnico di Torino

Torino, Italy

{gianpiero.cabodi,stefano.quer}@polito.it

http://staff.polito.it/{gianpiero.cabodi,stefano.quer}/

Reference

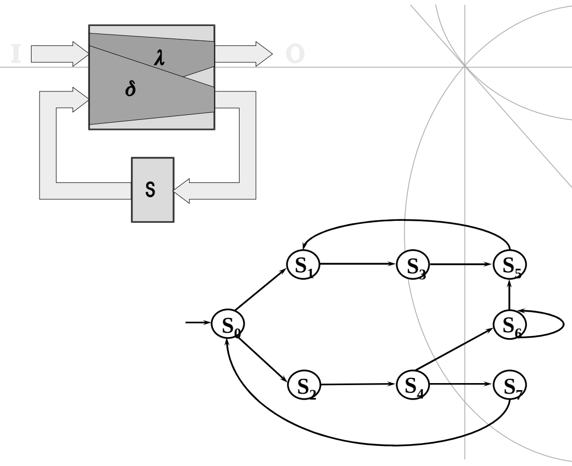
- Paper
- Books
  - C. Meinel, T. Theobald
    "Algorithms and Data Structure in VLSI Design"
    Springer-Verlag, Berlin, August 1998
    ISBN 3-540-64486-5
  - G. D. Hachtel, F. Somenzi
    "Loginc Synthesis and Verification Algorithms"
    Kluwer Academic Publishers

Outline

- Background
  - FSM Model and State Space Graph Representation
  - State Space Visit: DFS and BFS Paradigms
  - Functions and Sets Representation
  - Image Computation Concepts
  - Impact and Reference
- Transition Relation
- Image Computation
- Reachability Analysis
- Limits

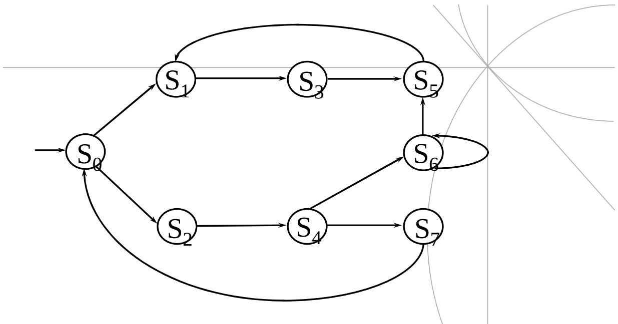
Outline

- Background
  - FSM Model and State Space Graph Representation
  - State Space Visit: DFS and BFS Paradigms
  - Functions and Sets Representation
  - Image Computation Concepts
  - Impact and Reference
- Transition Relation
- Image Computation
- Reachability Analysis
- Limits

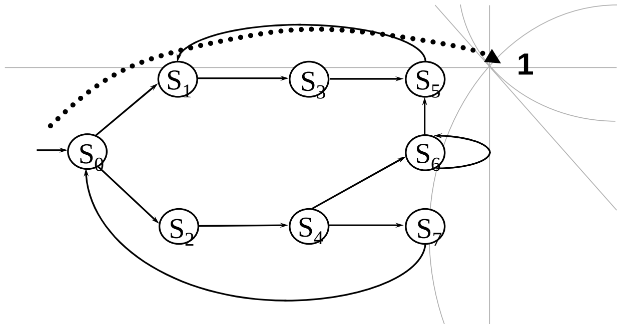


Outline

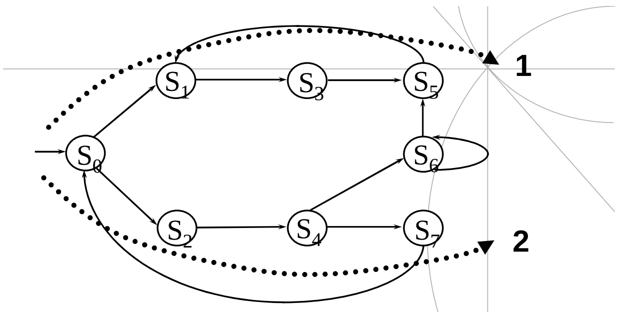
- Background
  - FSM Model and State Space Graph Representation
  - State Space Visit: DFS and BFS Paradigms
  - Functions and Sets Representation
  - Image Computation Concepts
  - Impact and Reference
- Transition Relation
- Image Computation
- Reachability Analysis
- Limits



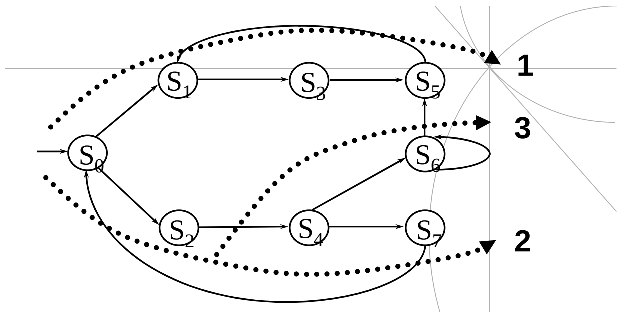
❖ **Depth-First Visit**



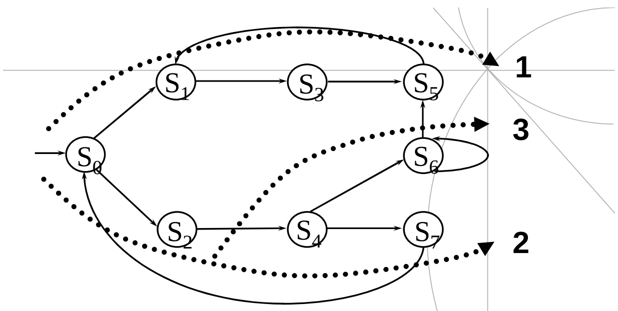
❖ **Depth-First Visit**



❖ **Depth-First Visit**

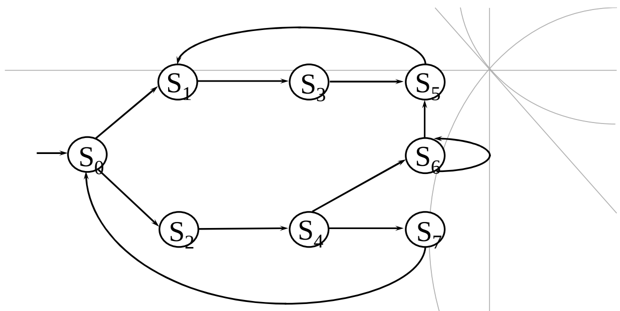


❖ **Depth-First Visit**

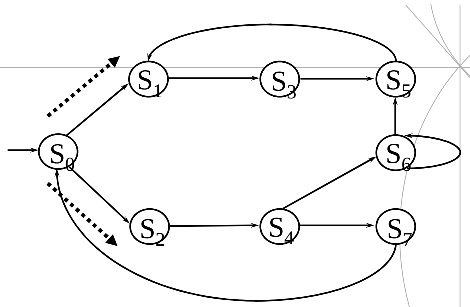


❖ **Depth-First Visit**

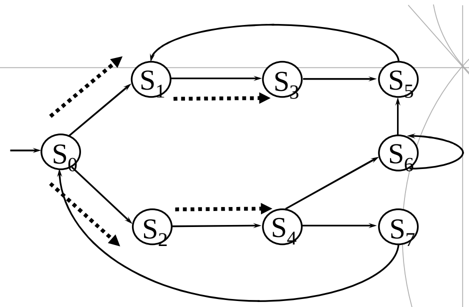
INEFFICIENT (it deals one state at a time)



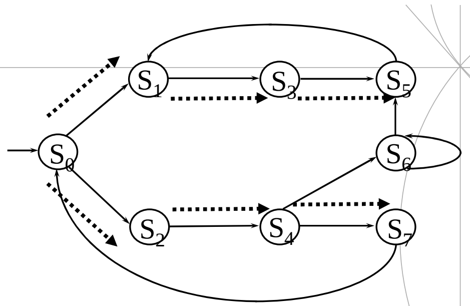
❖ **Breadth-First Visit**



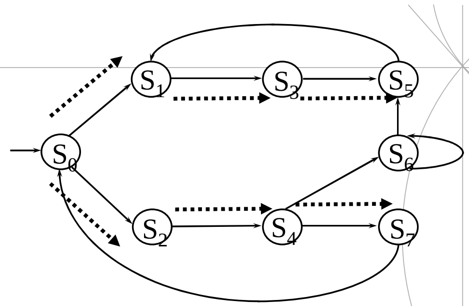
❖ **Breadth-First Visit**



❖ **Breadth-First Visit**

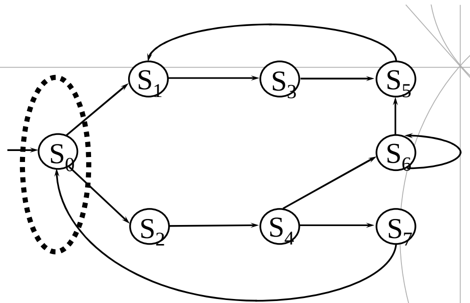


❖ **Breadth-First Visit**



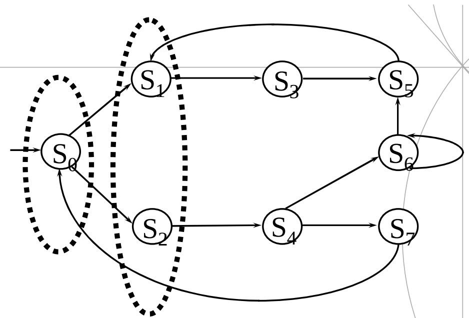
❖ **Breadth-First Visit**

EFFICIENT IFF we can deals with multiple states (set of state)



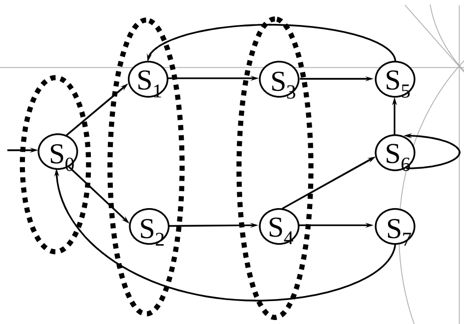
❖ **Breadth-First Visit**

EFFICIENT IFF we can deals with multiple states (set of state)



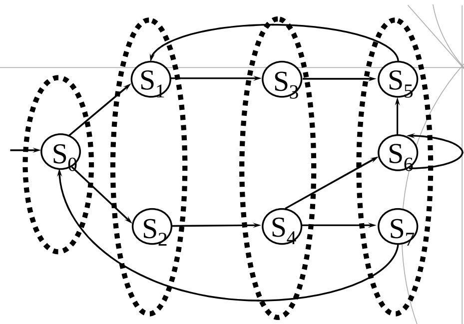
❖ **Breadth-First Visit**

EFFICIENT IFF we can deals with multiple states (set of state)



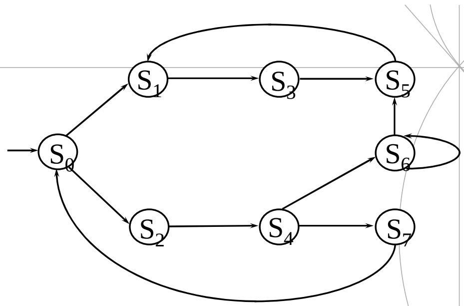
- **Breadth-First Visit**

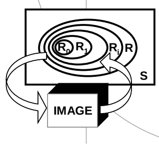
  EFFICIENT IFF we can deals with multiple states (set of state)



- **Breadth-First Visit**

  EFFICIENT IFF we can deals with multiple states (set of state)

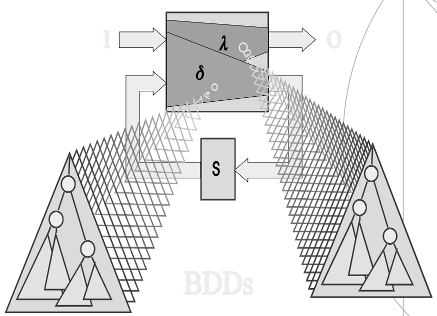




Outline

- **Background**
  - **FSM Model and State Space Graph Representation**
  - **State Space Visit: DFS and BFS Paradigms**
  - **Functions and Sets Representation**
  - **Image Computation Concepts**
  - **Impact and Reference**
- **Transition Relation**
- **Image Computation**
- **Reachability Analysis**
- **Limits**

Function Representation



Set Representation

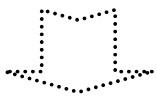
- **Idea**

  A formula can represent a set of states (its models)

- **Example**

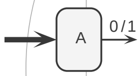
  $(x \oplus y) \oplus z$

  represents {100,010,110,111}

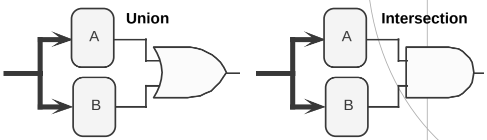


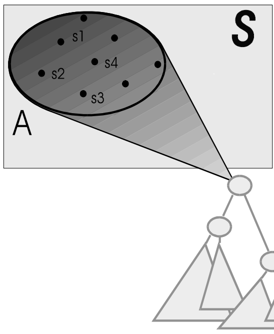
Characteristic Function

- $A \subseteq \{0,1\}^n$
  (Set of bit vectors of length *n*)
- **Represent set *A* as Boolean function** A **of *n* variables**
  ***X* ∈ *A* if and only if A(*X*) = 1**



Set Operations





Characteristic Function of set A:

$\chi_A(s)$ = 1 IFF s ∈ A
= 0 IFF s ∉ A

Outline

- **Background**
  - **FSM Model and State Space Graph Representation**
  - **State Space Visit: DFS and BFS Paradigms**
  - **Functions and Sets Representation**
  - **Image Computation Concepts**
  - **Impact and Reference**
- **Transition Relation**
- **Image Computation**
- **Reachability Analysis**
- **Limits**

Image and inverse image

Img (f, X) = f (X) = { y ∈ B^m | x ∈ X ∧ y = f (x) }

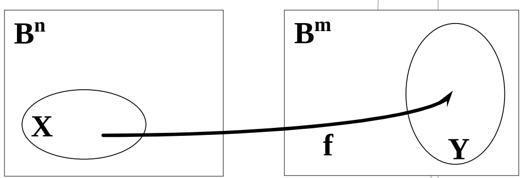
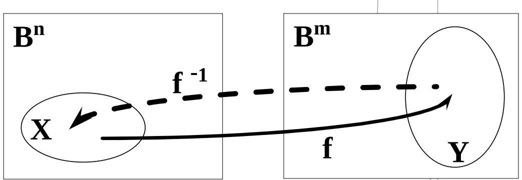


Image and inverse image

Img (f, X) = f (X) = { y ∈ B^m | x ∈ X ∧ y = f (x) }

PreImg (f, Y) = f^{-1}(Y) = { x ∈ B^n | y ∈ Y ∧ y = f (x) }



FSM Analysis Impact

- **Systems Represented as Finite State Machines**
  - **Sequential circuits**
  - **Communication protocols**
  - **Synchronization programs**
- **Analysis Tasks**
  - **State reachability**
  - **State machine comparison**
  - **Temporal logic model checking**
- **Traditional Methods Impractical for Large Machines**
  - **Polynomial in number of states**
  - **Number of states exponential in number of state variables**
  - **Example: single 32-bit register has 4,294,967,296 states!**

A Few Related Works

- Ranjan & co. IWLS-1995:
  - Clustering and ordering heuristics most widely used
- Hojati & co. ICCD-1996:
  - Theoretic results
- Moon & co. DAC-2000 & FMCAD-2000:
  - Transition function VS transition relation
  - Active lifetime – dependence matrix
- Gupta & co. FMCAD-2000:
  - BDD and SAT for computing images in traversal
- Meinel & co. FMCAD-2000, ICCD-2001:
  - Using hierarchical information for conjunction scheduling

Outline

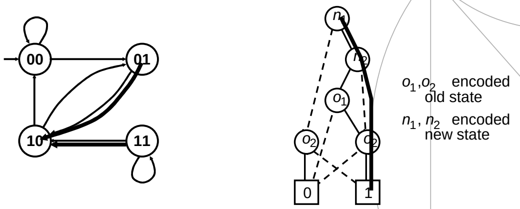
- Background
  - FSM Model and State Space Graph Representation
  - State Space Visit: DFS and BFS Paradigms
  - Functions and Sets Representation
  - Image Computation Concepts
  - Impact and Reference
- Transition Relation
- Image Computation
- Reachability Analysis
- Limits

The Transition Relation

$$TR(s, x, y) = \prod_{i=1}^{n} (y_i \equiv \delta_i (s, x))$$

The Transition Relation espresses
present-state, primary input ⇒ next state correspondence.

Deterministic FSM **Symbolic Representation**



- Represent set of transitions as function δ(*Old*, *New*)
  - Yields 1 if can have transition from state ***Old*** to state ***New***
- Represent as Boolean function
  - Over variables encoding states

TR(s,x,y) =

$= \prod_{i=1}^{n}(y_i \equiv \delta_i (s,x))$

TR(s,x,y) =

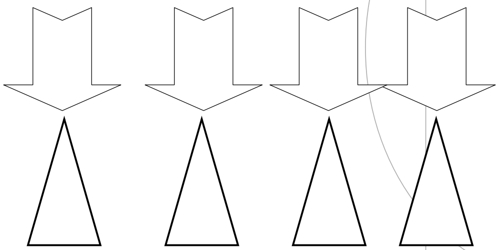
$= \prod_{i=1}^{n}(y_i \equiv \delta_i (s,x))$

$= [(y_1 \equiv \delta_1 (s,x)) \cdot (y_2 \equiv \delta_2 (s,x)) \cdot \dots \cdot (y_n \equiv \delta_n (s,x))]$

TR(s,x,y) =

$= \Pi_{i=1}^{n}(y_i \equiv \delta_i\ (s,x))$

$= [\ (y_1 \equiv \delta_1\ (s,x))\ \cdot\ (y_2 \equiv \delta_2\ (s,x))\ \cdot\ \ldots\ \cdot\ (y_n \equiv \delta_n\ (s,x))\ ]$



Outline

- Background
  - FSM Model and State Space Graph Representation
  - State Space Visit: DFS and BFS Paradigms
  - Functions and Sets Representation
  - Image Computation Concepts
  - Impact and Reference
- Transition Relation
- Image Computation
- Reachability Analysis
- Limits

Image Computation

$\mathrm{Img}\ (TR, From) = \exists_{s,x}\ [TR\ (s, x, y) \cdot From(s)]$

Image is computed through:

a conjunction-abstraction operation between present state set and transition relation.

Image Computation

$\mathrm{Img}\ (TR, From) = \exists_{s,x}\ [TR\ (s, x, y) \cdot From(s)]$

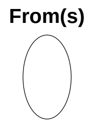


Image Computation

$\mathrm{Img}\ (TR, From) = \exists_{s,x}\ [TR\ (s, x, y) \cdot From(s)]$

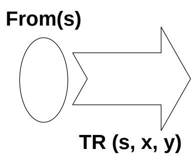


Image Computation

$\mathrm{Img}\ (TR, From) = \exists_{s,x}\ [TR\ (s, x, y) \cdot From(s)]$

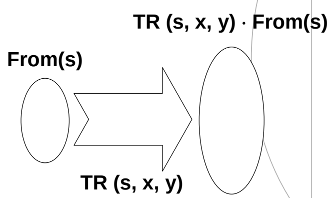


Image Computation

Img (TR, From) = $\exists_{s,x}$ [TR (s, x, y) · From(s)]

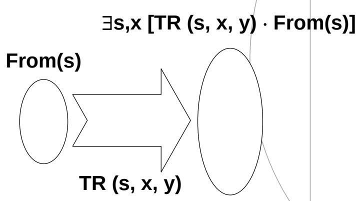


Image Computation

Img (TR, From) = $\exists_{s,x}$ [TR (s, x, y) · From(s)]

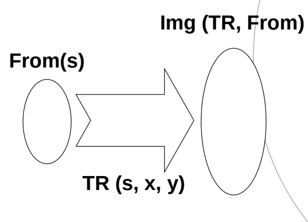
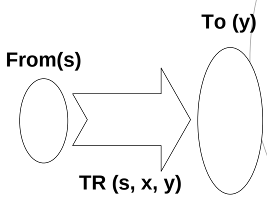


Image Computation

To (y) = Img (TR, From) = $\exists_{s,x}$ [TR (s, x, y) · From(s)]



To (y) =

= $\exists_{sx}$[TR (s,x,y) · From (s)]

To (y) =

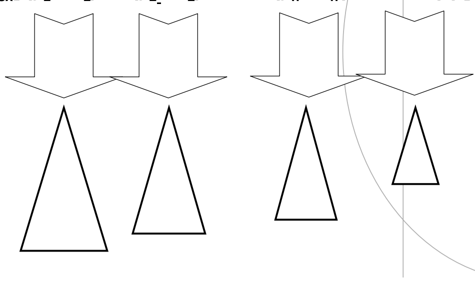
= $\exists_{sx}$[TR (s,x,y) · From (s)]

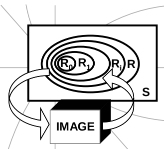
= $\exists_{sx}$[$(y_1 \equiv \delta_1)$ · $(y_2 \equiv \delta_2)$ · … · $(y_n \equiv \delta_n)$ · From (s)]

To (y) =

= $\exists_{sx}$[TR (s,x,y) · From (s)]

= $\exists_{sx}$[$(y_1 \equiv \delta_1)$ · $(y_2 \equiv \delta_2)$ · … · $(y_n \equiv \delta_n)$ · From (s)]





To (y) =

$= \exists_{sx}[\, TR (s,x,y) \cdot From (s)\,]$

$= \exists_{sx}[\, (y_1 \equiv \delta_1) \cdot (y_2 \equiv \delta_2) \cdot \ldots \cdot (y_n \equiv \delta_n) \cdot From (s)\,]$

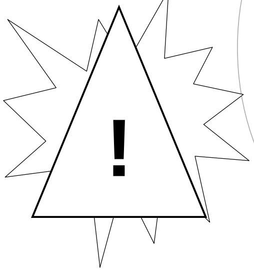
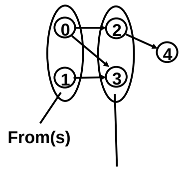


Image and Pre-Image of States: An Example

Image of a set of states From(s)

Example:



From (s) =

$(s \equiv 0) \vee (s \equiv 1)$	{0,1}

TR (s, y) =

$(s \equiv 0) \wedge (y \equiv 2) \vee$	{(0,2),
$(s \equiv 0) \wedge (y \equiv 3) \vee$	(0,3),
$(s \equiv 1) \wedge (y \equiv 3) \vee$	(1,3),
$(s \equiv 2) \wedge (y \equiv 4)$	(2,4)}

TR (s, y) $\wedge$ From (s) =

$(s \equiv 0) \wedge (y \equiv 2) \vee$	{(0,2),
$(s \equiv 0) \wedge (y \equiv 3) \vee$	(0,3),
$(s \equiv 1) \wedge (y \equiv 3)$	(1,3)}

To (y) = $\exists$s (TR $\wedge$ From) =

$(y \equiv 2) \vee (y \equiv 3)$	{(2,3)}

Outline

- Background
  - FSM Model and State Space Graph Representation
  - State Space Visit: DFS and BFS Paradigms
  - Functions and Sets Representation
  - Image Computation Concepts
  - Impact and Reference
- Transition Relation
- Image Computation
- Reachability Analysis
- Limits

Representations

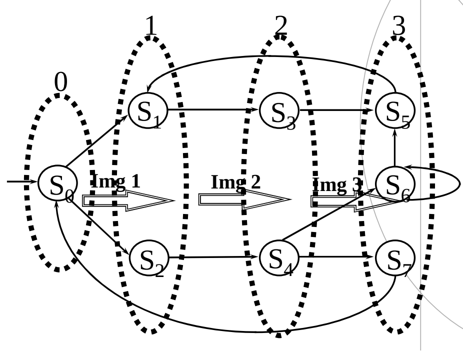
- **Explicit reachability analysis**
  - **Represent states explicitly (e.g. as bit string) => limited capacity**
  - **Use hashtable to find quickly whether state was reached before**
  - **Image operation: simple simulation**
  - **Preimage operation: SAT run**
- **Symbolic reachability analysis**
  - **Represent states and transition relation symbolically**
    - **E.g. BDDs, circuits, DNF, etc.**
  - **Use BDD operations to perform image and preimage operation (simple AND or AND_EXIST)**
  - **Lots of heuristic improvements to keep BDD size under control**

State Traversal Techniques

- **Forward Traversal**
  - **Start from initial state(s)**
  - **Traverse forward to check whether "bad"**
  - **State(s) is reachable**
- **Backward Traversal**
  - **Start from bad state(s)**
  - **Traverse backward to check whether intial**
  - **State(s) can reach them**
- **Combines Forward/Backward traversal**
  - **compute over-approximation of reachable**
  - **states by forward traversal**
  - **for all bad states in over-approximation, start backward traversal to see whether intial state can reach them**

Forward Reachability Analysis (Forward Traversal)

Sequence of image computations … till fixed point …



FwdTraversal **(TR, S_0)**

Reached = From = New = S_0 (s)

while (New $\neq \phi$)

To = *Img* (TR, From)

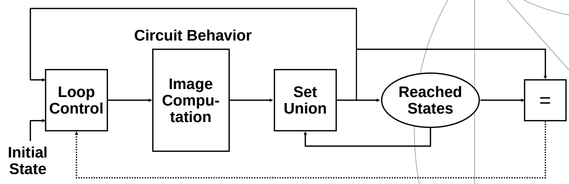
To $|_{y \rightarrow s}$

New = To $\wedge \neg$Reached

Reached = Reached $\vee$ New

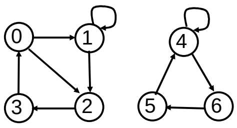
From = *Best_BDD* (New, Reached)

return (Reached (s))



- **Determine set of all reachable states of circuit**
- **Key step in model checking**
  - **Many (but not all) properties can be checked by some form of reachability computation**

- **Example**



Iteration:	**1**	**2**	**3**
From:	{0}	{1,2}	{1,2,3}
To:	{1,2}	{1,2,3}	{0,1,2,3}
Reached:	{0}	{0,1,2}	{0,1,2,3}

Backward State Traversal

BwdTraversal **(TR, S_0)**

Reached = From = New = S_0 (s)

while (New $\neq \phi$)

To = *PreImg* (TR, From)

To $|_{y \rightarrow s}$

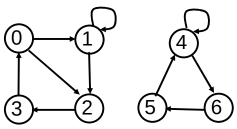
New = To $\wedge \neg$Reached

Reached = Reached $\vee$ New

From = *Best_BDD* (New, Reached)

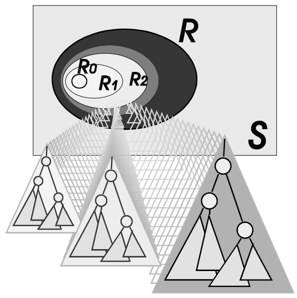
return (Reached (s))

- **Example**



Iteration:	**1**	**2**	**3**
From (current):	{6}	{4}	{4,5}
To (previous):	{4}	{4,5}	{4,5,6}
Reached:	{6}	{4,6}	{4,5,6}

To summarise



Forward Traversal

R_0 = *Initial State Set*

R_{i+1} = R_i + Img (TR, R_i)

Backward Traversal

R_0 = *Initial State Set*

R_{i+1} = R_i + PreImg (TR, R_i)

Outline

- ❖ Background
  - FSM Model and State Space Graph Representation
  - State Space Visit: DFS and BFS Paradigms
  - Functions and Sets Representation
  - Image Computation Concepts
  - Impact and Reference
- ❖ Transition Relation
- ❖ Image Computation
- ❖ Reachability Analysis
- ❖ Limits

Maximum Size at intermediate steps [Ravi & Somenzi, ICCAD'95]

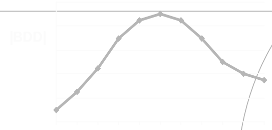


Image steps Traversal steps

Maximum Size at intermediate steps [Ravi & Somenzi, ICCAD'95]