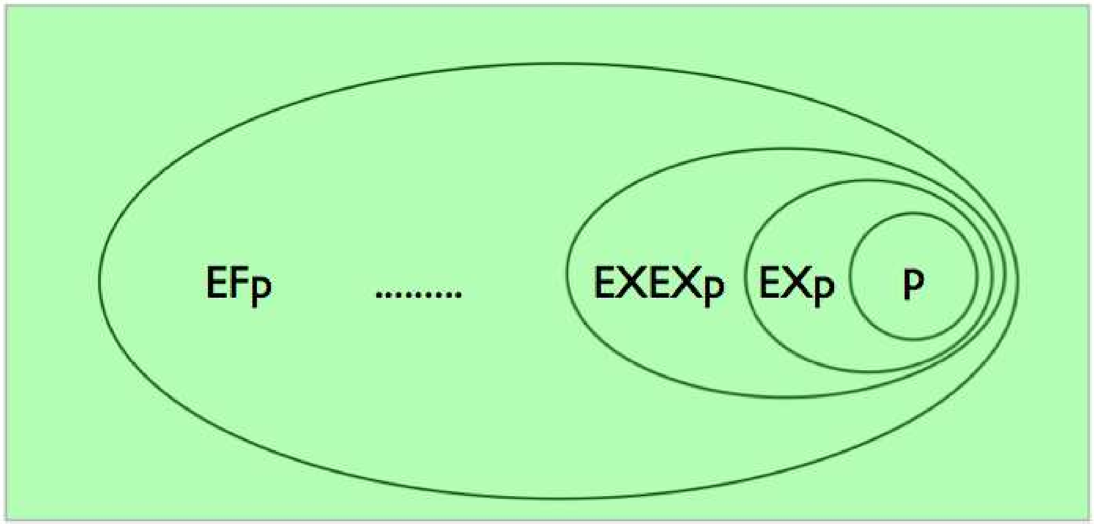
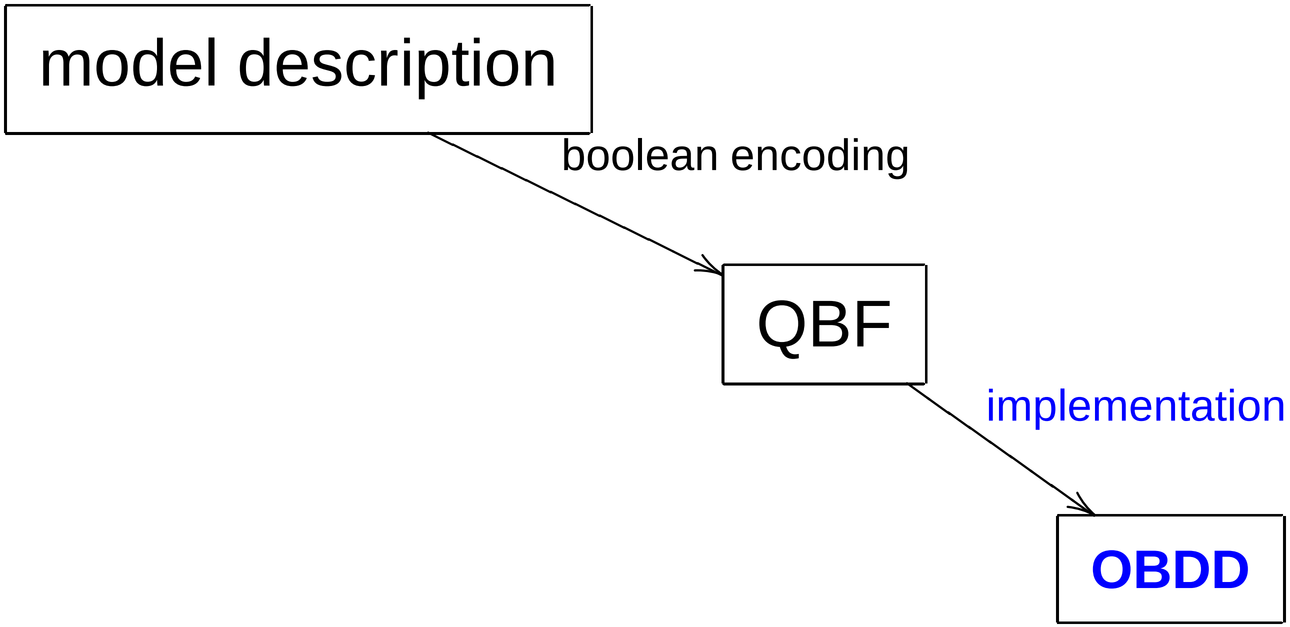
Computer aided verification

Lecture 6: Symbolic verification I

Idea: EF p



Symbolic model checking



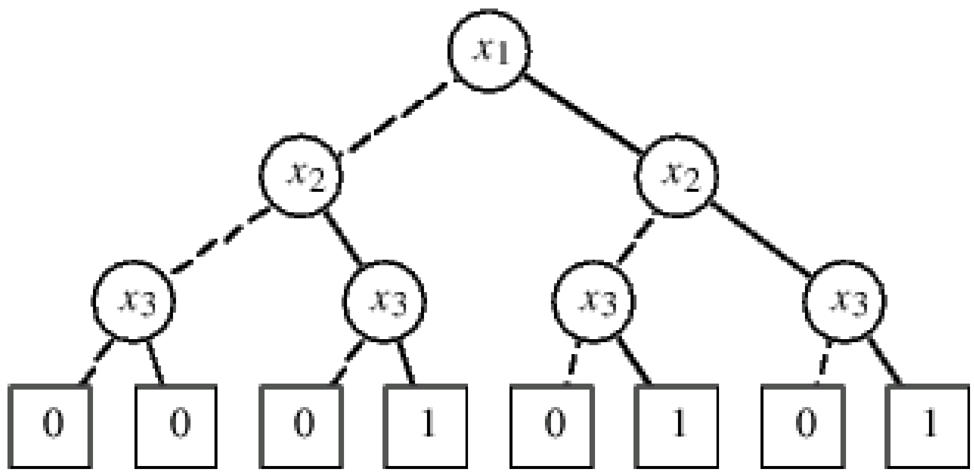
model checking = operations on OBDDs

I. OBDD

OBDD

Ordered Binary Decision Diagrams

x_1	x_2	x_3	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1



[Bryant 1992]

- $f : \{0,1\}^3 \rightarrow \{0,1\}$
- fixed order on variables: $x_1 < x_2 < x_3$

OBDD = rooted acyclic graph

Attributes of a node v:

- when v is a leaf
  - $\mathrm{val}(v) \in \{0, 1\}$
- when v is not a leaf
  - $\mathrm{var}(v) \in \{x_1, x_2, \ldots\}$
  - $\mathrm{lo}(v), \mathrm{hi}(v)$ – 2 nodes

The order of variables must be obeyed on every path.

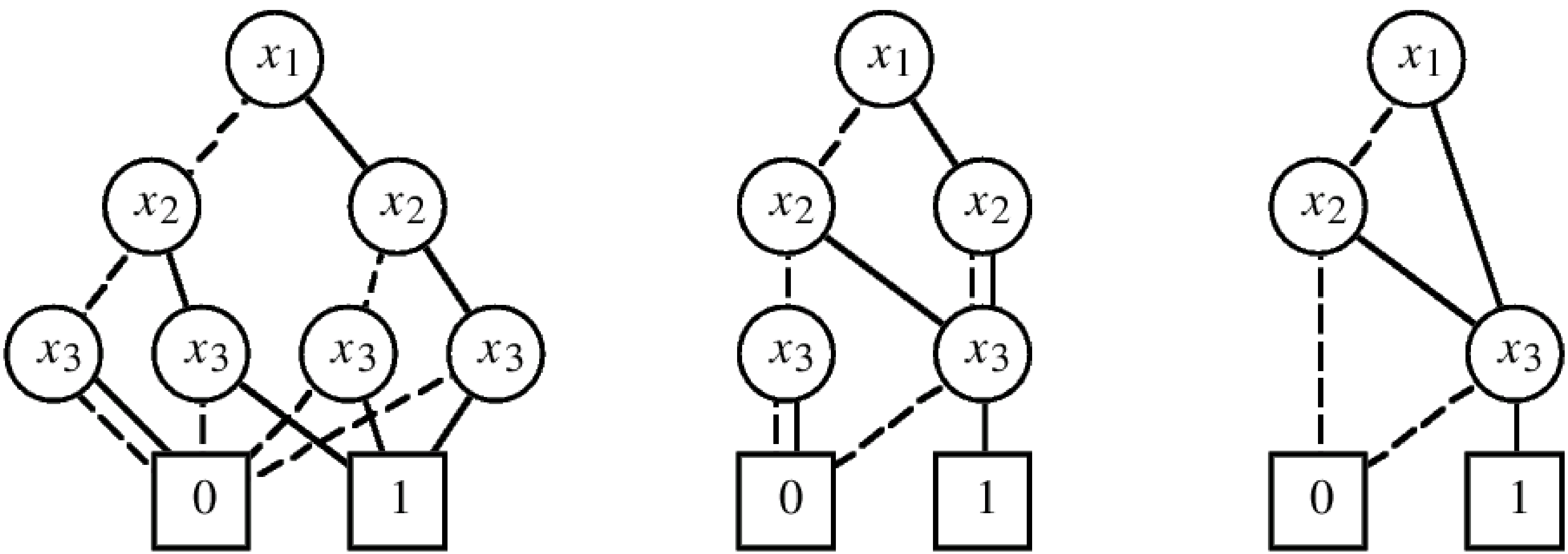
Simplifying OBDD

– remove redundant leaves

– remove redundant non-leaves

– remove redundant tests

OBDD $\longmapsto$ ROBDD



[Bryant 1992]

Canonical form: reduced OBDD

Canonical form for a boolean function:

– independent from the initial OBDD

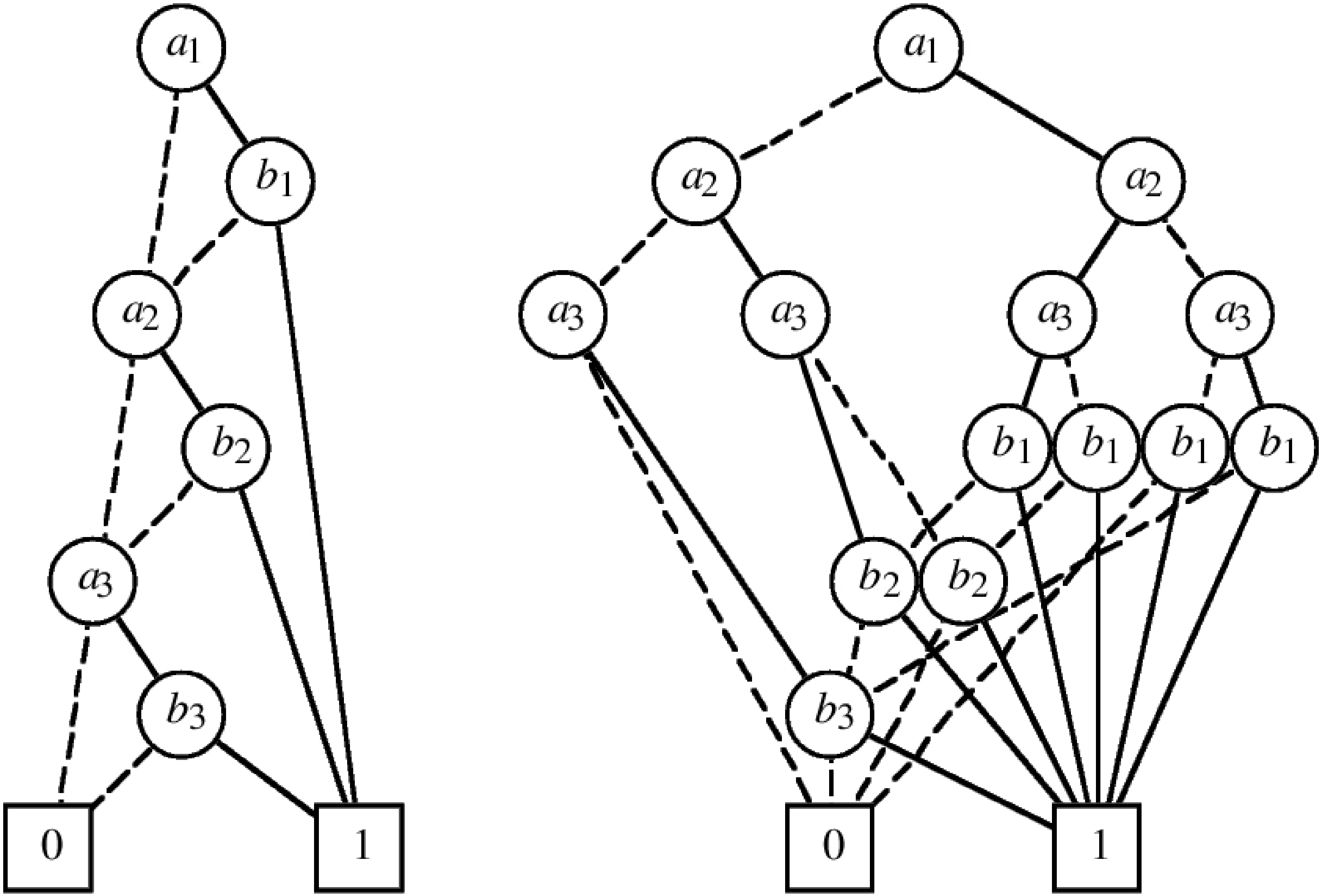
– (strongly) dependent on the order of variables

Naive construction of an OBDD for a boolean formula ϕ:

$$\phi \longmapsto \text{decision tree} \longmapsto \text{canonical OBDD}$$

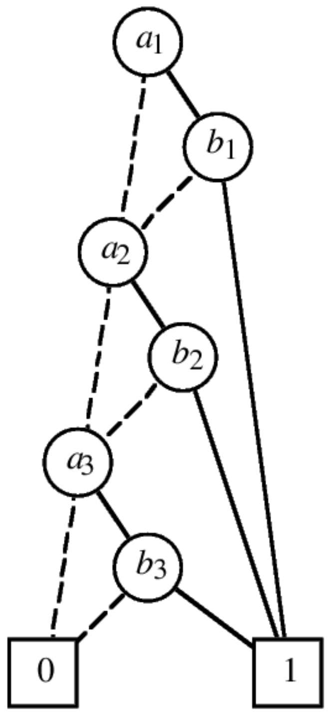
Finding an appropriate order of variables is crucial!

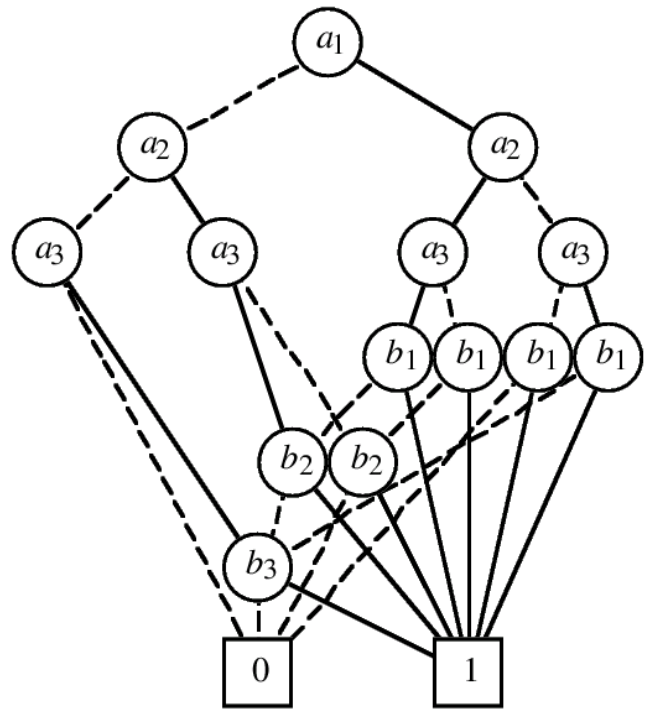
$a_1 \wedge b_1 \ \vee \ a_2 \wedge b_2 \ \vee \ a_3 \wedge b_3$



[Bryant 1992]

$$f(a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_n) = a_1 \wedge b_1 \;\vee\; a_2 \wedge b_2 \;\vee\; \ldots \;\vee\; a_n \wedge b_n$$





$2 \cdot n$

$2 \cdot (2^n - 1)$

Heuristic: closely related variables should be close in the order

example of a boolean function	lower bound	upper bound
symmetric functions	$\mathcal{O}(n)$	$\mathcal{O}(n^2)$
addition (oldest bits)	$\mathcal{O}(n)$	$\mathcal{O}(2^n)$
multiplication (middle bits)	$\mathcal{O}(2^n)$	$\mathcal{O}(2^n)$

Shannon's expansion

$$f \quad = \quad \neg x \wedge f|_{x \leftarrow 0} \quad \vee \quad x \wedge f|_{x \leftarrow 1}$$

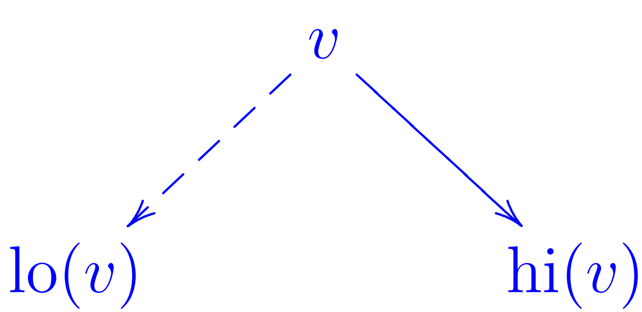
$$f|_{x_i \leftarrow b}(x_1, \ldots, x_n) \quad = \quad f(x_1, \ldots, x_{i-1}, b, x_{i+1}, \ldots, x_n)$$

Shannon's expansion

$$f \;=\; \neg x \wedge f|_{x\leftarrow 0} \;\vee\; x \wedge f|_{x\leftarrow 1}$$

$$f|_{x_i\leftarrow b}(x_1,\ldots,x_n) \;=\; f(x_1,\ldots,x_{i-1},b,x_{i+1},\ldots,x_n)$$

$$v \;=\; \neg x \wedge \mathrm{lo}(v) \;\vee\; x \wedge \mathrm{hi}(v)$$



$x = \mathrm{var}(v)$

OBDDs as an abstract data type

the order of variables the same in all OBDDs

Operations:

$f \vee g$, $f \wedge g$, $\neg f$, `false`, `true` $\quad BF \mapsto OBDD$

$f|_{x \leftarrow 0}$, $f|_{x \leftarrow 1}$

$\exists x.\ f$, $\forall x.\ f$ $\quad QBF \mapsto OBDD$

$f = g$

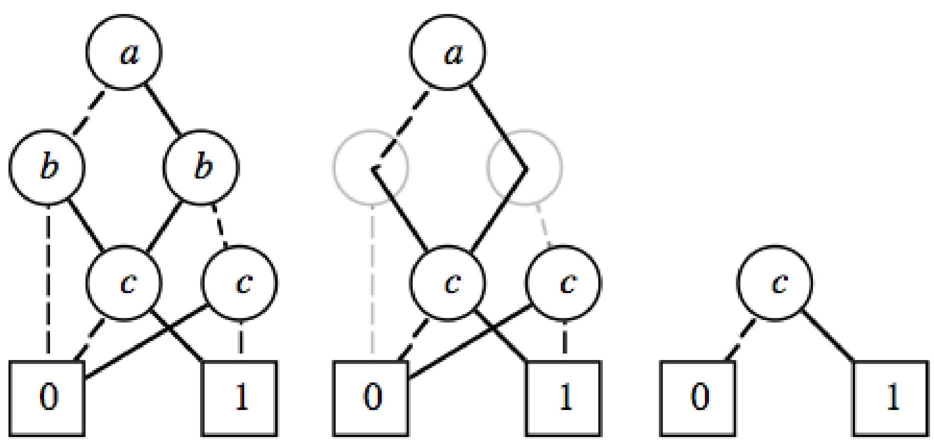
Note: operations on **functions**, not on values $\{0, 1\}$.

Implementation of unary operations

- $f|_{x \leftarrow b}$

  traverse nodes n of OBDD representing f:

  - if $x > \mathrm{var}(n)$ then recursively traverse $\mathrm{lo}(n)$ and $\mathrm{hi}(n)$;
  - if $x < \mathrm{var}(n)$ then stop;
  - otherwise replace n by: $\begin{cases} \mathrm{lo}(n) & \text{if } b = 0 \\ \mathrm{hi}(n) & \text{if } b = 1 \end{cases}$



[Bryant 1992]

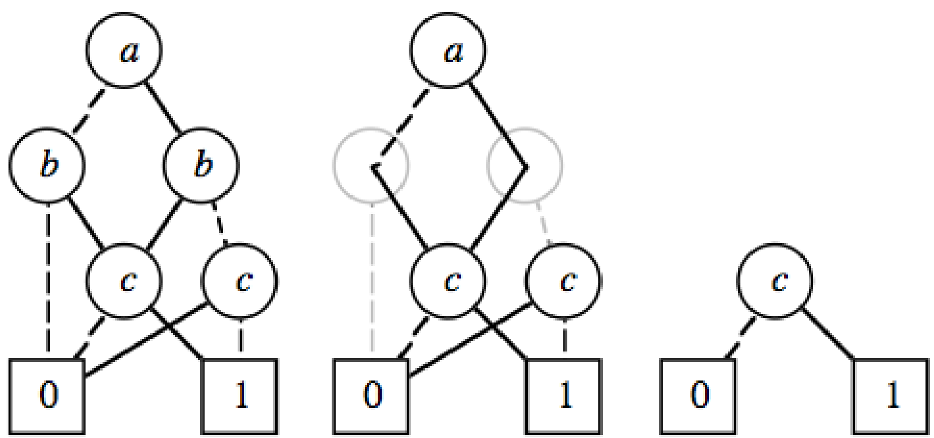
Implementation of unary operations (cont.)

– $\exists x.\ f \quad = \quad f|_{x \leftarrow 0} \vee f|_{x \leftarrow 1}$

the order of variables

remains the same!

– $\neg f$ **?**



[Bryant 1992]

Implementation of binary operations

the order of variables the same in all OBDDs

$\bullet : \{0,1\}^2 \to \{0,1\}$

$$f \bullet g \;=\; \neg x \wedge (f \bullet g)|_{x\leftarrow 0} \quad \vee \quad x \wedge (f \bullet g)|_{x\leftarrow 1}$$
$$f \bullet g \;=\; \neg x \wedge (f|_{x\leftarrow 0} \bullet g|_{x\leftarrow 0}) \quad \vee \quad x \wedge (f|_{x\leftarrow 1} \bullet g|_{x\leftarrow 1})$$

$\text{Apply}(\ f,\ g,\ \bullet\ )$ (think of f, g as roots of OBDDs)

- f, g leaves: $\text{val}(f \bullet g) \ = \ \text{val}(f) \bullet \text{val}(g)$
- f leaf, g not: $f \bullet g \ = \ op(g)$
- $\text{var}(f) = \text{var}(g) = x$:

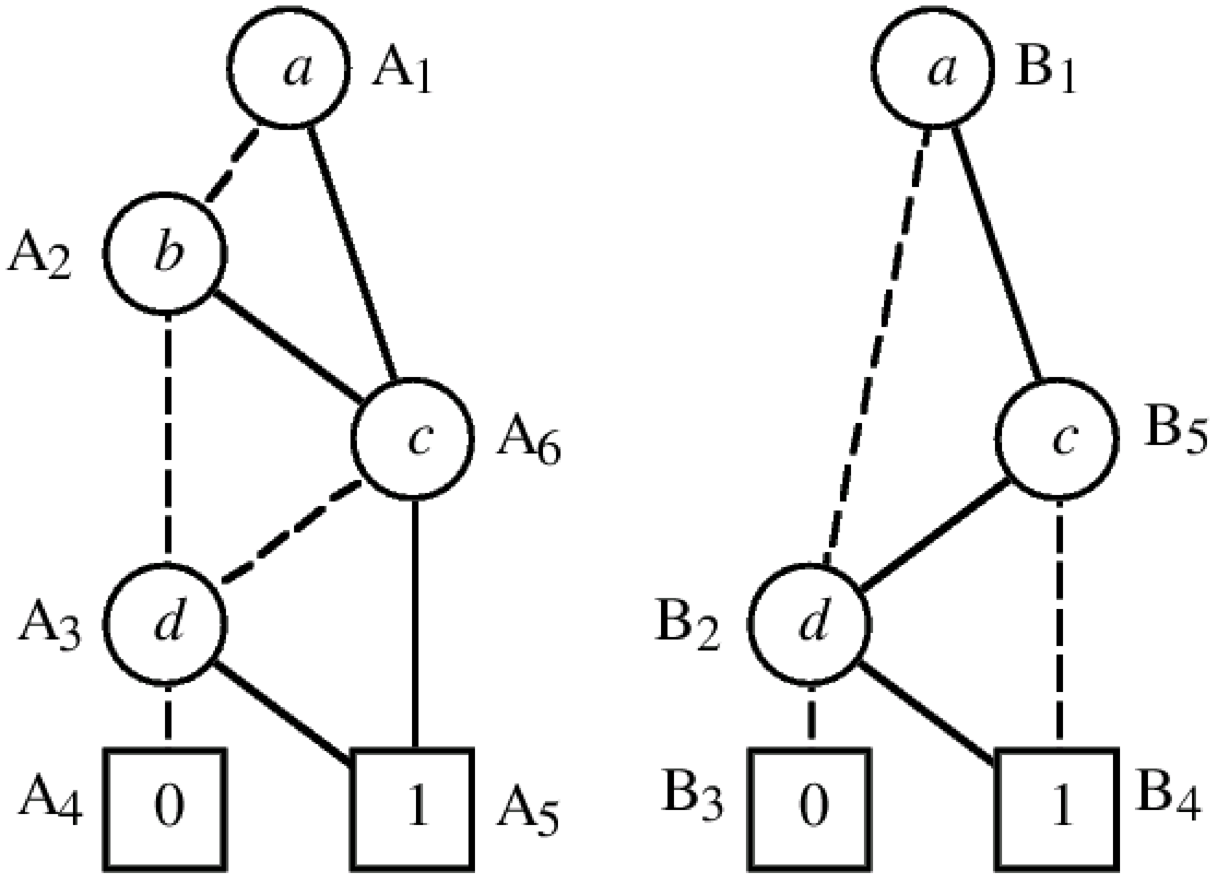
$$\text{lo}(f \bullet g) = \text{lo}(f) \bullet \text{lo}(g) \qquad \text{hi}(f \bullet g) = \text{hi}(f) \bullet \text{hi}(g)$$

- $\text{var}(f) = x < y = \text{var}(g)$:

$$\text{lo}(f \bullet g) = \text{lo}(f) \bullet g \qquad \text{hi}(f \bullet g) = \text{hi}(f) \bullet g$$

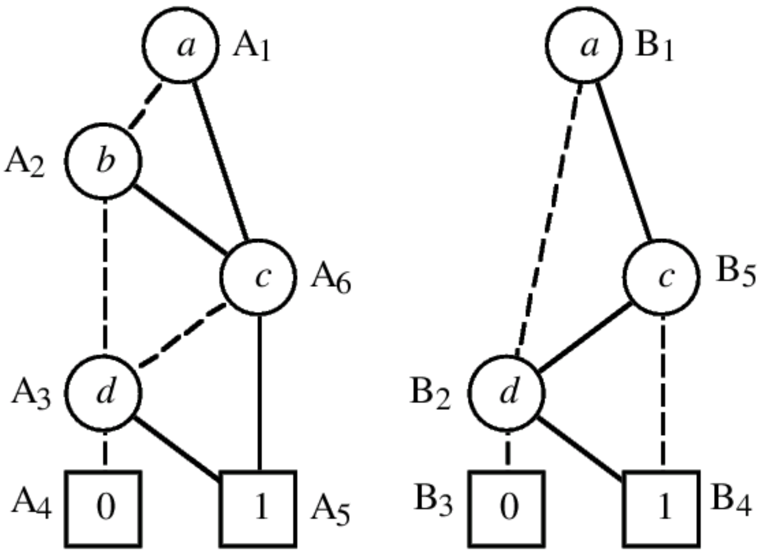
$$f \bullet g \ = \ \neg x \wedge (f \bullet g)|_{x \leftarrow 0} \quad \vee \quad x \wedge (f \bullet g)|_{x \leftarrow 1}$$
$$f \bullet g \ = \ \neg x \wedge (f|_{x \leftarrow 0} \bullet g|_{x \leftarrow 0}) \quad \vee \quad x \wedge (f|_{x \leftarrow 1} \bullet g|_{x \leftarrow 1})$$

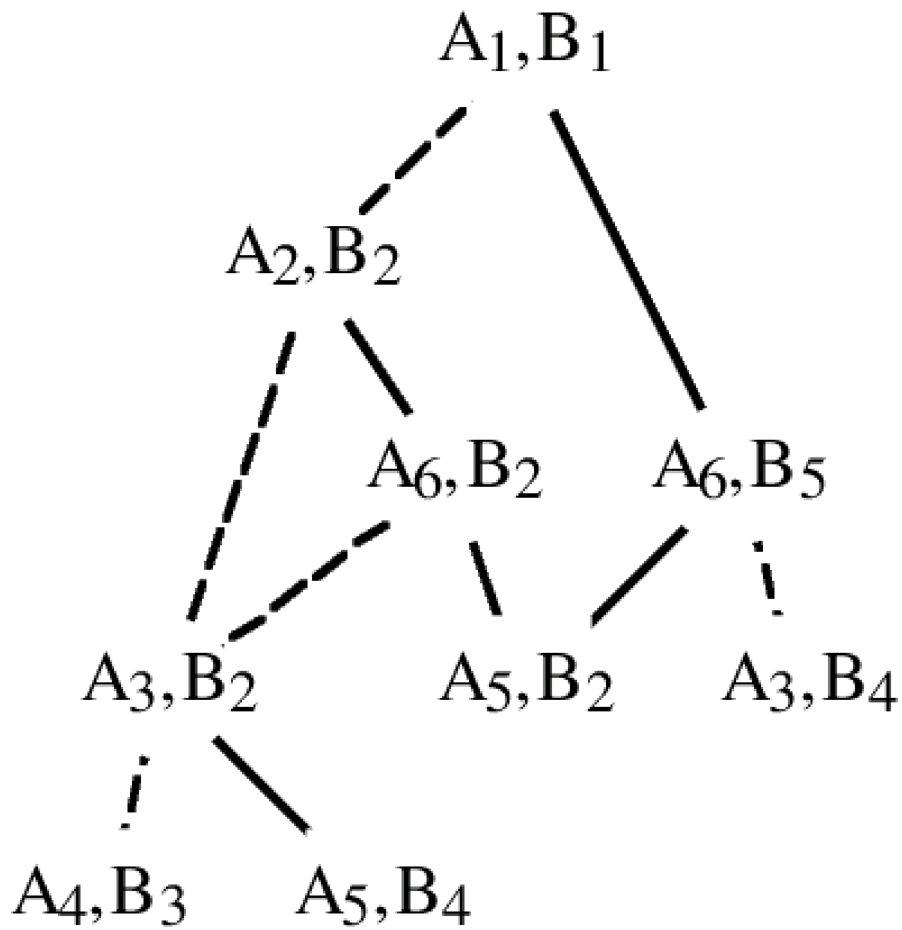
Example: input OBDDs



[Bryant 1992]

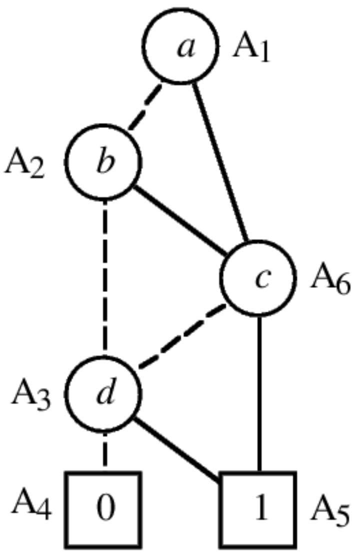
Example: recursive calls



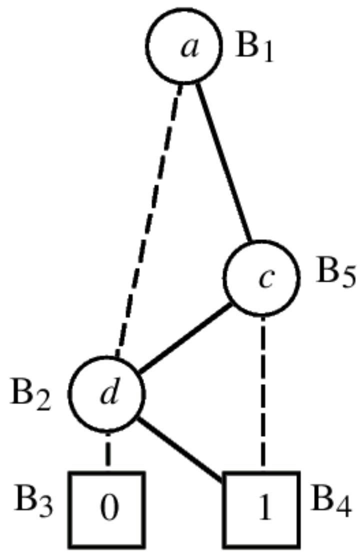


[Bryant 1992]

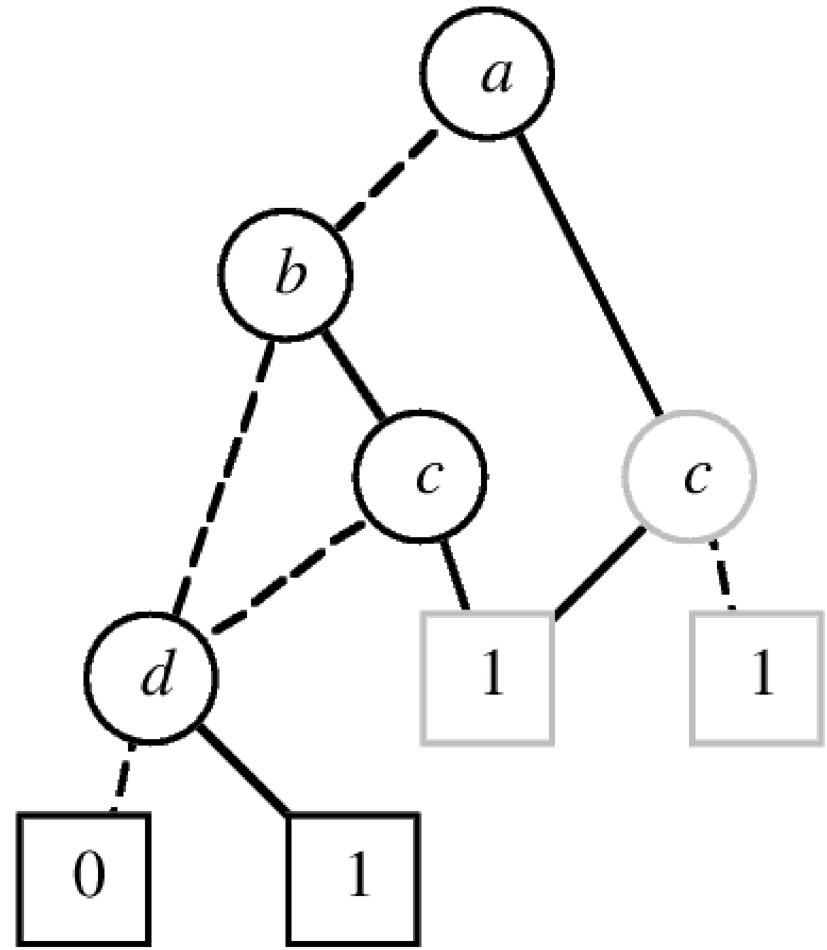
Example: result = $a \vee b \wedge c \vee d$



$(a \vee b) \wedge c \vee d$

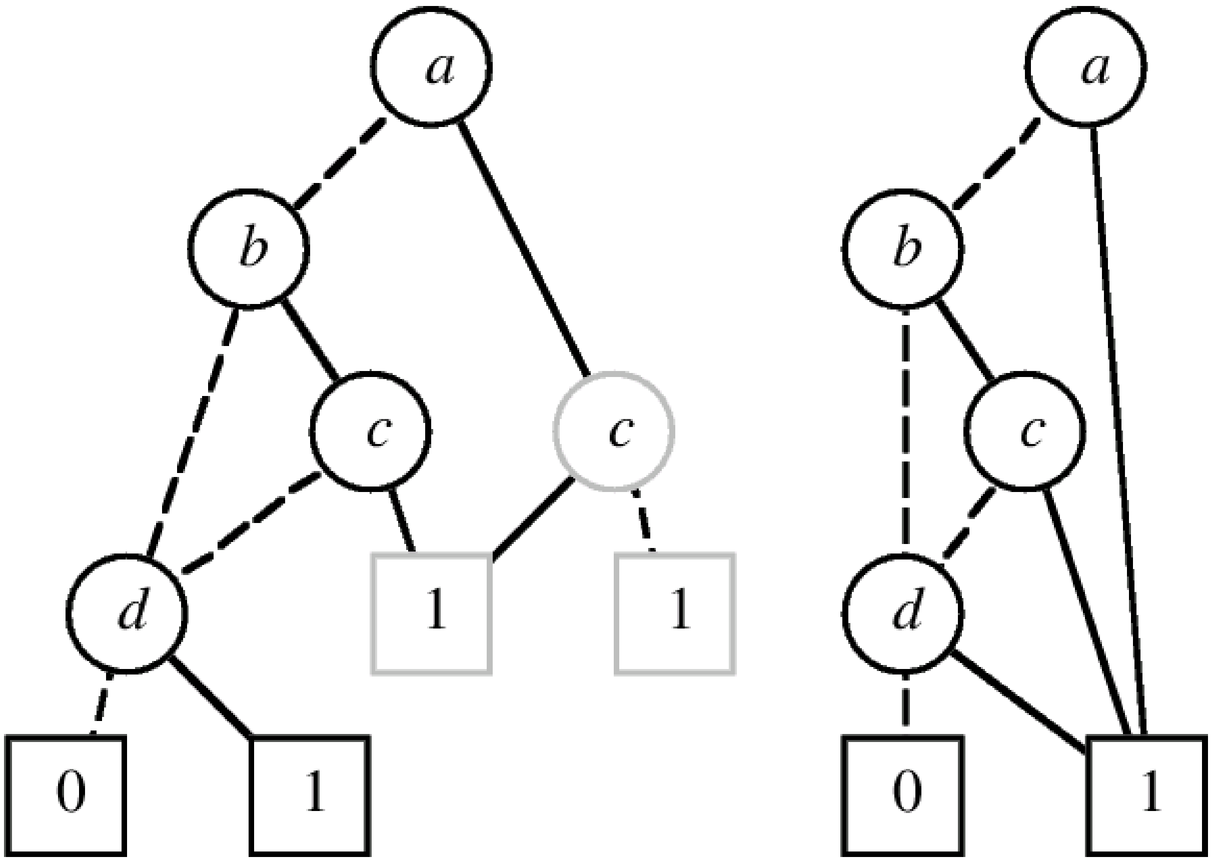


$a \wedge \neg c \vee d$



[Bryant 1992]

Example: reduced result = $a \vee b \wedge c \vee d$



[Bryant 1992]

Implementation of binary operations

$\text{Apply}(\ f,\ g,\ \bullet\ )$

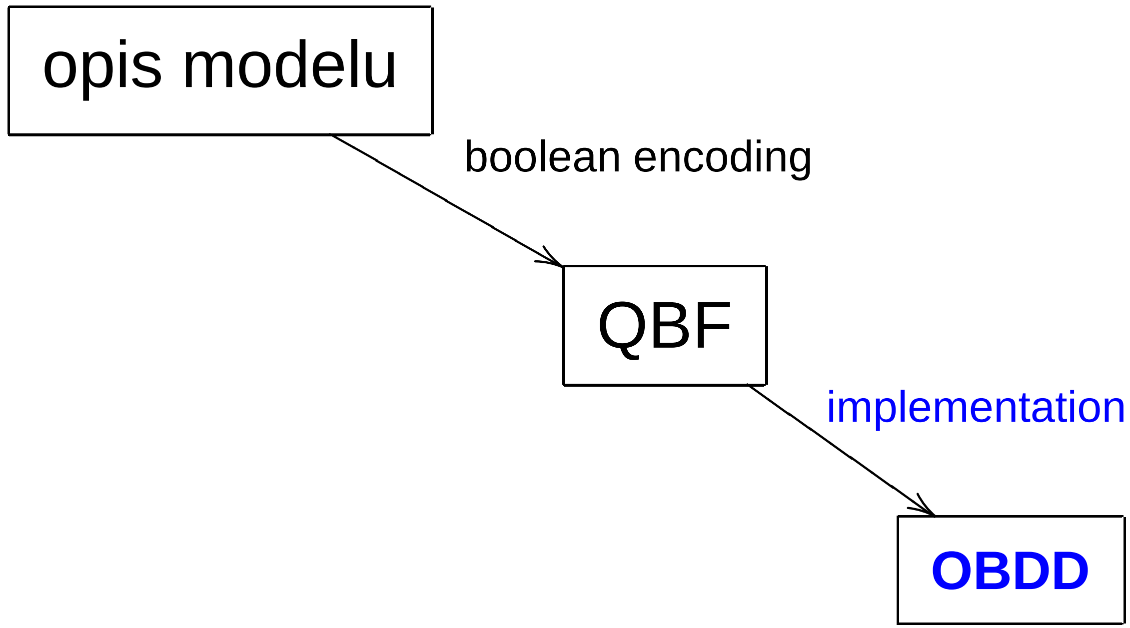
- running time: $\mathcal{O}(|f| \cdot |g|)$
- result in the canonical form

Question: $f \iff g$ **?** $f = g$ **?**

- one OBDD shared by all functions
  - $=$ in constant time
- edges representing $\neg$
- OZBDD
- ...

II. Boolean encoding

Boolean encoding



model checking = operations on OBDDs

Model description

– S described by m boolean variables: $S \equiv \{0,1\}^m$

– transition relation $R : \{0,1\}^m \times \{0,1\}^m \to \{0,1\}$

$$R(x_1, \ldots, x_m, x'_1, \ldots, x'_m) \in \{0,1\}$$

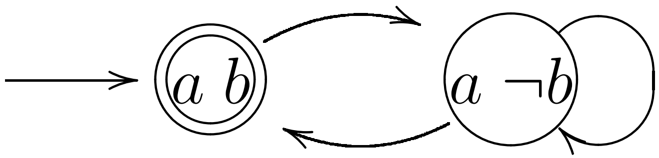
– initial states $S_0 : \{0,1\}^m \to \{0,1\}$

$$S_0(x_1, \ldots, x_m) \in \{0,1\}$$

– atomic properties $L_p = \{s \mid p \in L(s)\} : \{0,1\}^m \to \{0,1\}$

$$L_p(x_1, \ldots, x_m) \in \{0,1\}$$

Example



(reachable states)

$$
\begin{aligned}
R &= (a \wedge b \wedge a' \wedge \neg b') \vee (a \wedge \neg b \wedge a' \wedge \neg b') \vee (a \wedge \neg b \wedge a' \wedge b') \\
S_0 &= a \wedge b \\
L_p &= b
\end{aligned}
$$

From a model to OBDD

description of a Kripke structure

↓

Kripke structure ⟼ OBDD

NO!

From a model to OBDD

description of a Kripke structure ↦ Kripke structure ↦ OBDD **NO!**

description of a Kripke structure ↦ OBDD **YES!**

Compositional model desciption

Synchronous processes:

$$R \;=\; R_1 \wedge R_2 \wedge \ldots \wedge R_n$$

Asynchronous processes (interleaving):

$$R \;=\; R'_1 \vee R'_2 \vee \ldots \vee R'_n$$

$$R'_i \;=\; R_i \wedge (\textstyle\bigwedge_{j \neq i} \mathrm{Id}_j)$$

Asynchronous processes (simultaneous execution):

$$R \;=\; R'_1 \wedge R'_2 \wedge \ldots \wedge R'_n$$

$$R'_i \;=\; R_i \vee \mathrm{Id}_i$$

Restriction to reachable states

$$\widehat{R}(x_1, \ldots, x_m, x'_1, \ldots, x'_m) = 1$$

$$\Updownarrow$$

$$R(x_1, \ldots, x_m, x'_1, \ldots, x'_m) \;\; \wedge$$

$(x_1, \ldots, x_m), (x'_1, \ldots, x'_m)$ reachable

Restriction to reachable states

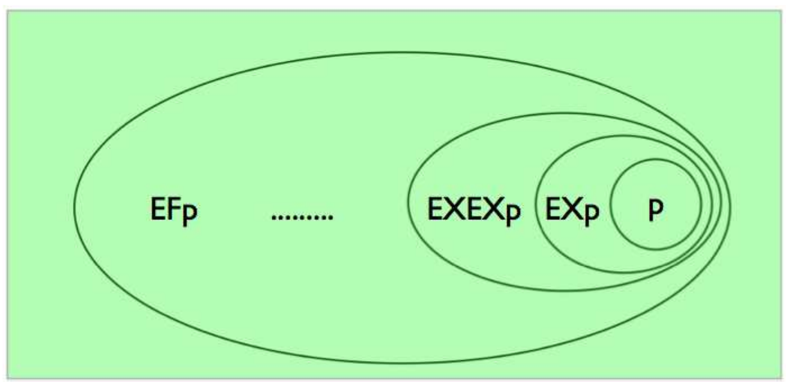
$$\widehat{R}(x_1, \ldots, x_m, x'_1, \ldots, x'_m) = 1$$

$$\Updownarrow$$

$$R(x_1, \ldots, x_m, x'_1, \ldots, x'_m) \ \wedge$$

$(x_1, \ldots, x_m)$ reachable

III. Symbolic verification



Fixed points in a complete lattice $\langle A, \leq \rangle$.

Let $f : A \to A$ monotonic.

- the least f.p.: $\bot \leq f(\bot) \leq f^2(\bot) \leq \ldots \rightsquigarrow \mu Z.\, f(Z)$
- the greatest f.p.: $\top \geq f(\top) \geq f^2(\top) \geq \ldots \rightsquigarrow \nu Z.\, f(Z)$

When A finite, the fixed points are reached after $\leq |A|$ iterations.

Fixed points in a complete lattice $\langle A, \leq \rangle$.

Let $f : A \to A$ monotonic.

- the least f.p.: $\bot \leq f(\bot) \leq f^2(\bot) \leq \ldots \;\rightsquigarrow\; \mu Z.\, f(Z)$
- the greatest f.p.: $\top \geq f(\top) \geq f^2(\top) \geq \ldots \;\rightsquigarrow\; \nu Z.\, f(Z)$

Example: $\langle A, \leq \rangle \;=\; \langle \mathcal{P}(S), \subseteq \rangle$

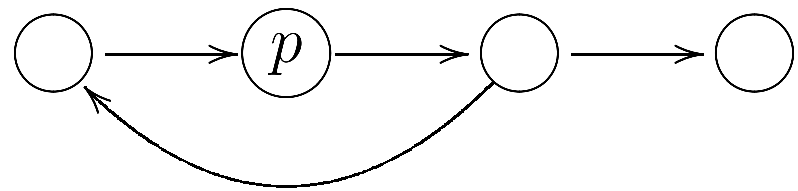
$Z \mapsto$ EX Z $\qquad \mu Z.$ EX $Z \;=\bot=\emptyset$ $\qquad \nu Z.$ EX $Z \;=$ **?**

$Z \mapsto p \vee$ EX Z $\qquad \mu Z.\, p \vee$ EX $Z \;=$ **?**

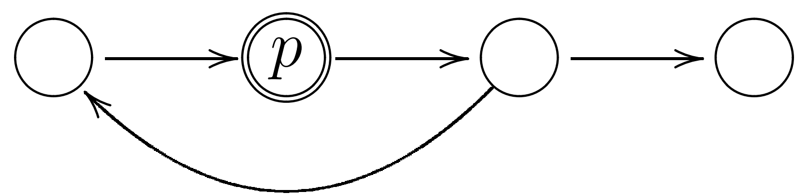
$\mathsf{EF}\, p \;=\; \mu Z.\, p \vee \mathsf{EX}\, Z$

$Z \mapsto p \vee \mathsf{EX}\, Z$

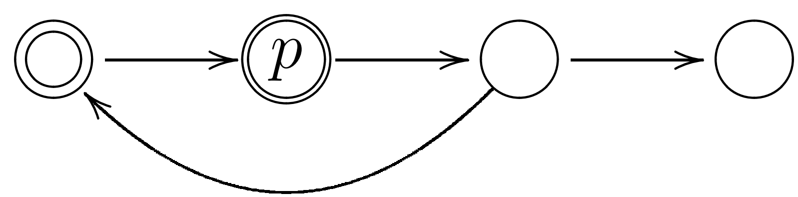
$\mathtt{false}$



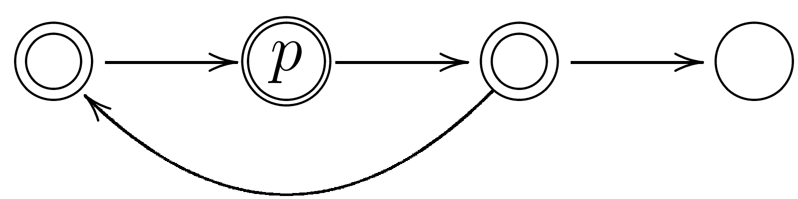
$p \vee \mathsf{EX}\,\mathtt{false} \equiv p$



$p \vee \mathsf{EX}\, p$



$p \vee \mathsf{EX}\,(p \vee \mathsf{EX}\, p)$

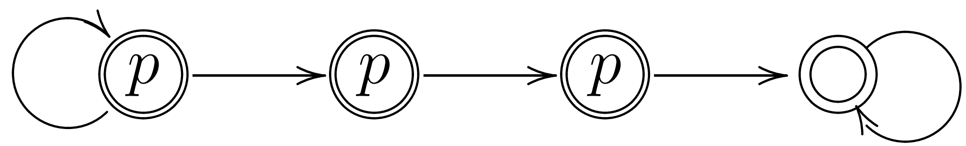


CTL via fixed points

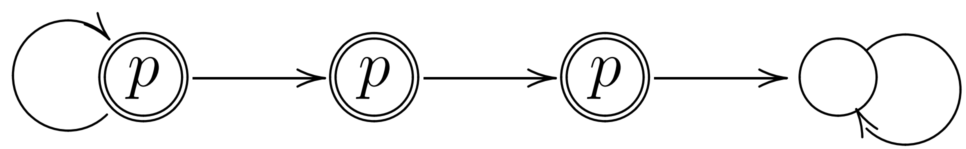
- $\mathsf{EF}\,\phi \;=\; \mu Z.\, \phi \vee \mathsf{EX}\, Z$ $\qquad Z \mapsto \phi \vee \mathsf{EX}\, Z$
- $\mathsf{AF}\,\phi \;=\; \mu Z.\, \phi \vee \mathsf{AX}\, Z$ $\qquad Z \mapsto \phi \vee \mathsf{AX}\, Z$
- $\mathsf{EG}\,\phi \;=\; \nu Z.\, \phi \wedge \mathsf{EX}\, Z$ $\qquad Z \mapsto \phi \wedge \mathsf{EX}\, Z$
- $\mathsf{AG}\,\phi \;=\; \nu Z.\, \phi \wedge \mathsf{AX}\, Z$ $\qquad Z \mapsto \phi \wedge \mathsf{AX}\, Z$
- $\mathsf{E}\,\phi\,\mathsf{U}\,\psi \;=\; \mu Z.\, \psi \vee (\phi \wedge \mathsf{EX}\, Z)$ $\qquad Z \mapsto \psi \vee (\phi \wedge \mathsf{EX}\, Z)$
- $\mathsf{A}\,\phi\,\mathsf{U}\,\psi \;=\; \mu Z.\, \psi \vee (\phi \wedge \mathsf{AX}\, Z)$ $\qquad Z \mapsto \psi \vee (\phi \wedge \mathsf{AX}\, Z)$
- ...

$\mathsf{EG}\,p \;=\; \nu Z.\, p \wedge \mathsf{EX}\,Z$ $\qquad$ $Z \mapsto p \wedge \mathsf{EX}\,Z$

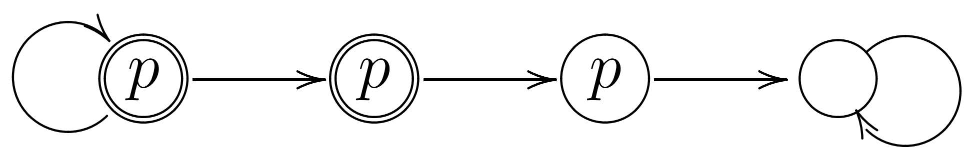
$\texttt{true}$



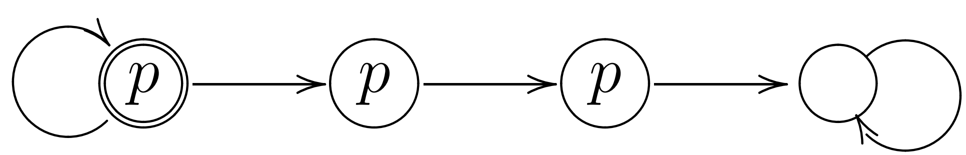
$p \wedge \mathsf{EX}\,\texttt{true} \;\equiv p$



$p \wedge \mathsf{EX}\,p$



$p \wedge \mathsf{EX}\,(p \wedge \mathsf{EX}\,p)$



CTL ($\neg$, $\wedge$, EX, E_U_, EG) (these connectives are sufficient)

$$\mathrm{Check} : \text{CTL} \mapsto \text{OBDD}$$

$$\boxed{\mathrm{Check}(\phi) \text{ represents } \{s \mid s \models \phi\}}$$

Example: $\mathrm{Check}(p)$ represents L_p

The order of variables is often crucial!

Symbolic model checking (EX _)

$\text{Check} : \text{CTL} \rightarrow \text{OBDD}$

$\text{Check}(\phi)$ represents $\{s \mid s \models \phi\}$

$$\text{Check}(\,\text{EX}\,\phi) \quad := \quad \exists \vec{x}'.\ R(\vec{x}, \vec{x}') \wedge f[\vec{x}'/\vec{x}] \qquad \text{where} \quad f = \text{Check}(\phi)$$

$$\text{Check}(\,\text{EX}\,\phi) \quad := \quad \text{EX}\, f$$

$\text{EX}\,\phi$

$\text{EX}\,Z$

$\text{EX}\,f$

Order of variables

$$\exists \vec{x}'.\ R(\vec{x}, \vec{x}') \wedge f[\vec{x}'/\vec{x}])$$

$$\vec{x} = x_1, x_2, \ldots, x_m$$

$$x_1 < x_1' < x_2 < x_2' < \ldots < x_m < x_m'$$

Symbolic model checking (E _ U _)

$\text{Check} : \text{CTL} \to \text{OBDD}$

$\text{Check}(\phi)$ represents $\{s \mid s \models \phi\}$

$$\text{Check}(\mathsf{E}\,\phi\,\mathsf{U}\,\psi) \quad := \quad \mu Z.\, g \vee (f \wedge \mathsf{EX}\,Z) \qquad \text{where} \quad f = \text{Check}(\phi)$$
$$g = \text{Check}(\psi)$$

$$h \mapsto g \vee (f \wedge \mathsf{EX}\,h)$$

$$h \mapsto g \vee (f \wedge \exists \vec{x}'.\, R(\vec{x}, \vec{x}') \wedge h[\vec{x}'/\vec{x}])$$

$$\texttt{false}$$
$$g \vee (f \wedge \mathsf{EX}\,\texttt{false}) \quad \equiv \quad g$$
$$g \vee (f \wedge \mathsf{EX}\,(g \vee (f \wedge \mathsf{EX}\,\texttt{false}))) \quad \equiv \quad g \vee (f \wedge \mathsf{EX}\,g)$$
$$\ldots \quad \equiv \quad g \vee (f \wedge \mathsf{EX}\,(g \vee (f \wedge \mathsf{EX}\,g)$$

$$\mu Z.\, g \vee (f \wedge \mathsf{EX}\,Z)$$

Symbolic model checking (EG _)

Check : CTL $\rightarrow$ OBDD

$\mathrm{Check}(\phi)$ represents $\{s \mid s \models \phi\}$

$\mathrm{Check}(\mathsf{EG}\,\phi) \quad := \quad \nu Z.\ f \wedge \mathsf{EX}\, Z$ where $f = \mathrm{Check}(\phi)$

$$h \mapsto f \wedge \mathsf{EX}\, h$$

$$h \mapsto f \wedge \exists \vec{x}'.\ R(\vec{x}, \vec{x}') \wedge h[\vec{x}'/\vec{x}]$$

EX ϕ	E ϕ U ψ	EG ϕ
EX Z	E Z U Z'	EG Z
EX f	E f U g	EG f