Interpolation and Total Search Problems

Noah Fleming
University of California, San Diego

Last Time

Interpolation

For many (all?) proof systems P it is possible to relate their complexity to the complexity of circuits in some associated model C_P of monotone computation

Last Time

Interpolation

For many (all?) proof systems P it is possible to relate their complexity to the complexity of circuits in some associated model C_P of monotone computation

P-proof of F
of size s

$\longrightarrow$

C_P-Computation of f_F
of size poly(s)

Where $f_F : \{0,1\}^m \to \{0,1\}$ is a monotone function associated with F

Last Time

Interpolation

For many (all?) proof systems P it is possible to relate their complexity to the complexity of circuits in some associated model C_P of monotone computation

P-proof of F of size s $\rightarrow$ **C_P-Computation of f_F** of size poly(s)

Where $f_F : \{0,1\}^m \to \{0,1\}$ is a monotone function associated with F

Last time, we showed such a translation when F is a split formula

Last Time

Interpolation

For many (all?) proof systems P it is possible to relate their complexity to the complexity of circuits in some associated model C_P of monotone computation

P-proof of F
of size s

→

C_P-Computation of f_F
of size $\text{poly}(s)$

Where $f_F : \{0,1\}^m \to \{0,1\}$ is a monotone function associated with F

Last time, we showed such a translation when F is a split formula

Split Formula:

$$F(x, y, z) = A(x, z) \wedge B(y, z)$$

Where A, B are CNF and all z variables occur positively in A

Last Time

Split Formula:

$$F(x, y, z) = A(x, z) \wedge B(y, z)$$

Where A, B are CNF and all z variables occur positively in A

Last Time

Split Formula: $$F(x, y, z) = A(x, z) \wedge B(y, z)$$

Where A, B are CNF and all z variables occur positively in A

The Function Computed

Let $\alpha \in \{0,1\}^y$ be any assignment to $y \Longrightarrow A(x, \alpha)$ or $B(y, \alpha)$ is unsatisfiable

Define monotone "interpolant" function $I_F(z) = \begin{cases} 0 & \text{if } A(x, \alpha) \text{ is satisfiable} \\ 1 & \text{if } B(y, \alpha) \text{ is satisfiable} \end{cases}$

Last Time

Split Formula: $$F(x, y, z) = A(x, z) \wedge B(y, z)$$

Where A, B are CNF and all z variables occur positively in A

The Function Computed

Let $\alpha \in \{0,1\}^y$ be any assignment to $y \implies A(x, \alpha)$ or $B(y, \alpha)$ is unsatisfiable

Define monotone "interpolant" function $I_F(z) = \begin{cases} 0 & \text{if } A(x, \alpha) \text{ is satisfiable} \\ 1 & \text{if } B(y, \alpha) \text{ is satisfiable} \end{cases}$

Interpolation for CP [P97]: Let F be a split formula

CP proof of F
of size s

→

Monotone Real Circuit computing I_F
of size poly(s)

Today: Interpolation for any Formula

Goal for today: remove the “split” assumption!

Today: Interpolation for any Formula

Goal for today: remove the “split” assumption!

Take a different view of the connection between proofs and circuits

→ Proofs and circuits as computations of search problems

Today: Interpolation for any Formula

Goal for today: remove the "split" assumption!

Take a different view of the connection between proofs and circuits

$\rightarrow$ Proofs and circuits as computations of search problems

Search Problem: A relation $S \subseteq I \times O$

Given $x \in I$, output $y \in O$ such that $(x, y) \in S$

Today: Interpolation for any Formula

Goal for today: remove the "split" assumption!

Take a different view of the connection between proofs and circuits

→ Proofs and circuits as computations of search problems

Search Problem: A relation $S \subseteq I \times O$

Given $x \in I$, output $y \in O$ such that $(x, y) \in S$

e.g.

Falsified Clause search problem: for an unsatisfiable CNF

$F = C_1 \wedge \ldots \wedge C_m$, given $x \in \{0,1\}^n$, output $i \in [m]$ such that $C_i(x) = 0$

Today: Interpolation for any Formula

First, we will characterize the complexity of circuits and proofs in terms of the complexity of search problems

Karchmer-Wigderson

The Karchmer-Wigderson search problem characterizes models of monotone circuit complexity

Karchmer-Wigderson

The Karchmer-Wigderson search problem characterizes models of monotone circuit complexity

Monotone Karchmer-Wigdreson: $\mathsf{mKW}_f \subseteq f^{-1}(1) \times f^{-1}(0) \times \{0,1\}^n$

Given $x \in f^{-1}(1)$, $y \in f^{-1}(0)$ output $i \in [n]$ such that $x_i = 1$ and $y_i = 0$

Karchmer-Wigderson

The Karchmer-Wigderson search problem characterizes models of monotone circuit complexity

Monotone Karchmer-Wigdreson: $\text{mKW}_f \subseteq f^{-1}(1) \times f^{-1}(0) \times \{0,1\}^n$

Given $x \in f^{-1}(1)$, $y \in f^{-1}(0)$ output $i \in [n]$ such that $x_i = 1$ and $y_i = 0$

Intuition: any monotone circuit computing f must must differentiate between 0-inputs (y) and 1-inputs (x)

Karchmer-Wigderson and Circuits

The communication complexity of mKW will characterize monotone formulas

Karchmer-Wigderson and Circuits

The communication complexity of mKW will characterize monotone formulas

Communication Complexity: Two players

Alice	Bob
Knows $x \in f^{-1}(1)$	Knows $y \in f^{-1}(0)$

Communicate bits of information to find a solution $i \in [m]$ such that $(x, y, i) \in \mathrm{mKW}_f$

Karchmer-Wigderson and Circuits

The communication complexity of mKW will characterize monotone formulas

Communication Complexity: Two players

Alice

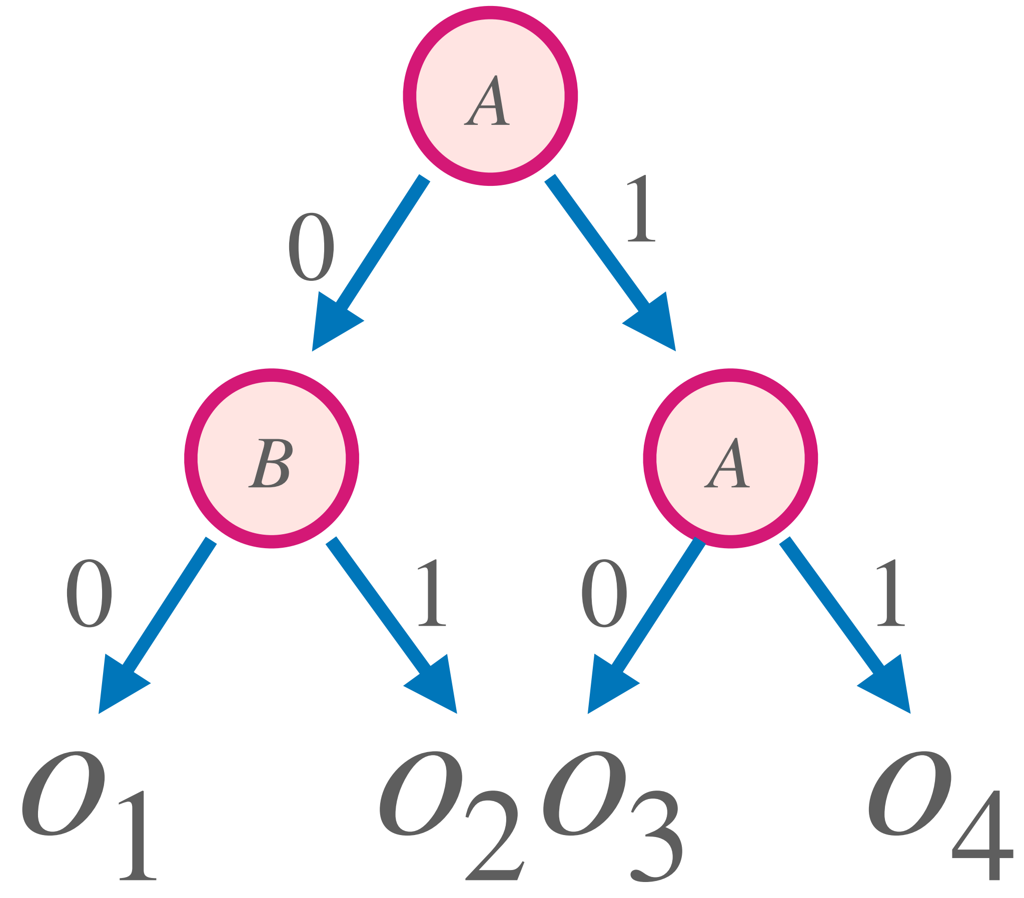
Knows $x \in f^{-1}(1)$

Bob

Knows $y \in f^{-1}(0)$

Communicate bits of information to find a solution $i \in [m]$ such that $(x, y, i) \in \text{mKW}_f$

View cc-protocol as a tree:



Karchmer-Wigderson and Circuits

The communication complexity of mKW will characterize monotone formulas

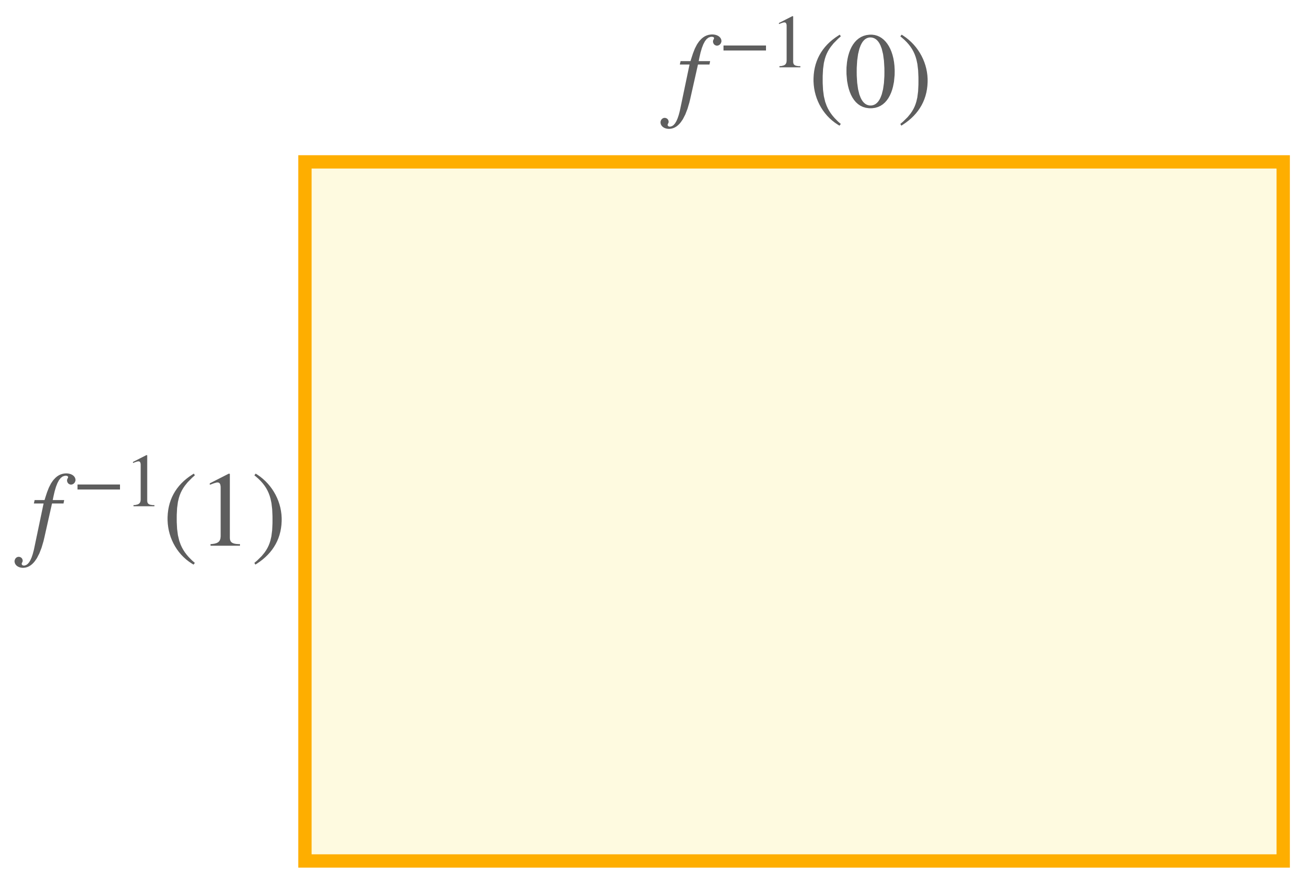
Communication Complexity: Two players

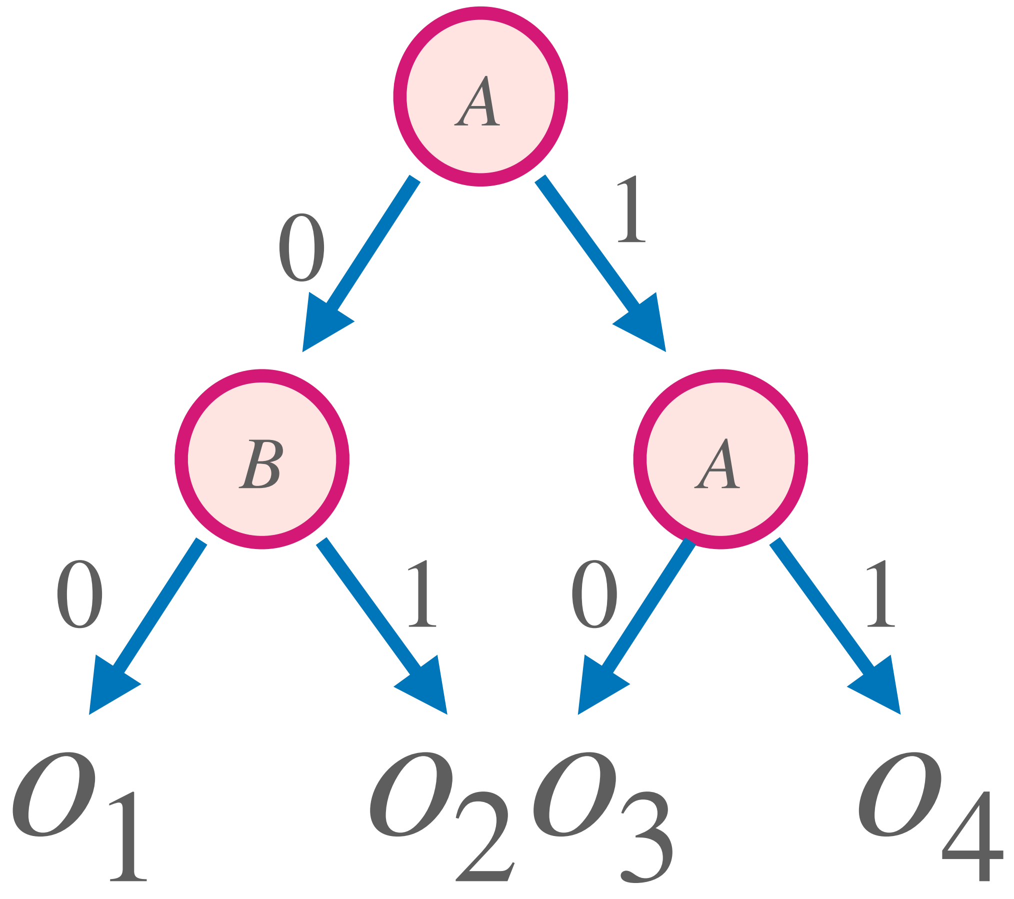
Alice		Bob
Knows $x \in f^{-1}(1)$	$\Longleftrightarrow$	Knows $y \in f^{-1}(0)$

Communicate bits of information to find a solution $i \in [m]$ such that $(x, y, i) \in \mathrm{mKW}_f$

View cc-protocol as a tree:

Protocols as rectangle partitions:





Karchmer-Wigderson and Circuits

The communication complexity of mKW will characterize monotone formulas

Communication Complexity: Two players

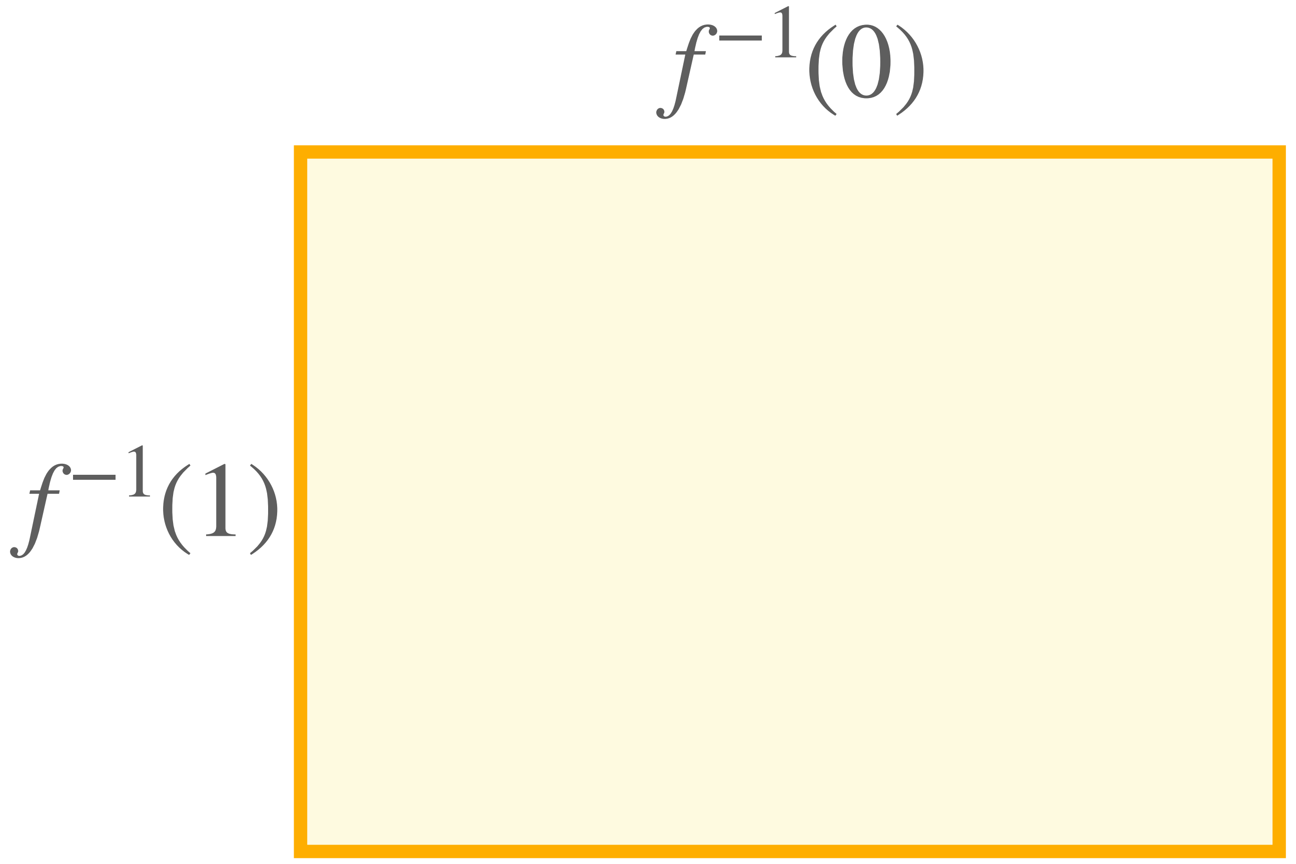
Alice		Bob
Knows $x \in f^{-1}(1)$	⟺	Knows $y \in f^{-1}(0)$

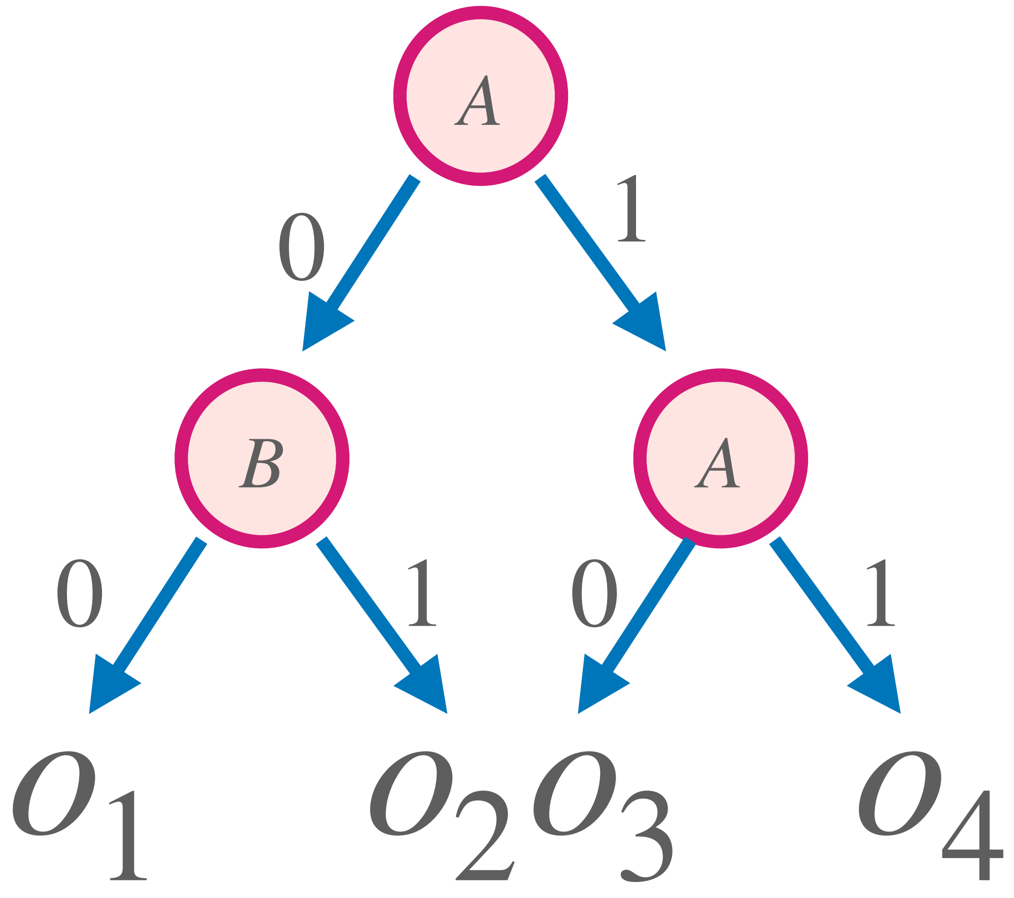
Communicate bits of information to find a solution $i \in [m]$ such that $(x, y, i) \in \mathrm{mKW}_f$

View cc-protocol as a tree:

Protocols as rectangle partitions:

If Alice sends a bit, partitions rows





Karchmer-Wigderson and Circuits

The communication complexity of mKW will characterize monotone formulas

Communication Complexity: Two players

Alice

Knows $x \in f^{-1}(1)$

Bob

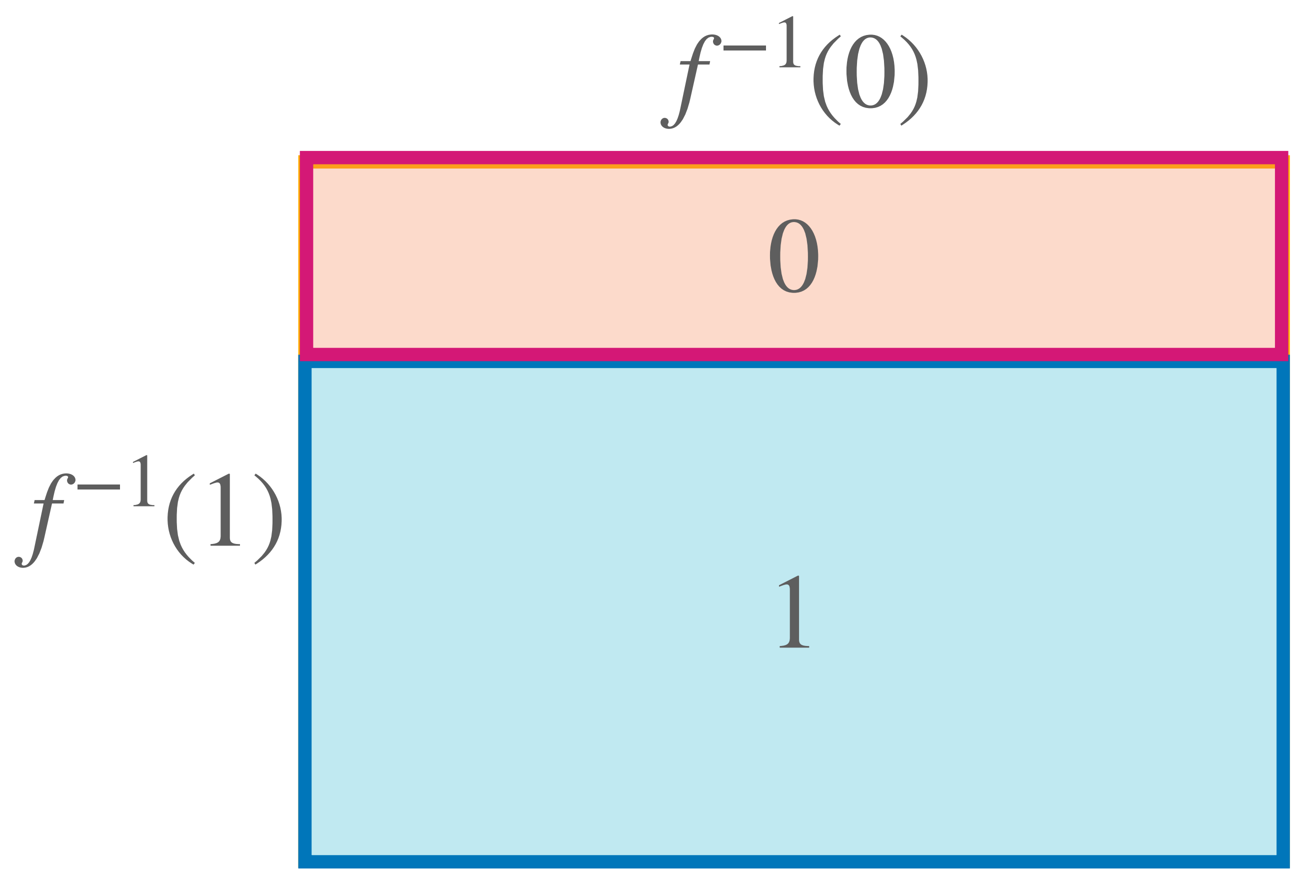
Knows $y \in f^{-1}(0)$

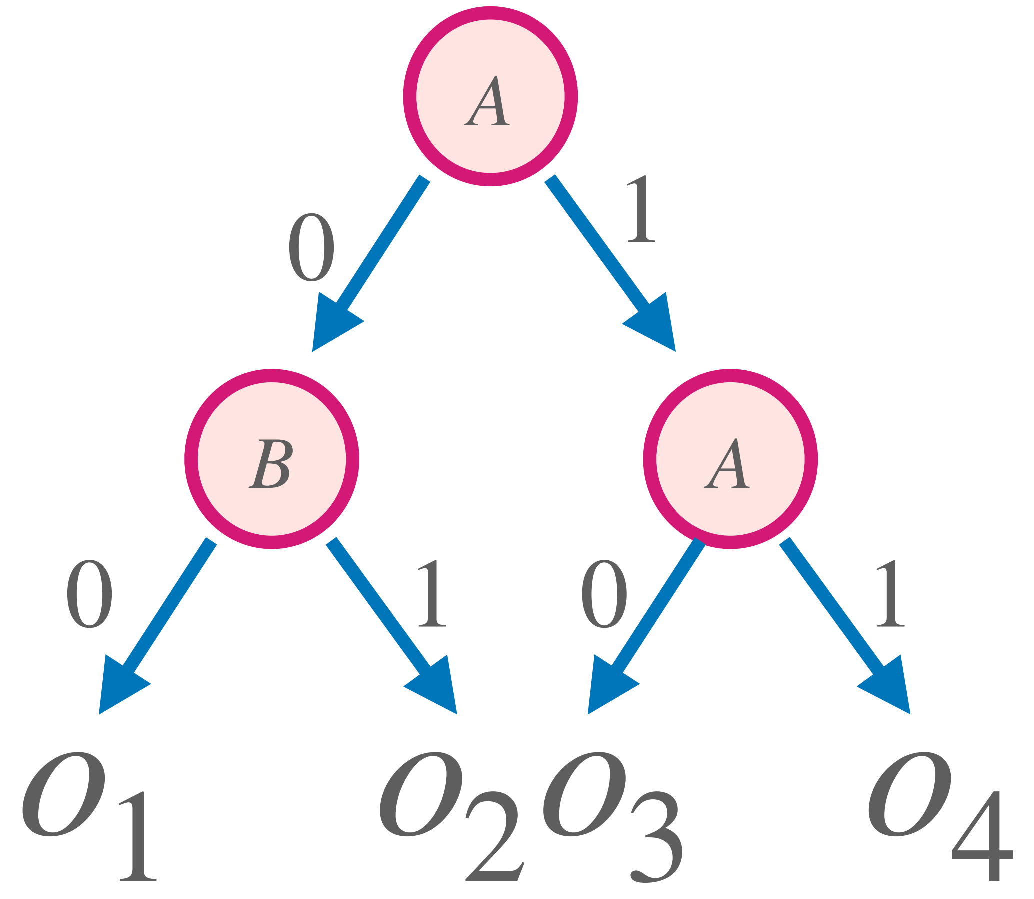
Communicate bits of information to find a solution $i \in [m]$ such that $(x, y, i) \in \text{mKW}_f$

View cc-protocol as a tree:

Protocols as rectangle partitions:

If Alice sends a bit, partitions rows





Karchmer-Wigderson and Circuits

The communication complexity of mKW will characterize monotone formulas

Communication Complexity: Two players

Alice		Bob
Knows $x \in f^{-1}(1)$	$\Longleftrightarrow$	Knows $y \in f^{-1}(0)$

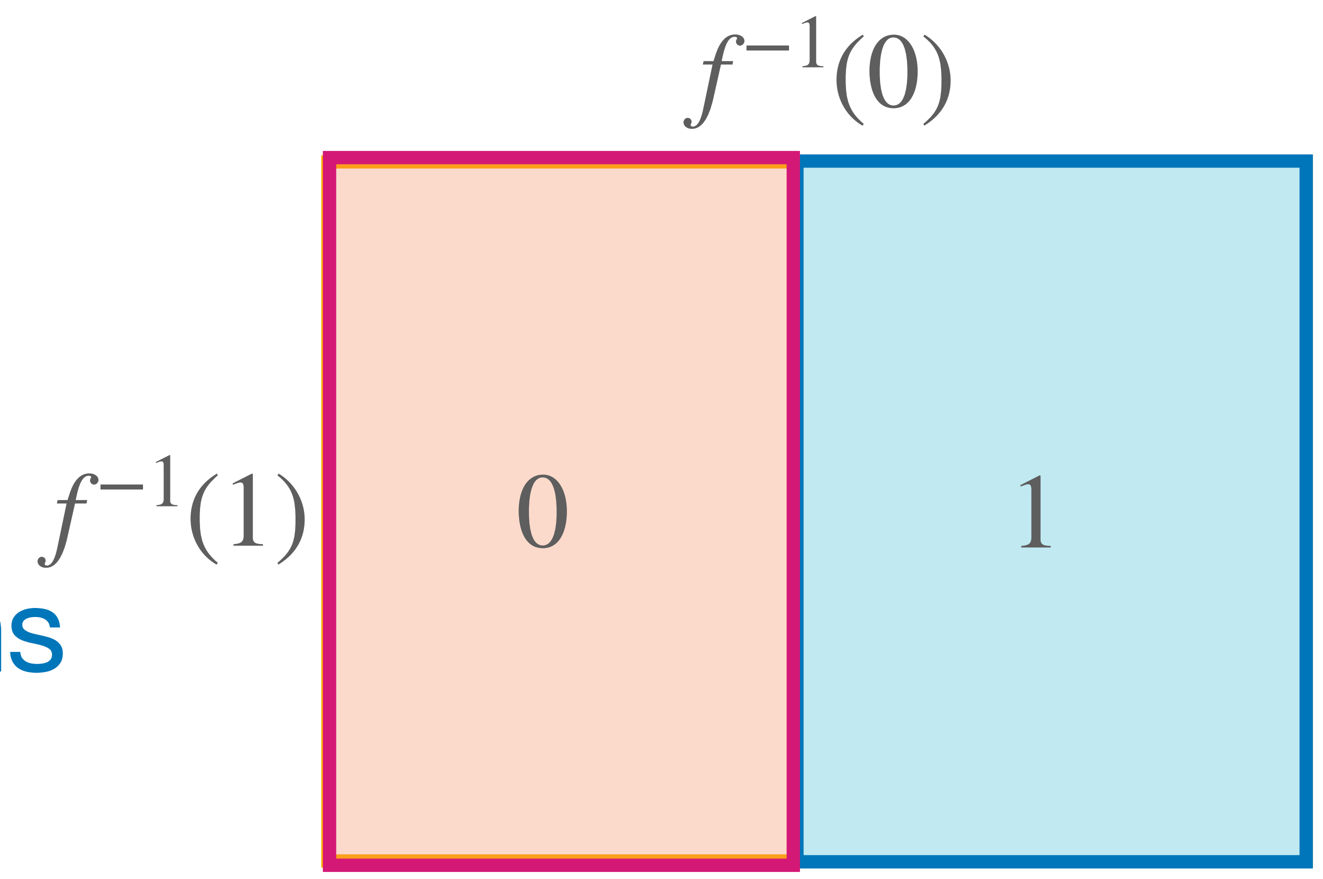
Communicate bits of information to find a solution $i \in [m]$ such that $(x, y, i) \in \mathrm{mKW}_f$

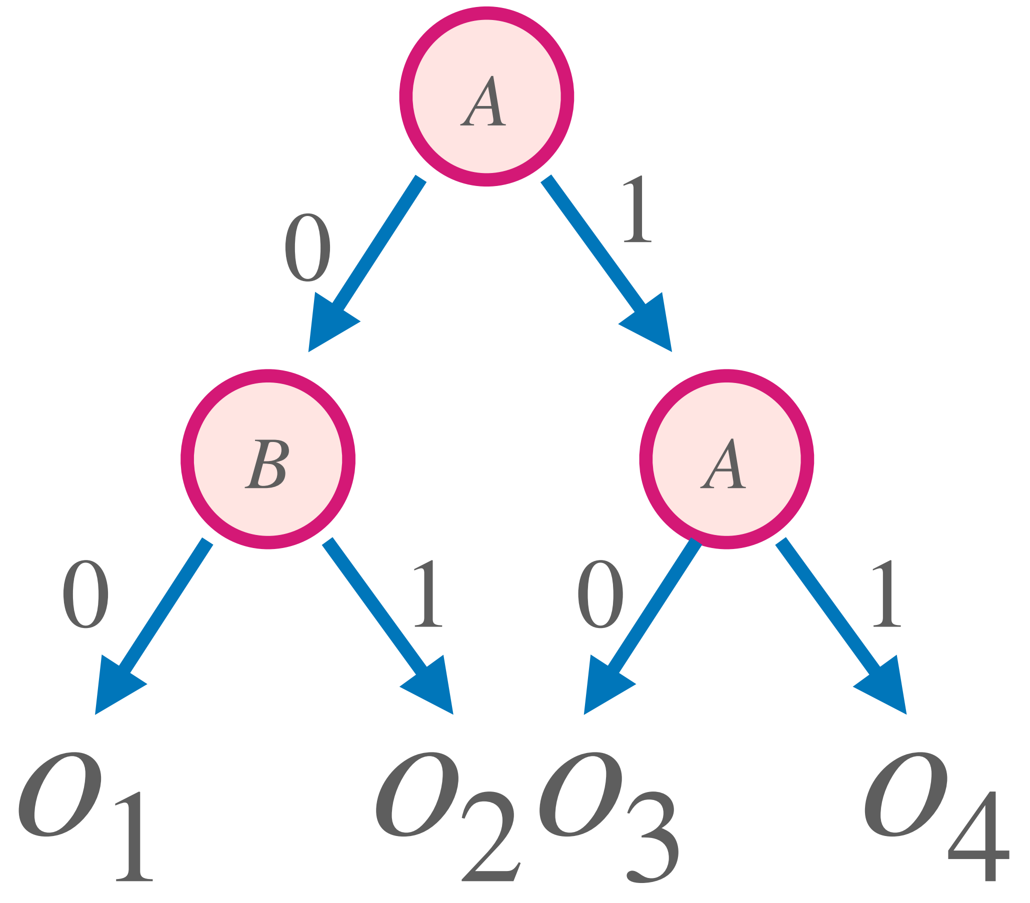
View cc-protocol as a tree:

Protocols as rectangle partitions:

If Alice sends a bit, partitions rows

If Bob sends a bit, partitions columns





Karchmer-Wigderson and Circuits

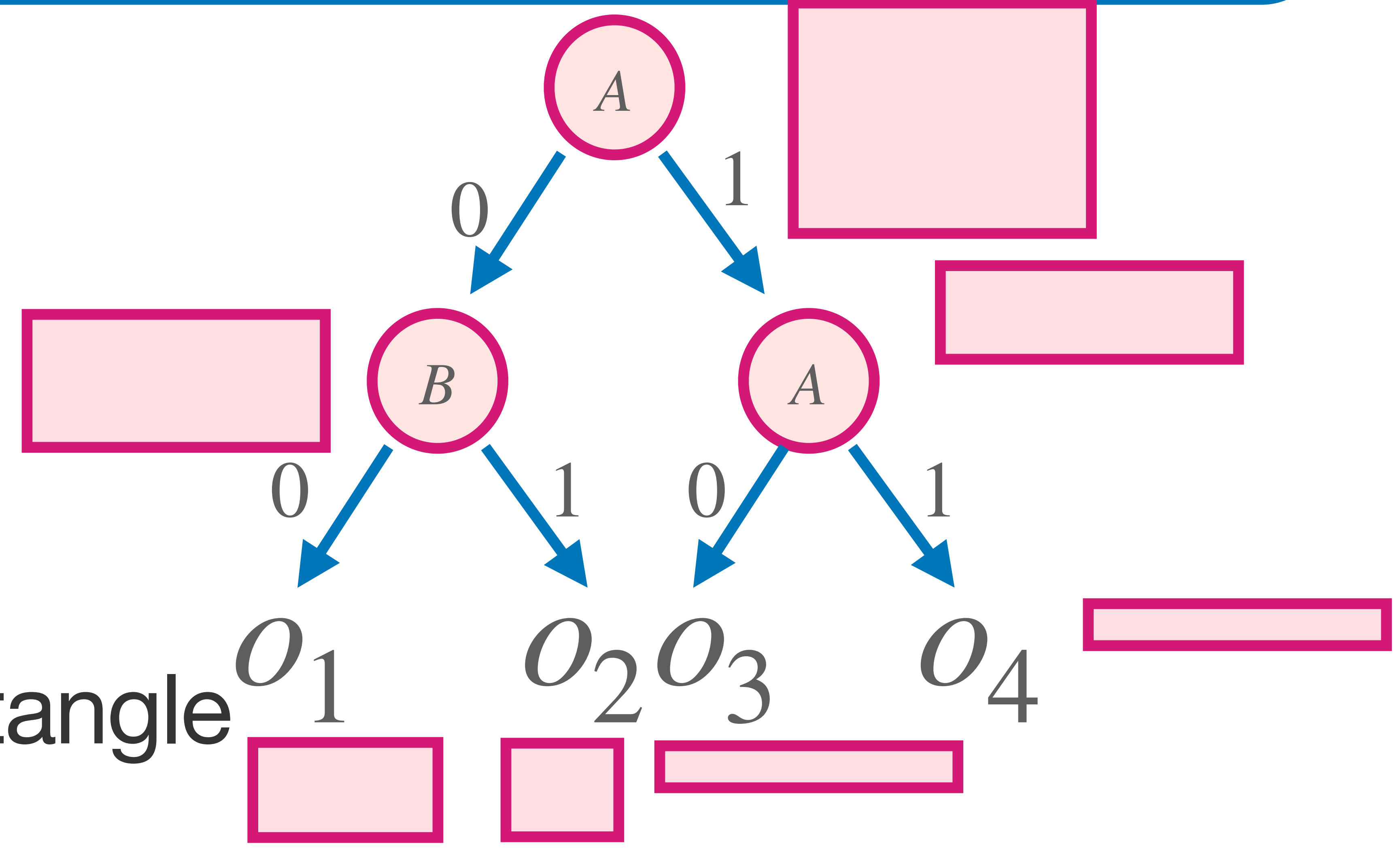
The communication complexity of mKW will characterize monotone formulas

Communication Complexity: Two players

Alice		Bob
Knows $x \in f^{-1}(1)$	⟺	Knows $y \in f^{-1}(0)$

Communicate bits of information to find a solution $i \in [m]$ such that $(x, y, i) \in \mathrm{mKW}_f$

View cc-protocol as a tree:



Protocols as rectangle partitions:

If Alice sends a bit, partitions rows

If Bob sends a bit, partitions columns

$\implies$ Every node in the protocol tree is associated with a rectangle

Karchmer-Wigderson and Circuits

The communication complexity of mKW will characterize monotone formulas

Communication Complexity: Two players

Alice

Knows $x \in f^{-1}(1)$

Bob

Knows $y \in f^{-1}(0)$

Communicate bits of information to find a solution $i \in [m]$ such that $(x, y, i) \in \mathrm{mKW}_f$

View cc-protocol as a tree:

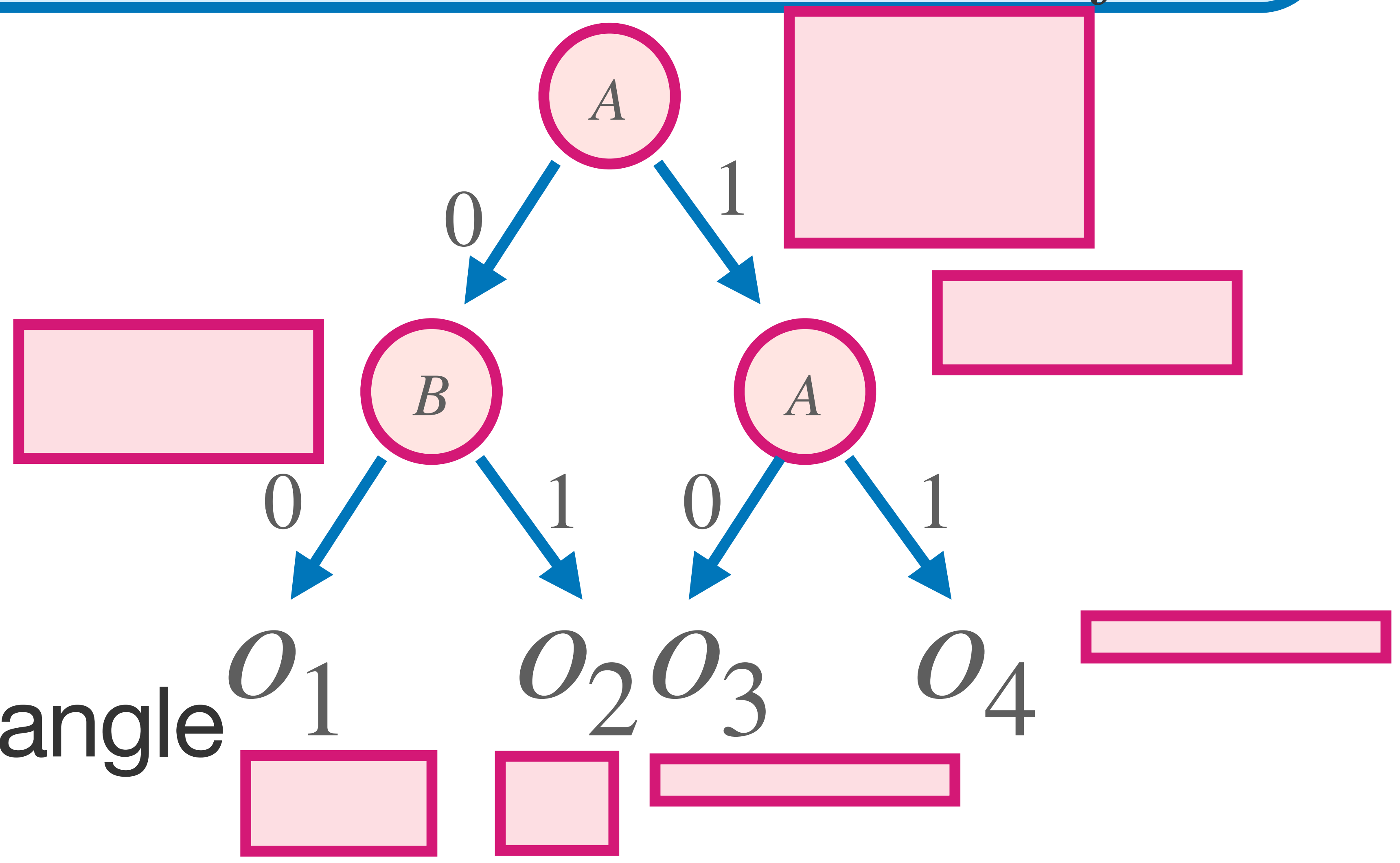
Protocols as rectangle partitions:

If Alice sends a bit, partitions rows

If Bob sends a bit, partitions columns

$\Longrightarrow$ Every node in the protocol tree is associated with a rectangle

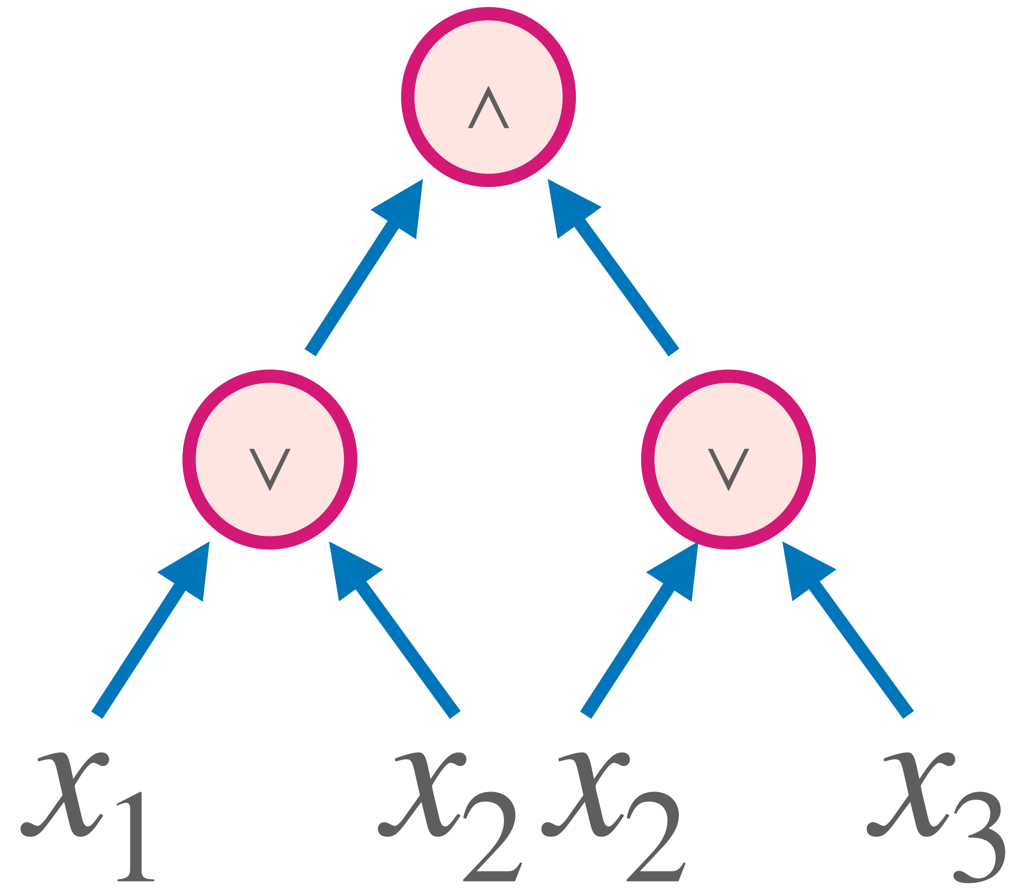
$\rightarrow$ Leaves are monochromatic rectangles



Karchmer-Wigderson and Circuits

Monotone Formula: A tree-like circuit using only $\wedge$ and $\vee$ gates

$\rightarrow$ Can only compute monotone functions



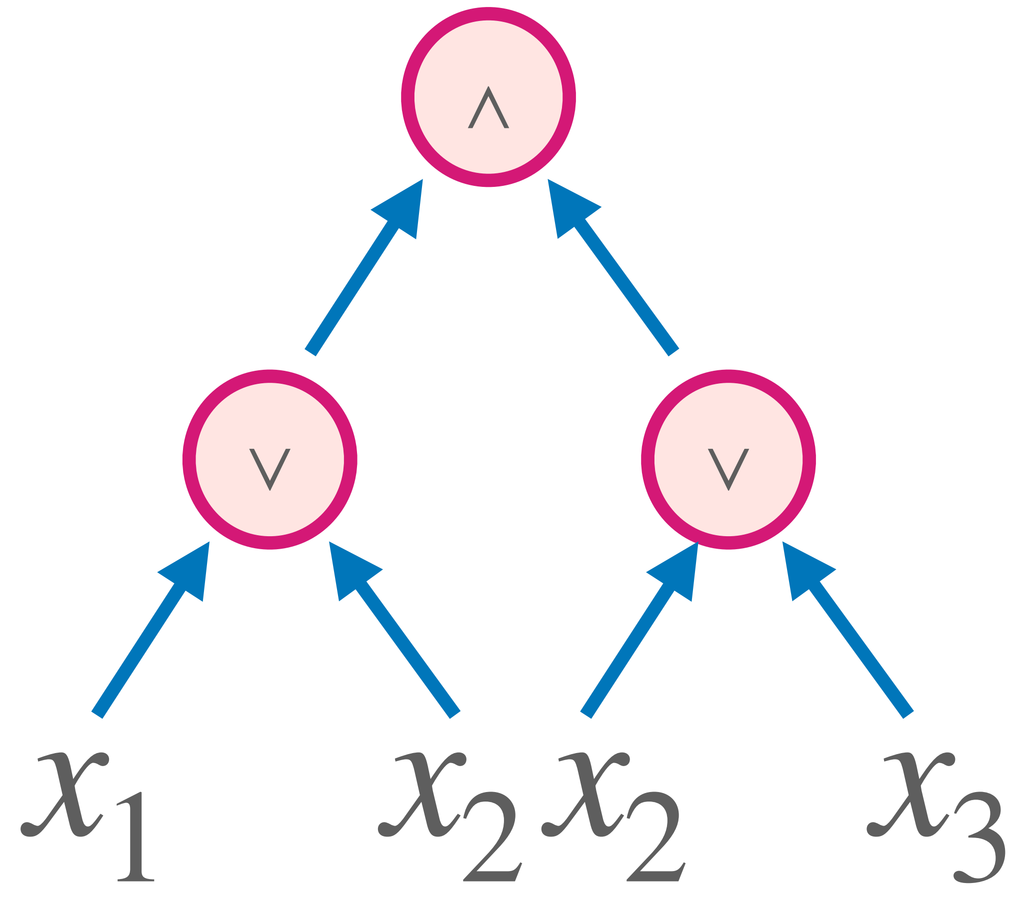
Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f:\{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

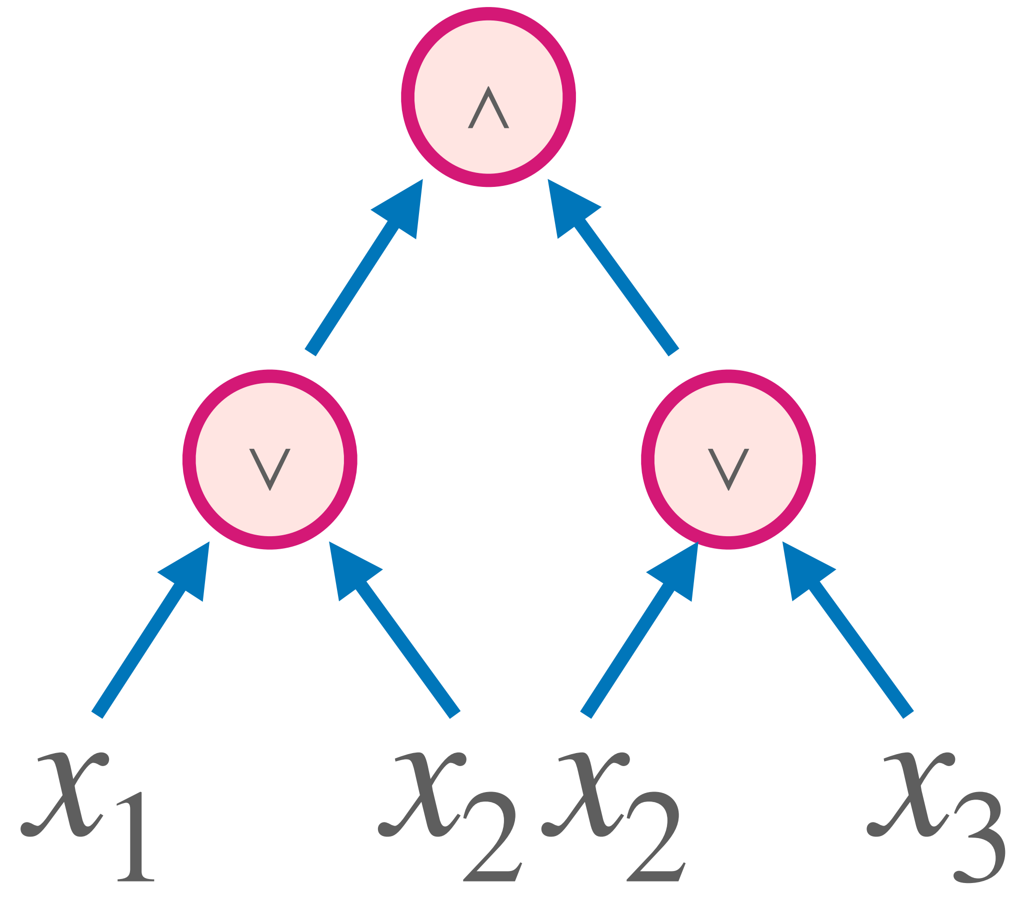


Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \rightarrow \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

Alice and Bob walk from the root to a leaf



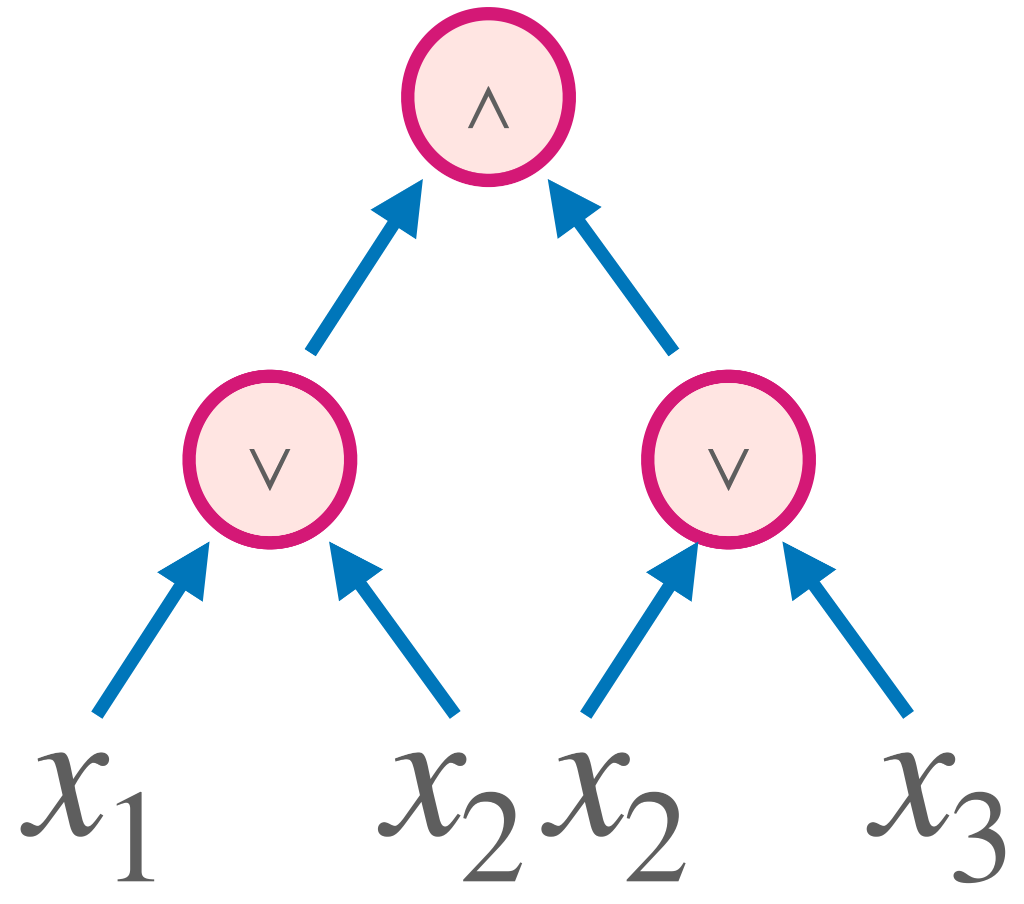
Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \rightarrow \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2



Karchmer-Wigderson and Circuits

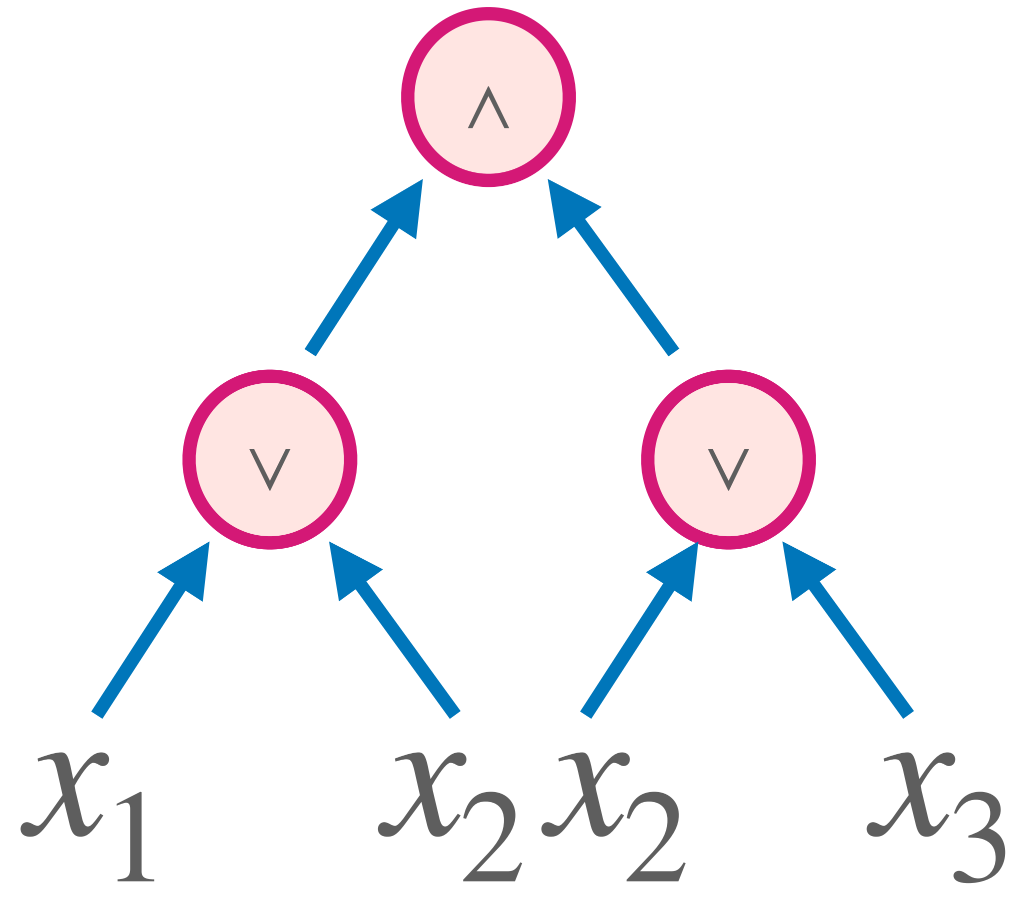
Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v



Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

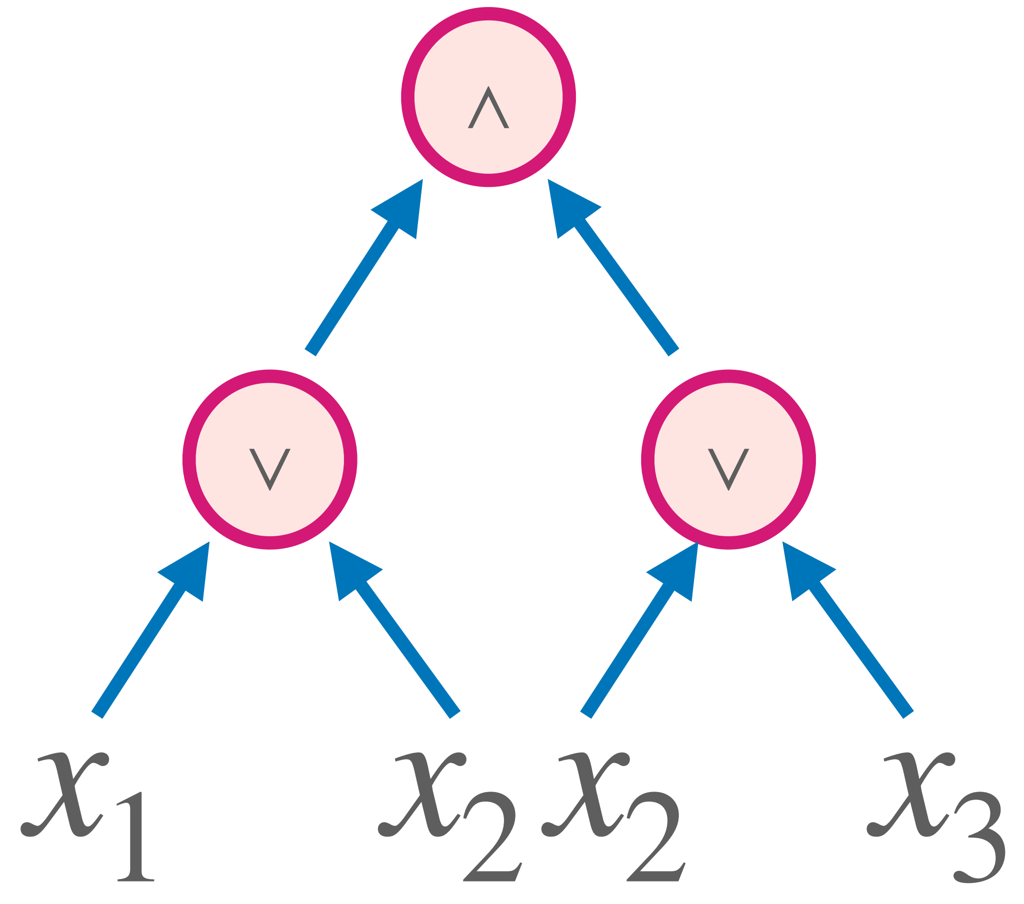
Proof→: From a monotone formula construct a communication protocol

Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v

Invariant: $C_v(x) = 1, C_v(y) = 0$



Karchmer-Wigderson and Circuits

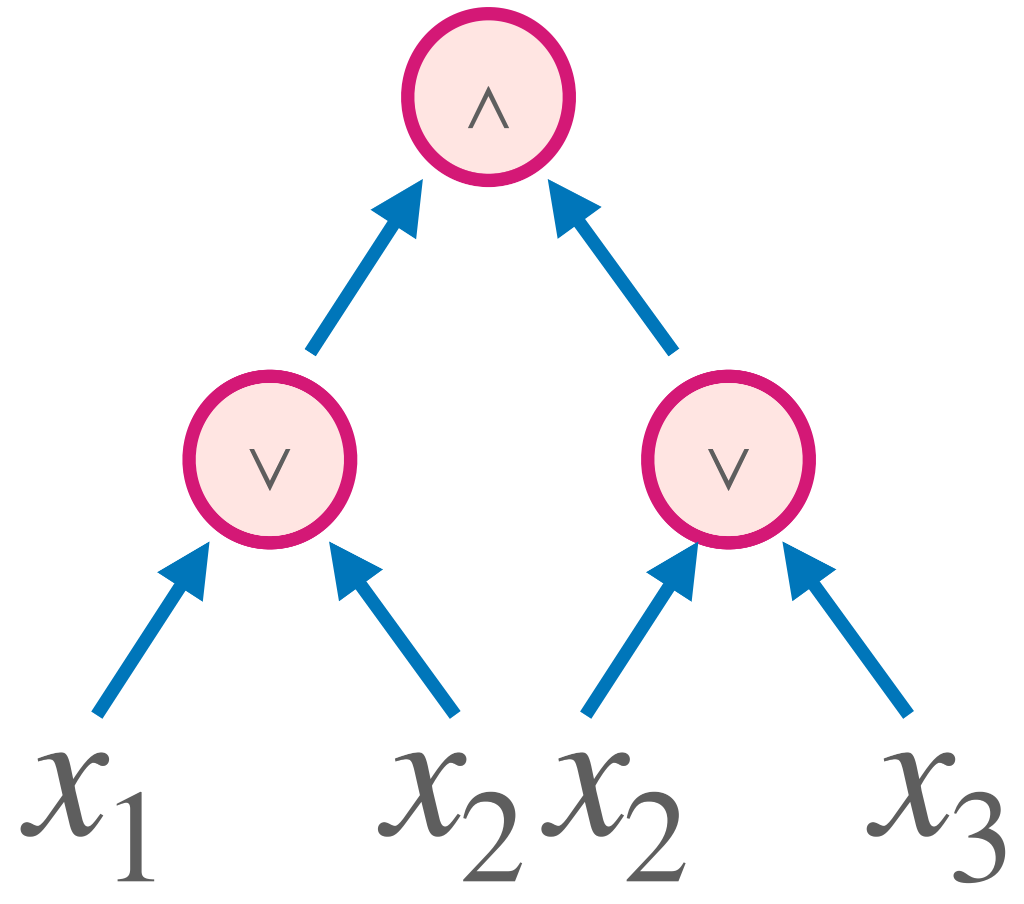
Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v



Invariant: $C_v(x) = 1, C_v(y) = 0$

- ∧: Bob sends 0 if $C_{u_1}(y) = 0, 1$ o.w. $(C_{u_2}(y) = 0)$

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

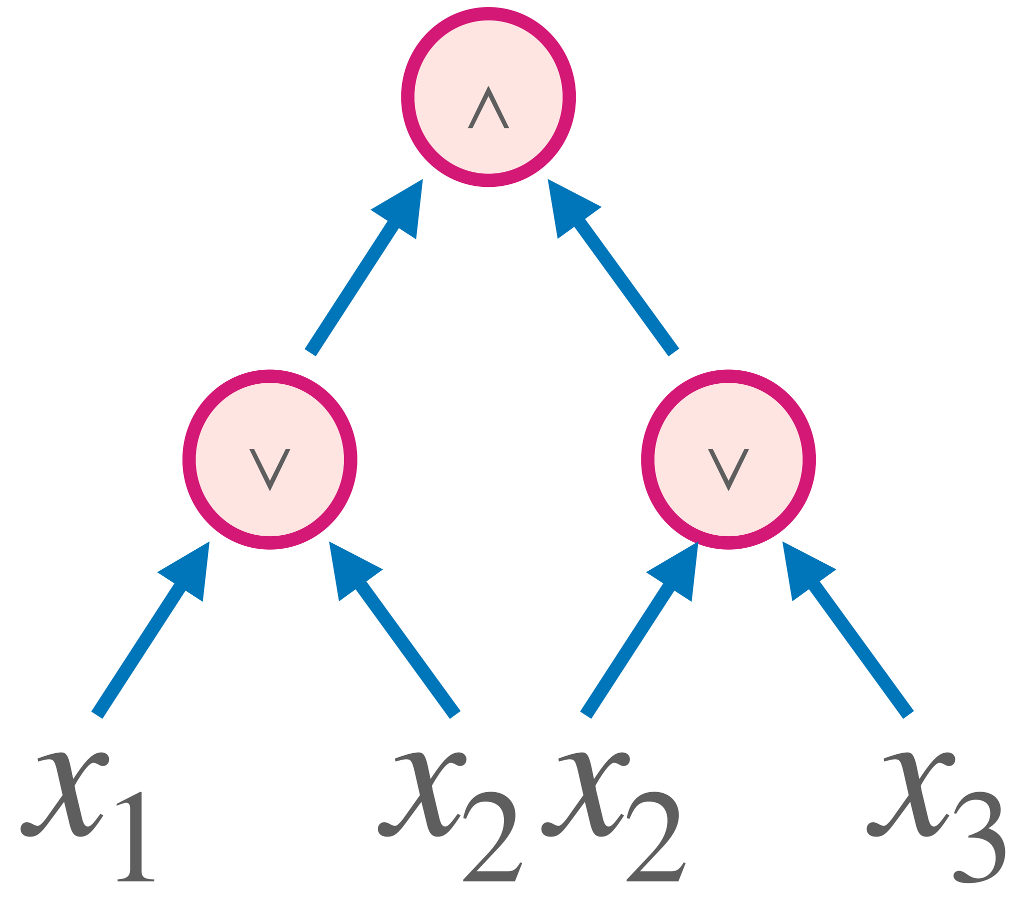
Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v

Invariant: $C_v(x) = 1, C_v(y) = 0$

- ∧: Bob sends 0 if $C_{u_1}(y) = 0$, 1 o.w. ($C_{u_2}(y) = 0$)
- ∨: Alice sends 0 if $C_{u_1}(x) = 1$, 1 o.w. ($C_{u_2}(x) = 1$)



Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \rightarrow \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

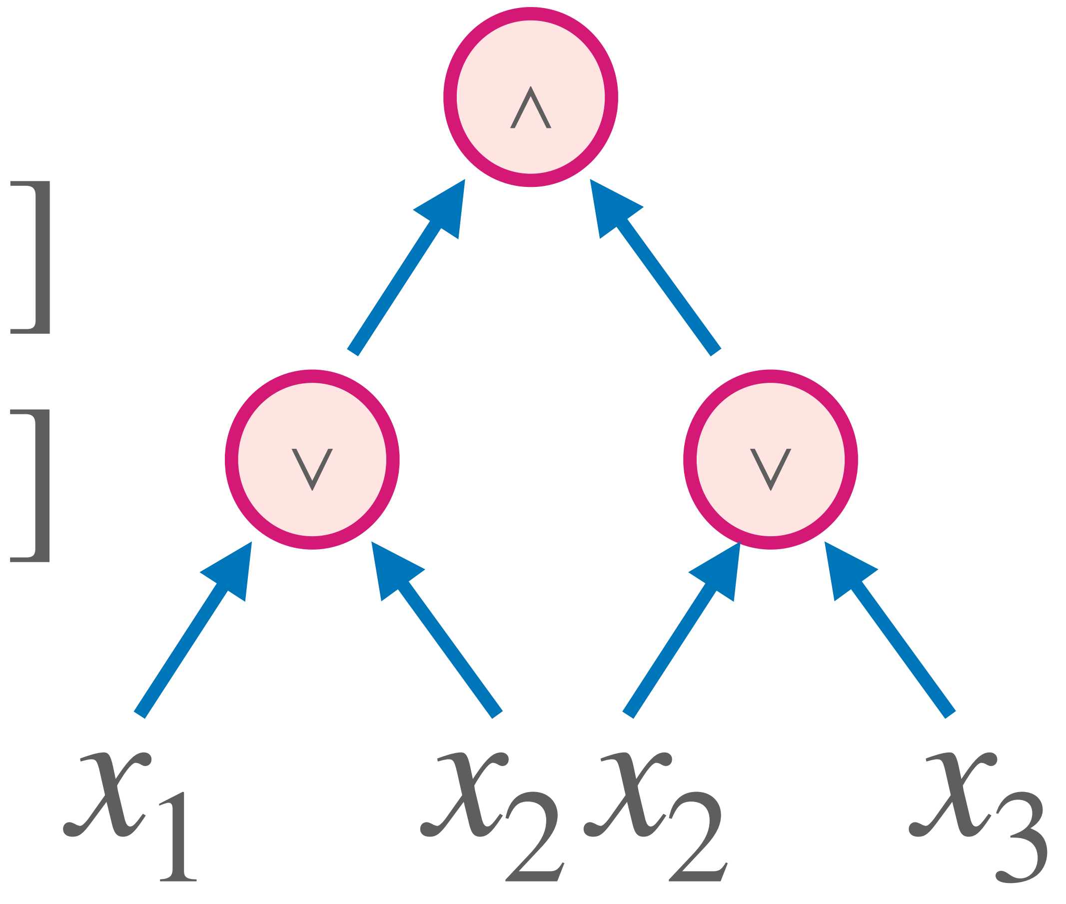
Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v

$x = [1,1,0]$

$y = [1,0,0]$



Invariant: $C_v(x) = 1, C_v(y) = 0$

- ∧: Bob sends 0 if $C_{u_1}(y) = 0$, 1 o.w. ($C_{u_2}(y) = 0$)
- ∨: Alice sends 0 if $C_{u_1}(x) = 1$, 1 o.w. ($C_{u_2}(x) = 1$)

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

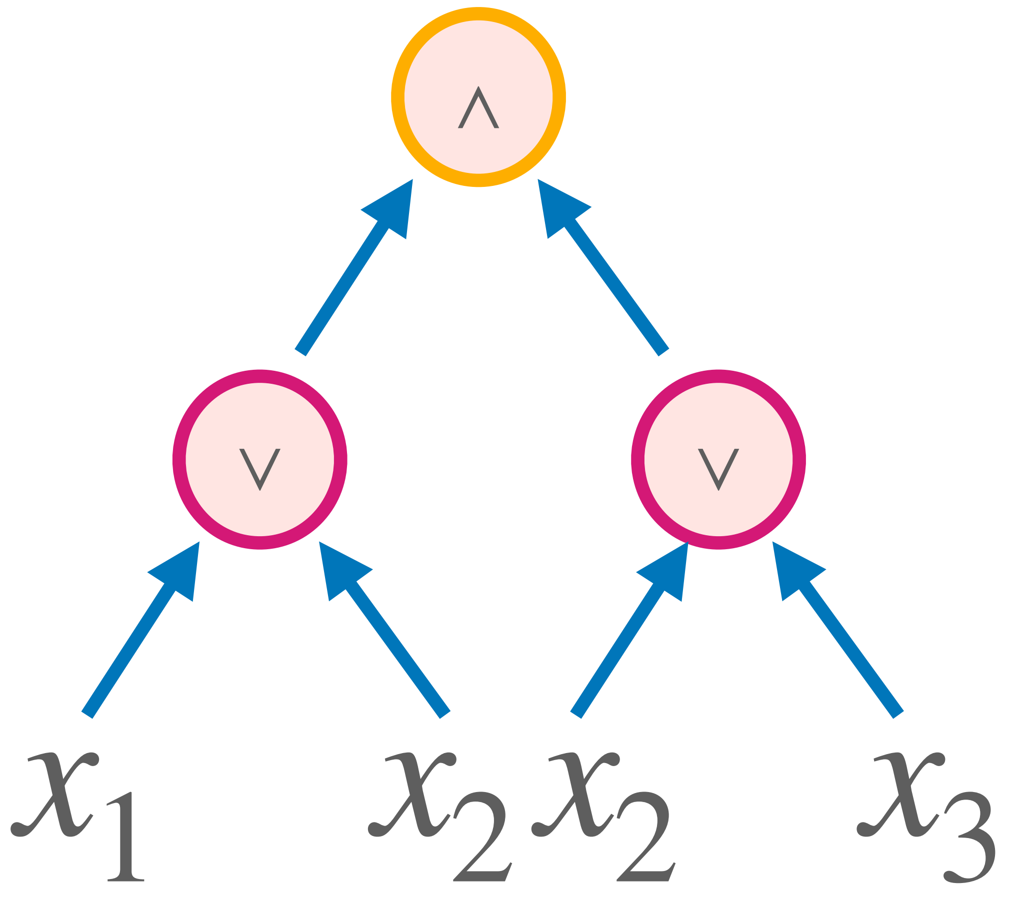
Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v

$x = [1,1,0]$

$y = [1,0,0]$



Invariant: $C_v(x) = 1$, $C_v(y) = 0$

- ∧: Bob sends 0 if $C_{u_1}(y) = 0$, 1 o.w. ($C_{u_2}(y) = 0$)
- ∨: Alice sends 0 if $C_{u_1}(x) = 1$, 1 o.w. ($C_{u_2}(x) = 1$)

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

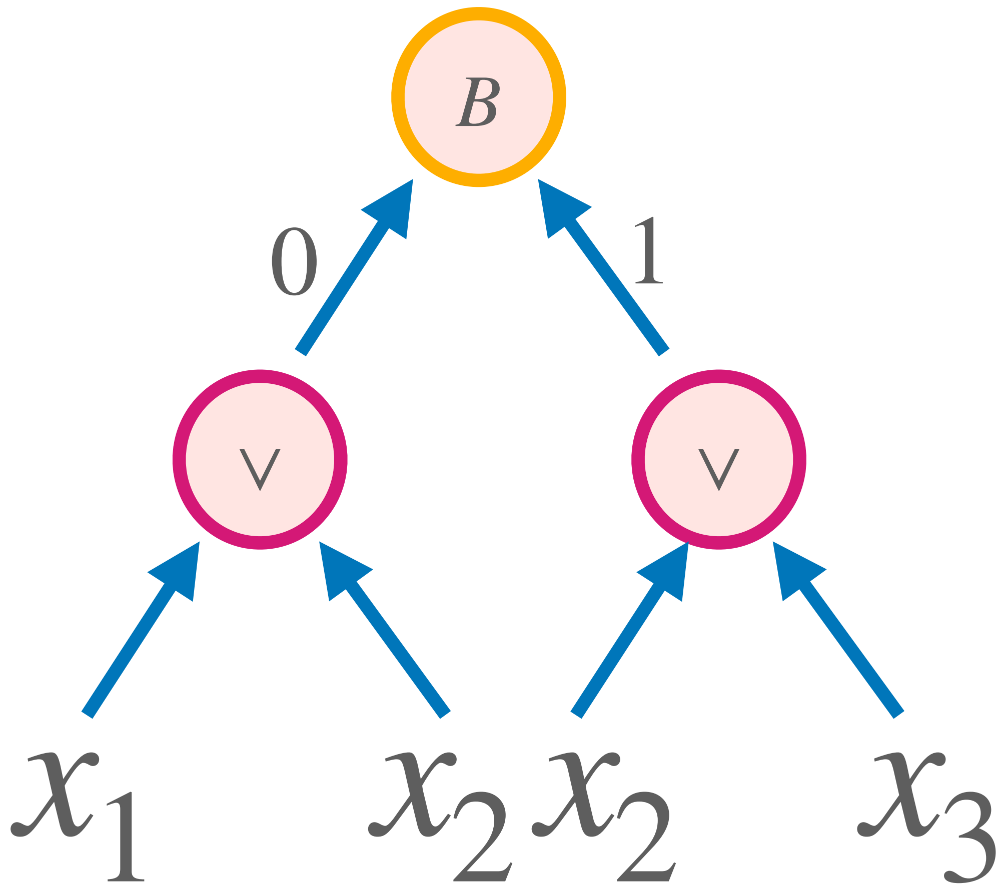
Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v

$x = [1,1,0]$

$y = [1,0,0]$



Invariant: $C_v(x) = 1, C_v(y) = 0$

- $\wedge$: Bob sends 0 if $C_{u_1}(y) = 0$, 1 o.w. ($C_{u_2}(y) = 0$)
- $\vee$: Alice sends 0 if $C_{u_1}(x) = 1$, 1 o.w. ($C_{u_2}(x) = 1$)

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

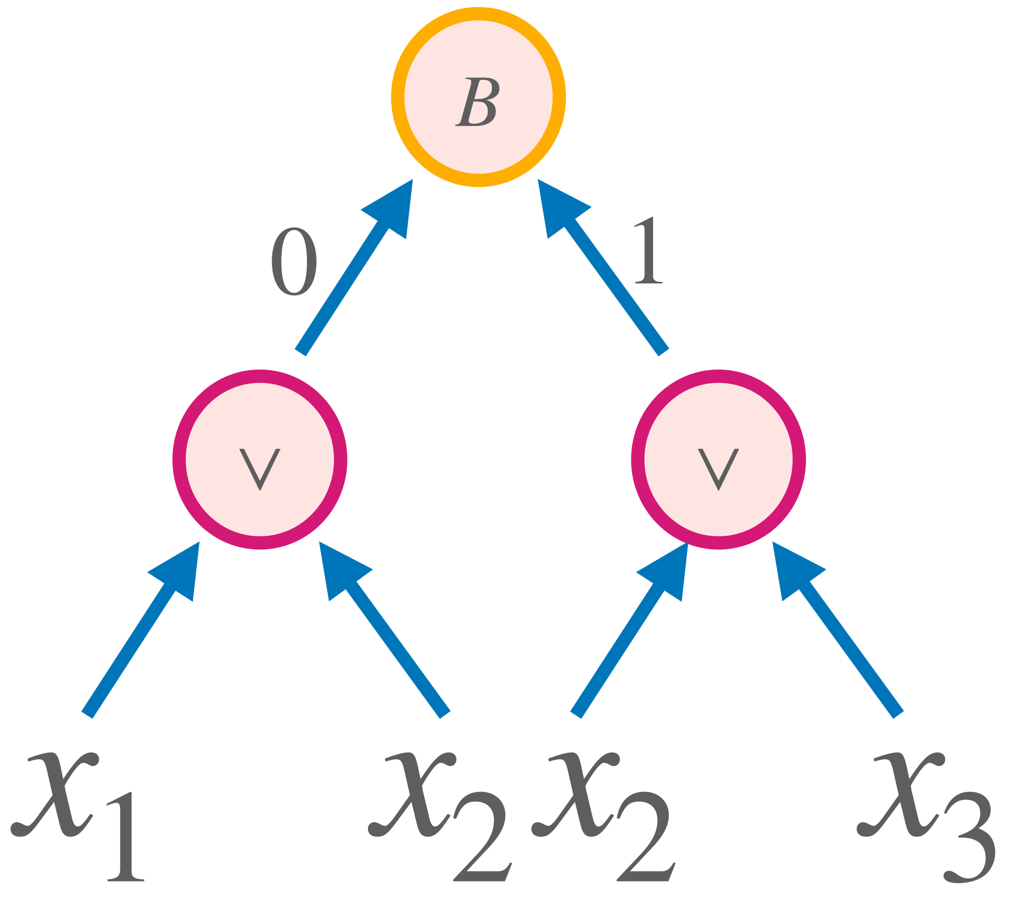
Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v

$x = [1,1,0]$

$y = [1,0,0]$



Bob sends 1

Invariant: $C_v(x) = 1$, $C_v(y) = 0$

- $\wedge$: Bob sends 0 if $C_{u_1}(y) = 0$, 1 o.w. ($C_{u_2}(y) = 0$)
- $\vee$: Alice sends 0 if $C_{u_1}(x) = 1$, 1 o.w. ($C_{u_2}(x) = 1$)

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

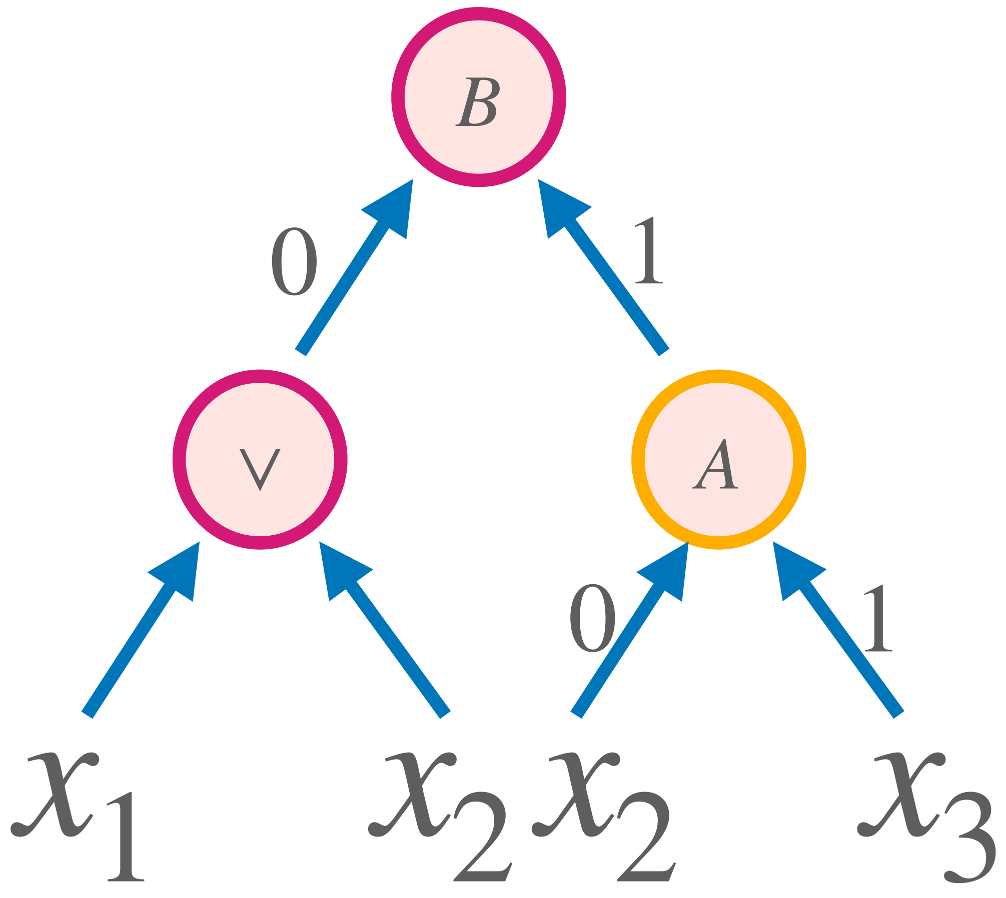
Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v

$x = [1,1,0]$

$y = [1,0,0]$



Invariant: $C_v(x) = 1, C_v(y) = 0$

- ∧: Bob sends 0 if $C_{u_1}(y) = 0$, 1 o.w. ($C_{u_2}(y) = 0$)
- ∨: Alice sends 0 if $C_{u_1}(x) = 1$, 1 o.w. ($C_{u_2}(x) = 1$)

Alice sends 0

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f: \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

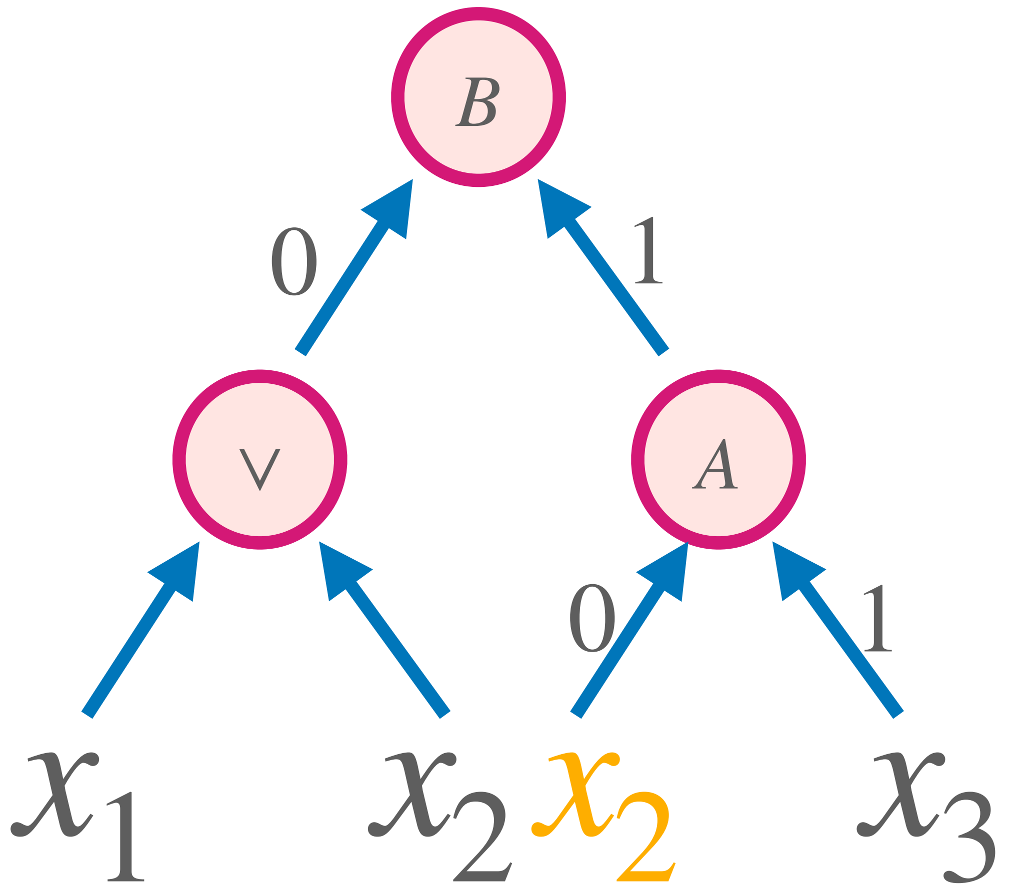
Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v

$x = [1,1,0]$

$y = [1,0,0]$



Invariant: $C_v(x) = 1$, $C_v(y) = 0$

- $\wedge$: Bob sends 0 if $C_{u_1}(y) = 0$, 1 o.w. ($C_{u_2}(y) = 0$)
- $\vee$: Alice sends 0 if $C_{u_1}(x) = 1$, 1 o.w. ($C_{u_2}(x) = 1$)

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a monotone formula construct a communication protocol

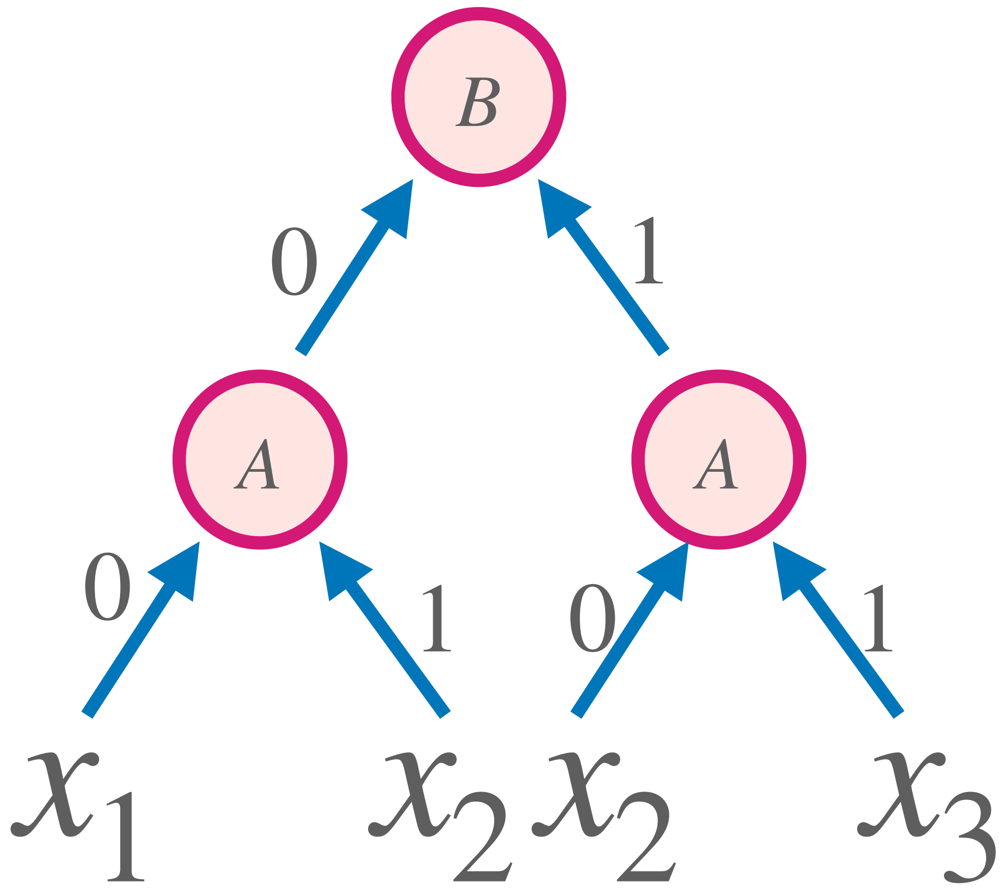
Alice and Bob walk from the root to a leaf

Let v be current node with children u_1, u_2

Let C_v be sub-circuit rooted at v

$x = [1,1,0]$

$y = [1,0,0]$



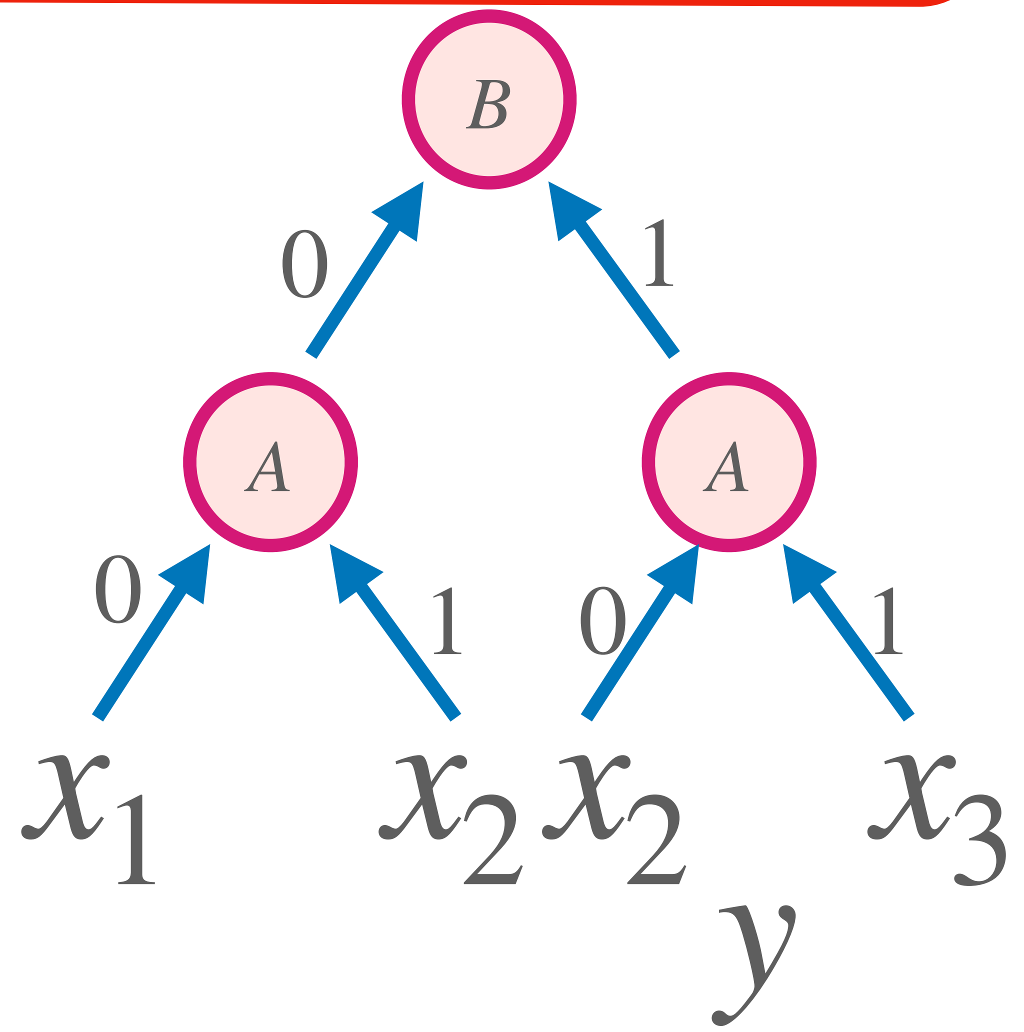
Invariant: $C_v(x) = 1, C_v(y) = 0$

- $\wedge$: Bob sends 0 if $C_{u_1}(y) = 0$, 1 o.w. ($C_{u_2}(y) = 0$)
- $\vee$: Alice sends 0 if $C_{u_1}(x) = 1$, 1 o.w. ($C_{u_2}(x) = 1$)

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

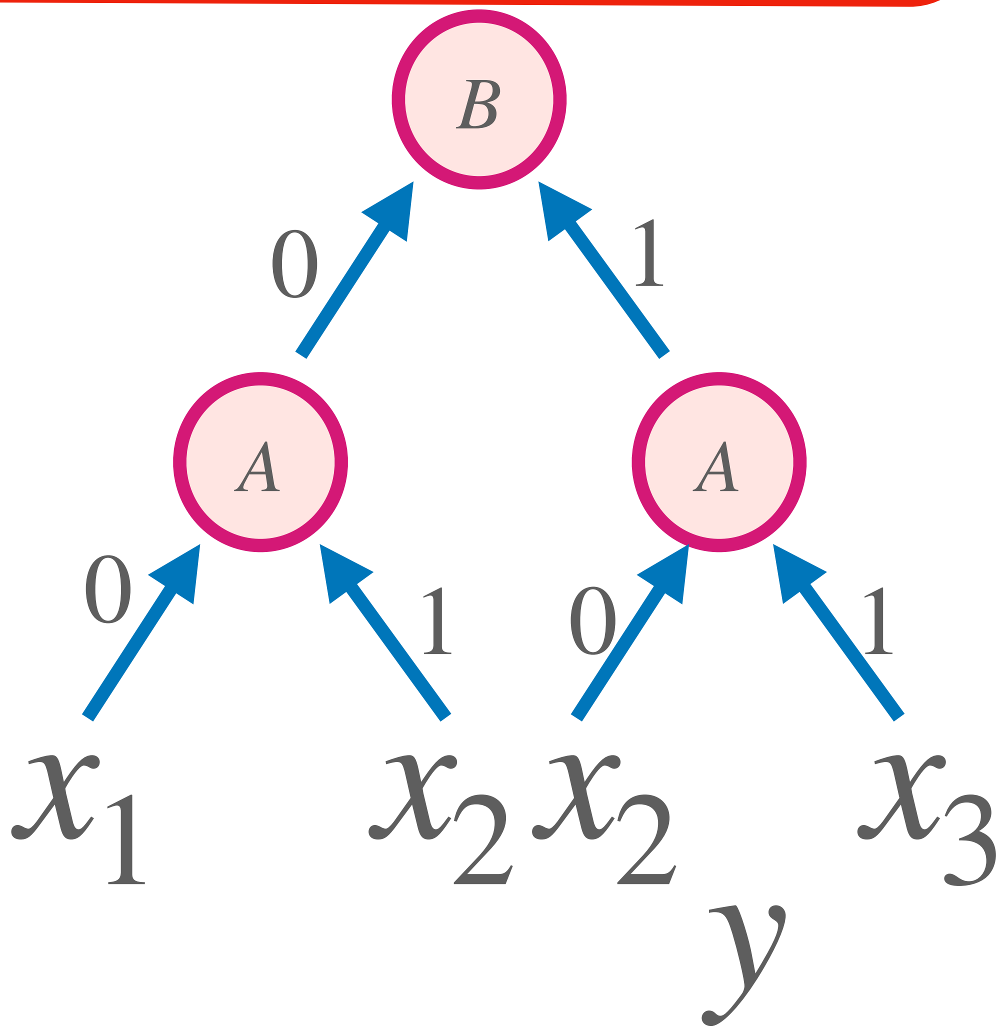


Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with ∨
- If Bob speaks at node v label it with ∧

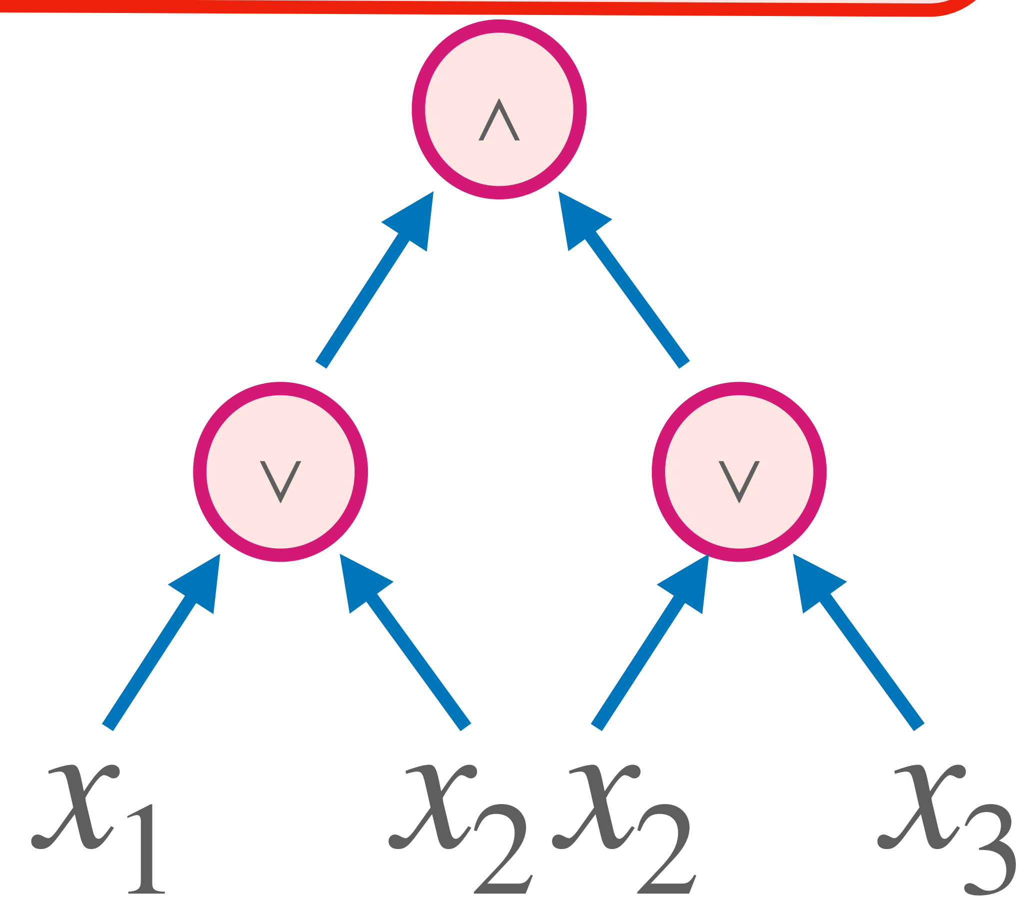


Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$

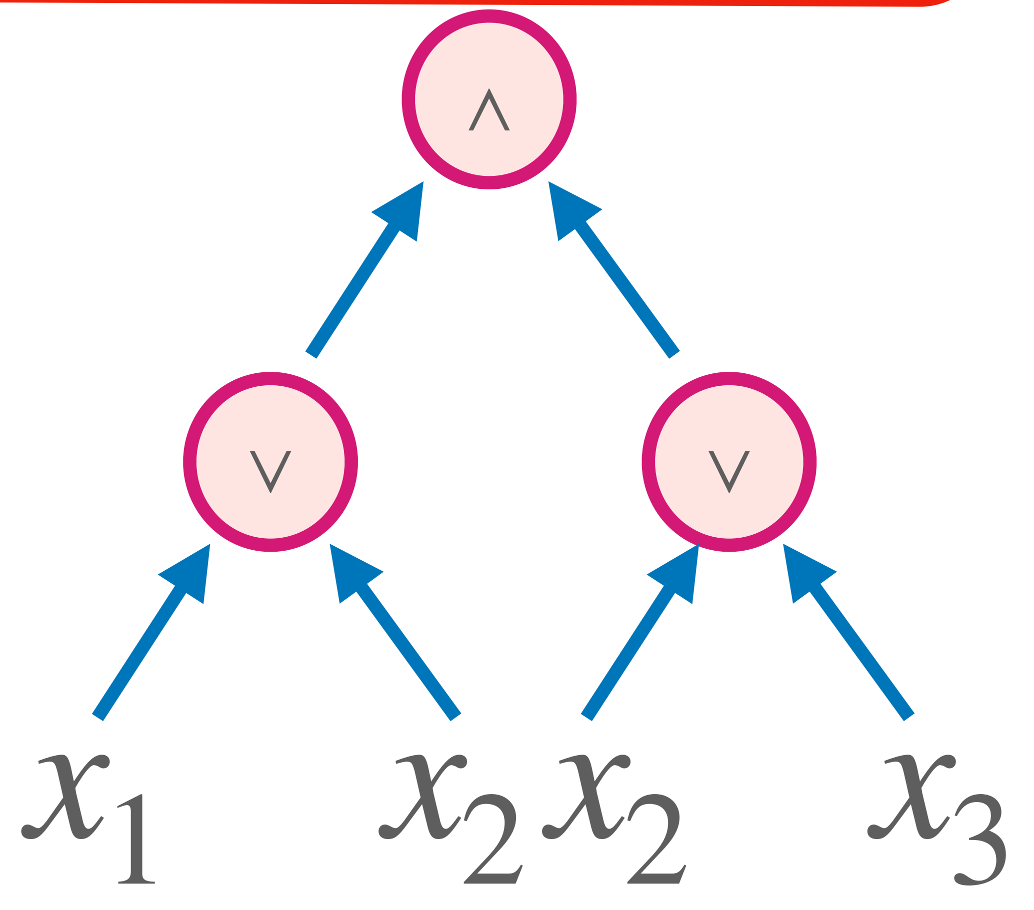


Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$



Correctness: Let R_v be the rectangle at node v.

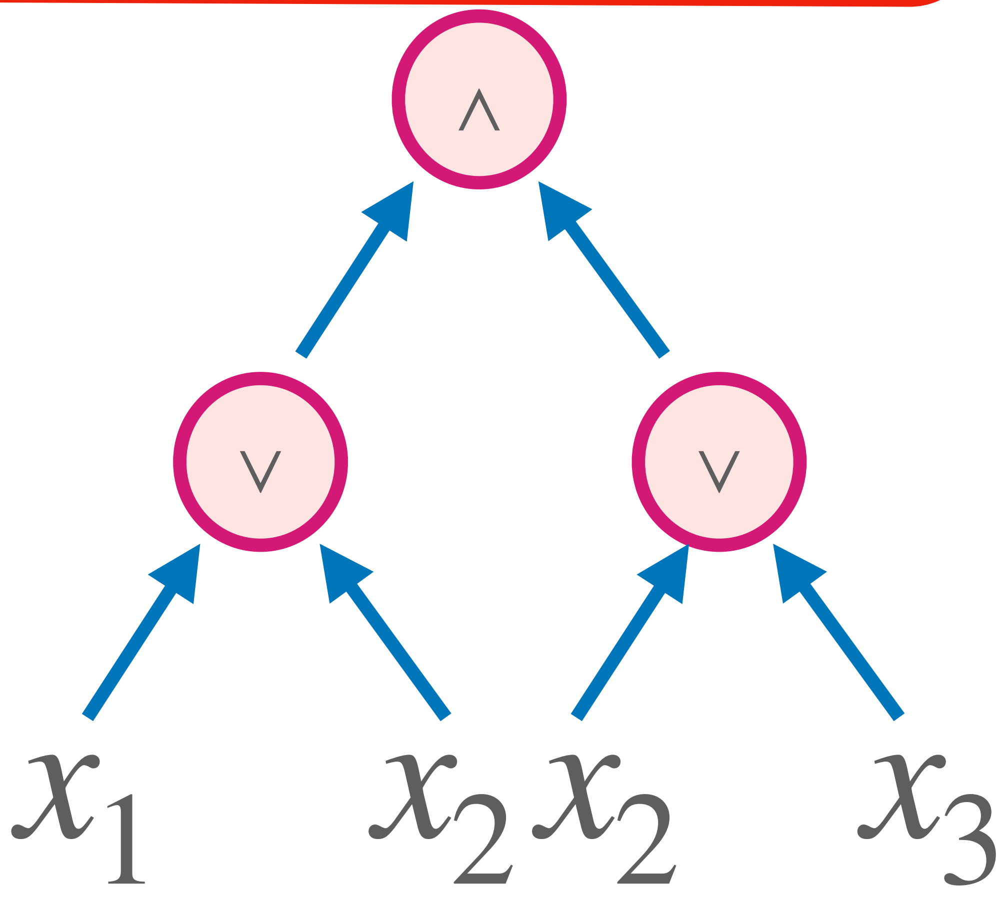
→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$

Karchmer-Wigderson and Circuits

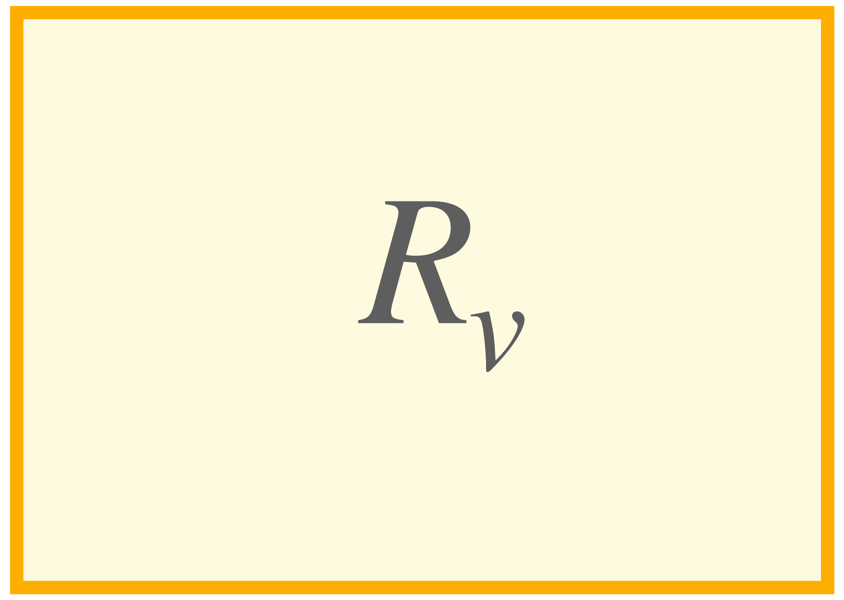
Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$



Correctness: Let R_v be the rectangle at node v.
→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$

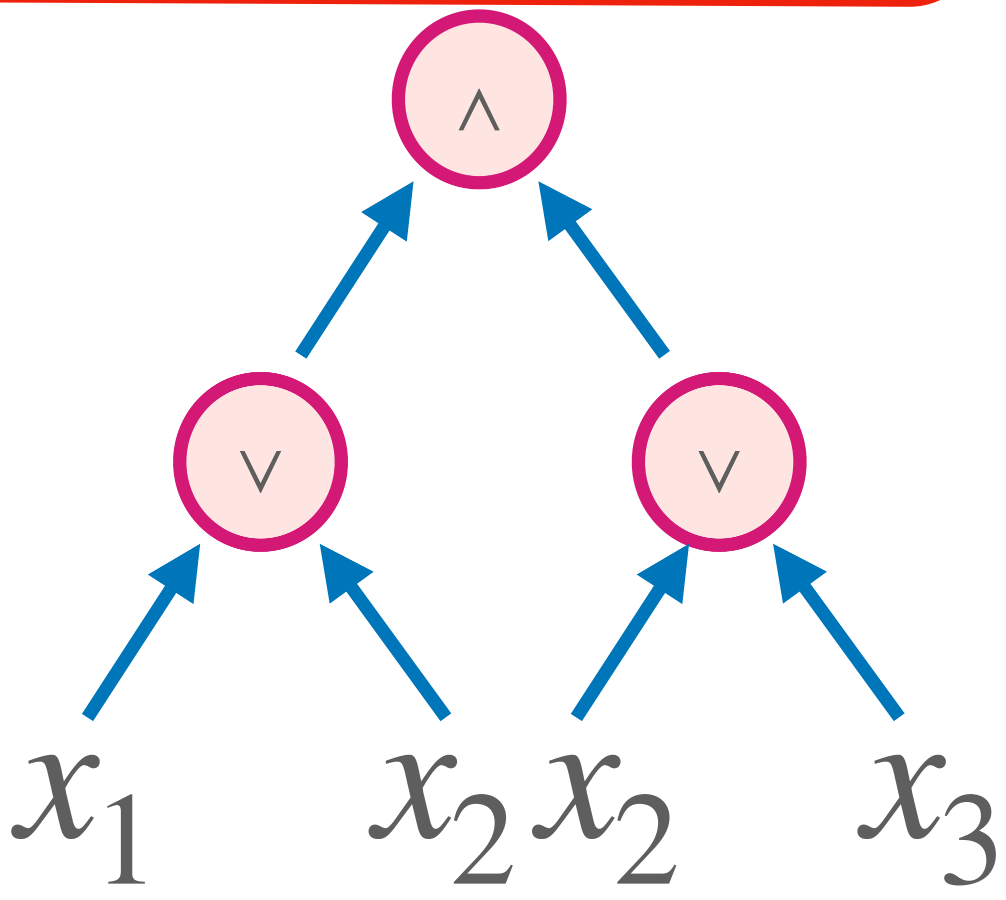


Karchmer-Wigderson and Circuits

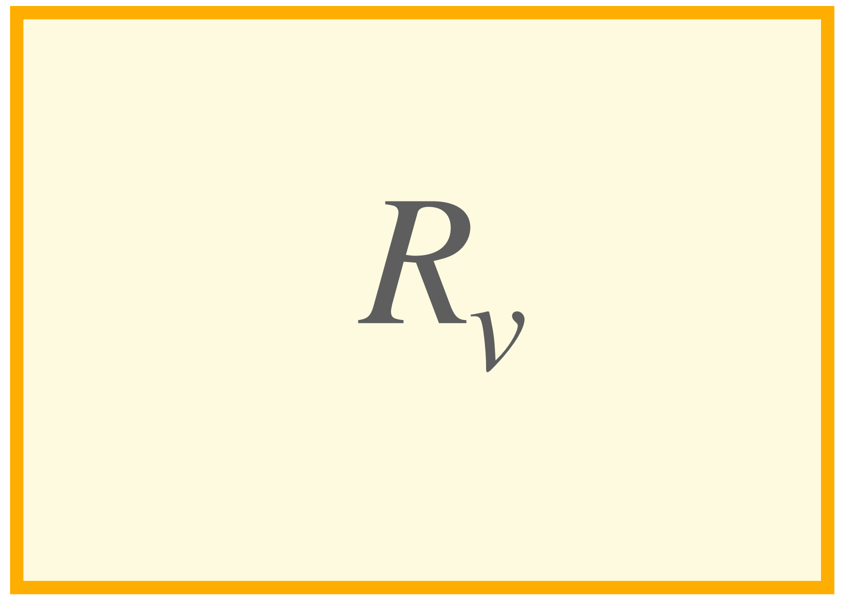
Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$



Correctness: Let R_v be the rectangle at node v.
→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$
By induction suppose this holds for C_{u_1}, C_{u_2}

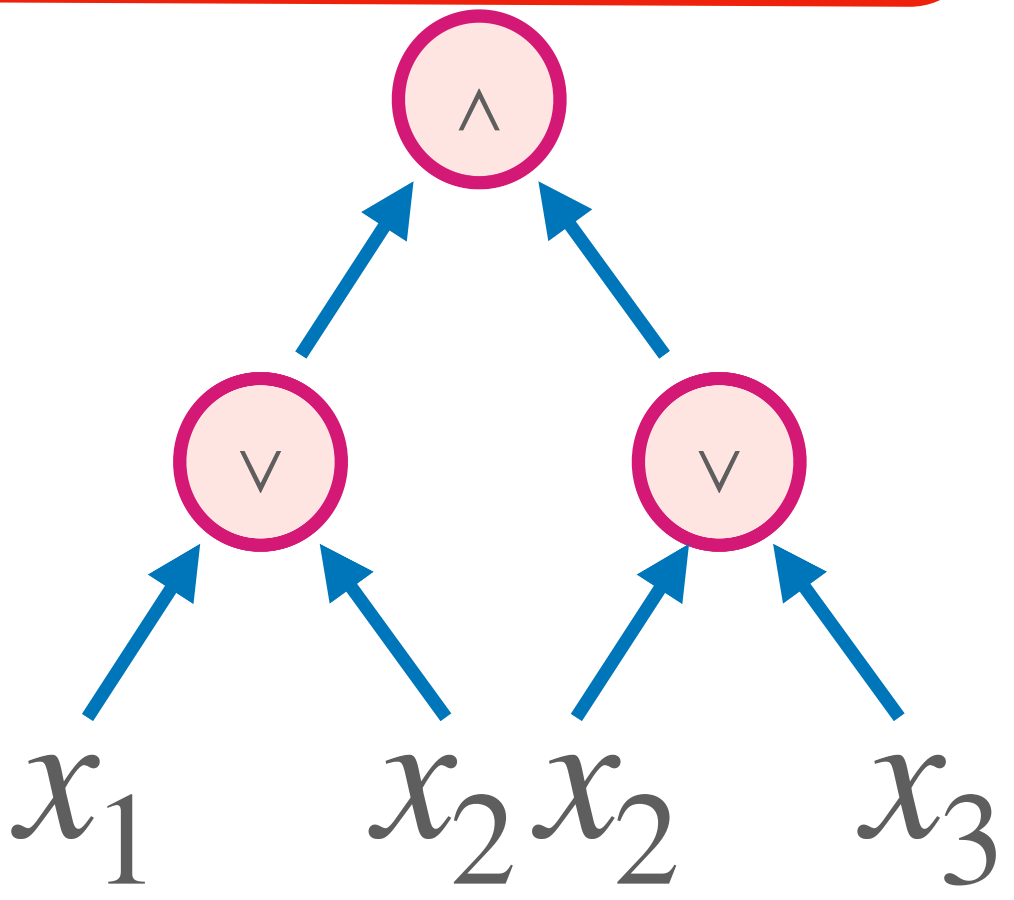


Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

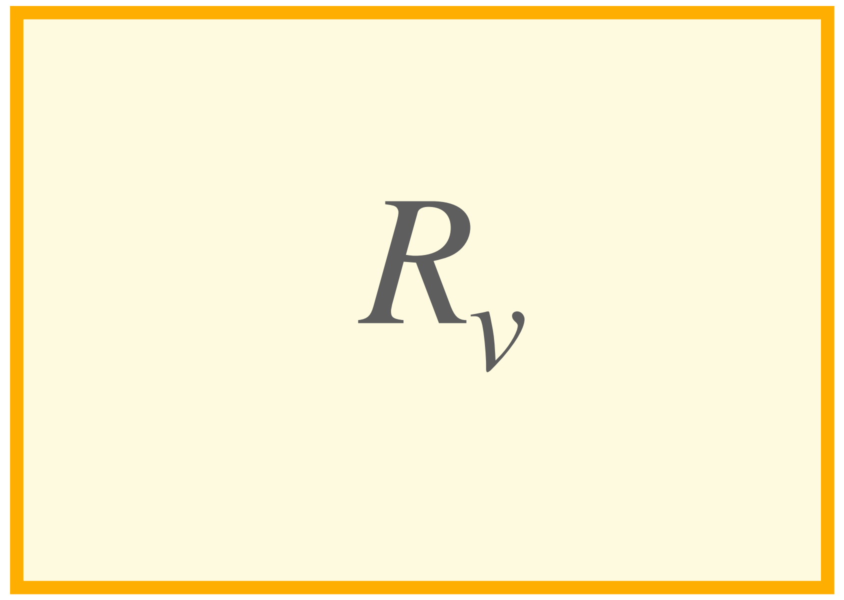
Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$



Correctness: Let R_v be the rectangle at node v.
→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$
By induction suppose this holds for C_{u_1}, C_{u_2}

- If $v = \wedge$ Bob speaks, partitioning R_v on columns. Let $(x, y) \in R_v$

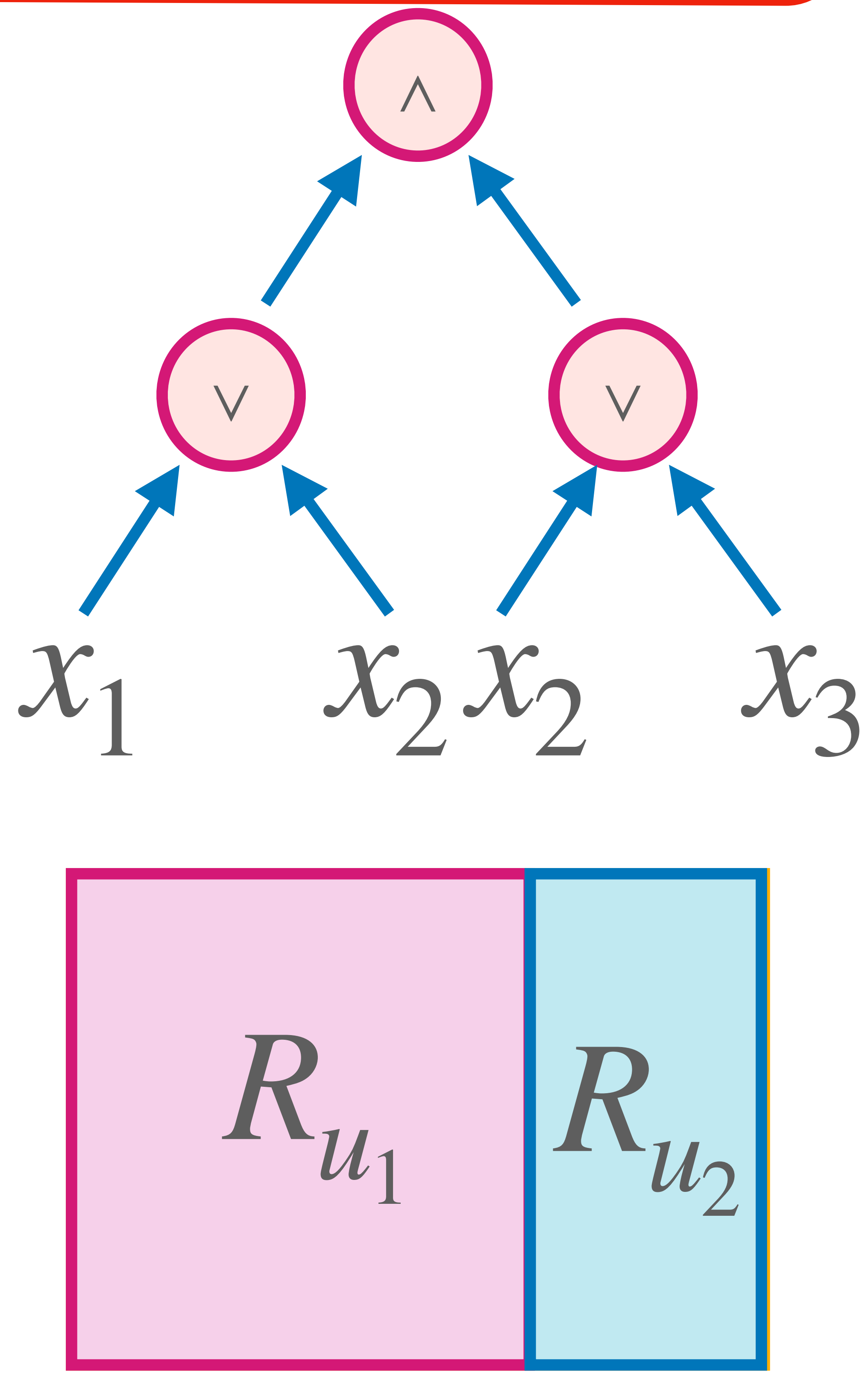


Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$



Correctness: Let R_v be the rectangle at node v.
→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$
By induction suppose this holds for C_{u_1}, C_{u_2}

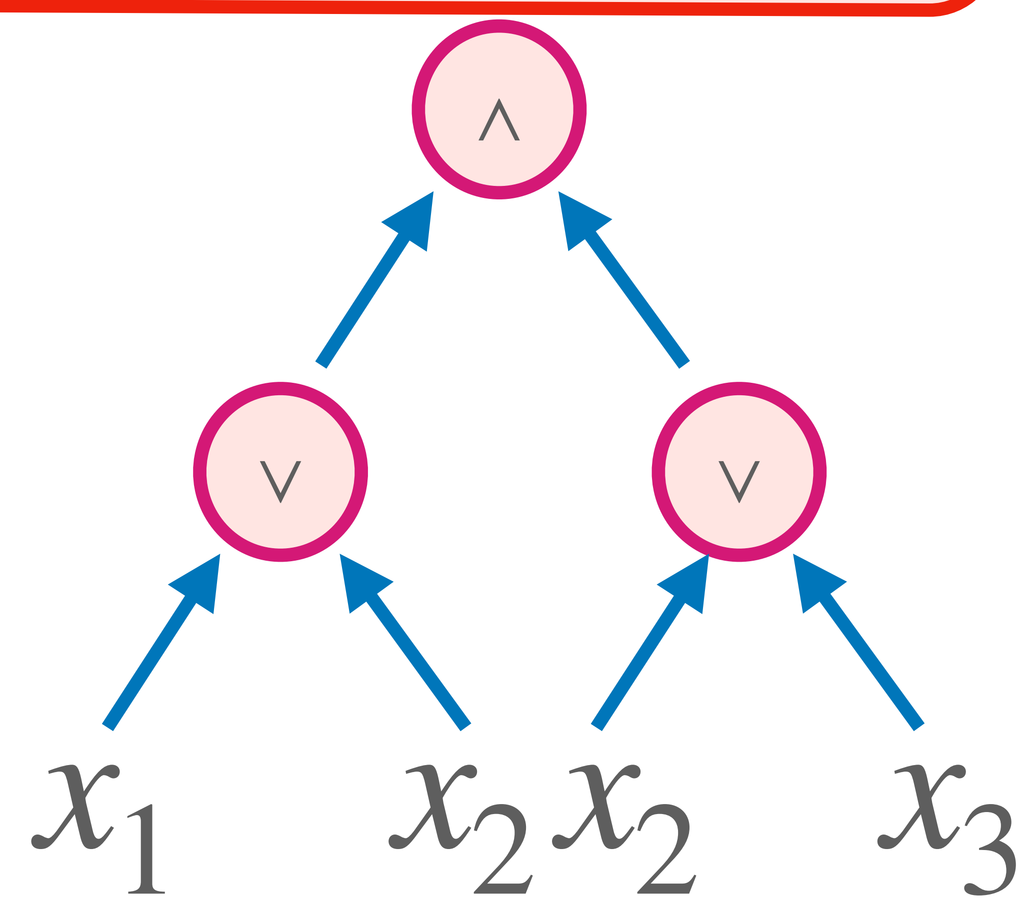
- If $v = \wedge$ Bob speaks, partitioning R_v on columns. Let $(x, y) \in R_v$

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$



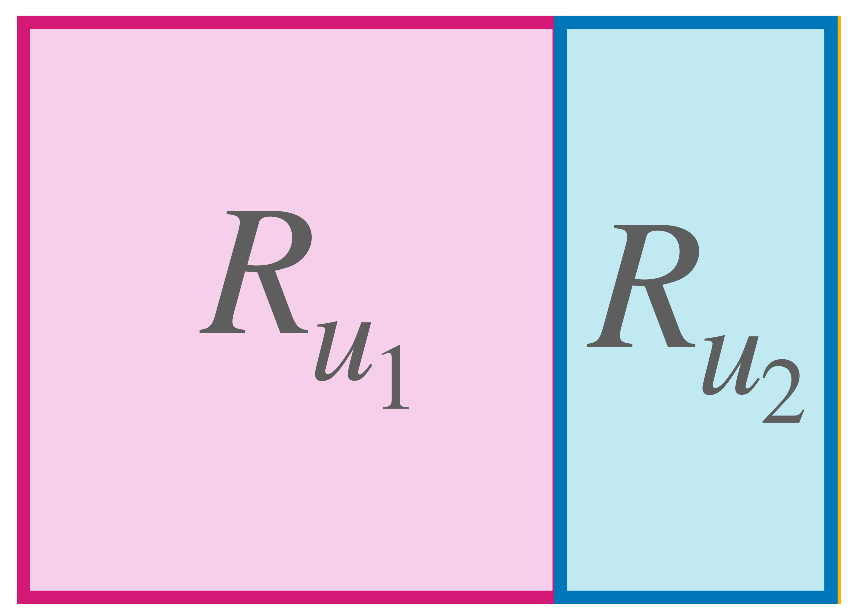
Correctness: Let R_v be the rectangle at node v.

→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$

By induction suppose this holds for C_{u_1}, C_{u_2}

- If $v = \wedge$ Bob speaks, partitioning R_v on columns. Let $(x, y) \in R_v$

  → $x \in R_{u_1}, R_{u_2} \implies C_{u_1}(x) = C_{u_2}(x) = 1$

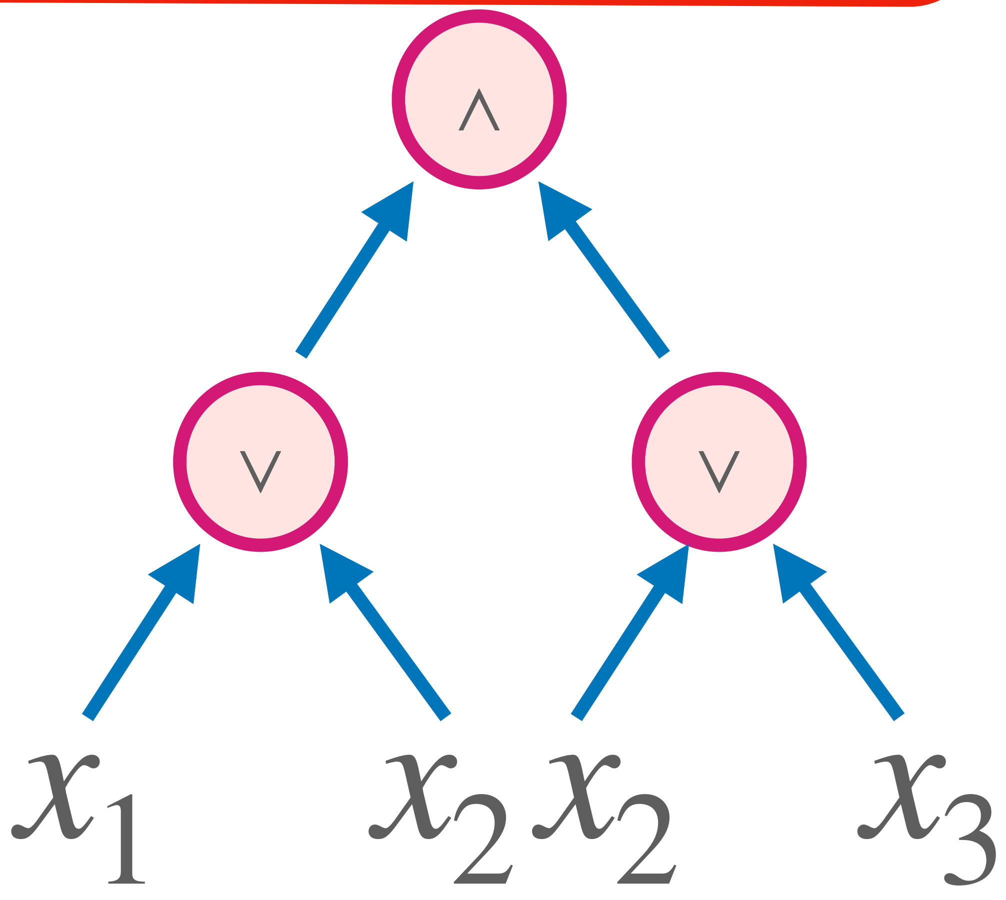


Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$

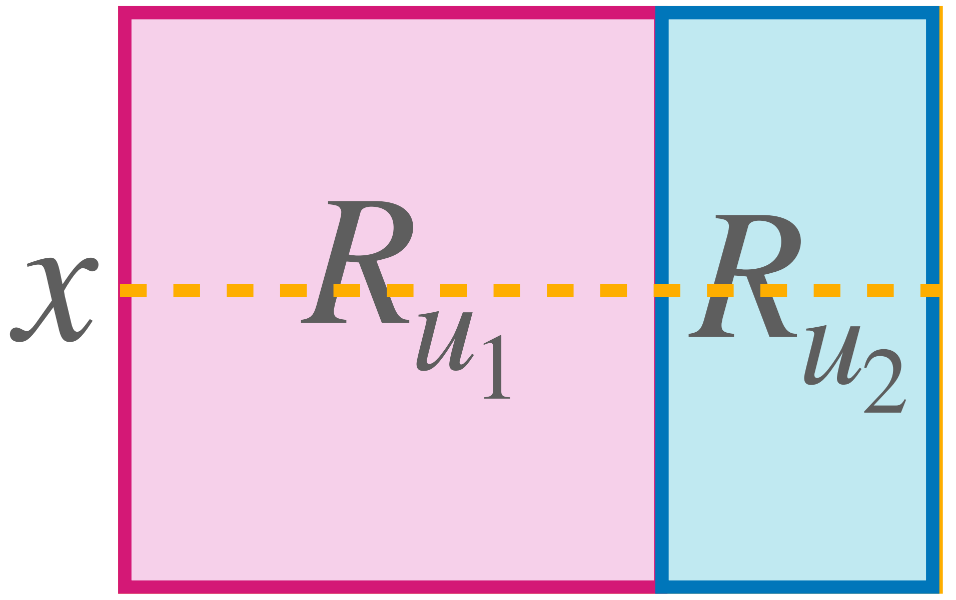


Correctness: Let R_v be the rectangle at node v.

→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$

By induction suppose this holds for C_{u_1}, C_{u_2}

- If $v = \wedge$ Bob speaks, partitioning R_v on columns. Let $(x, y) \in R_v$
  $\to x \in R_{u_1}, R_{u_2} \implies C_{u_1}(x) = C_{u_2}(x) = 1$

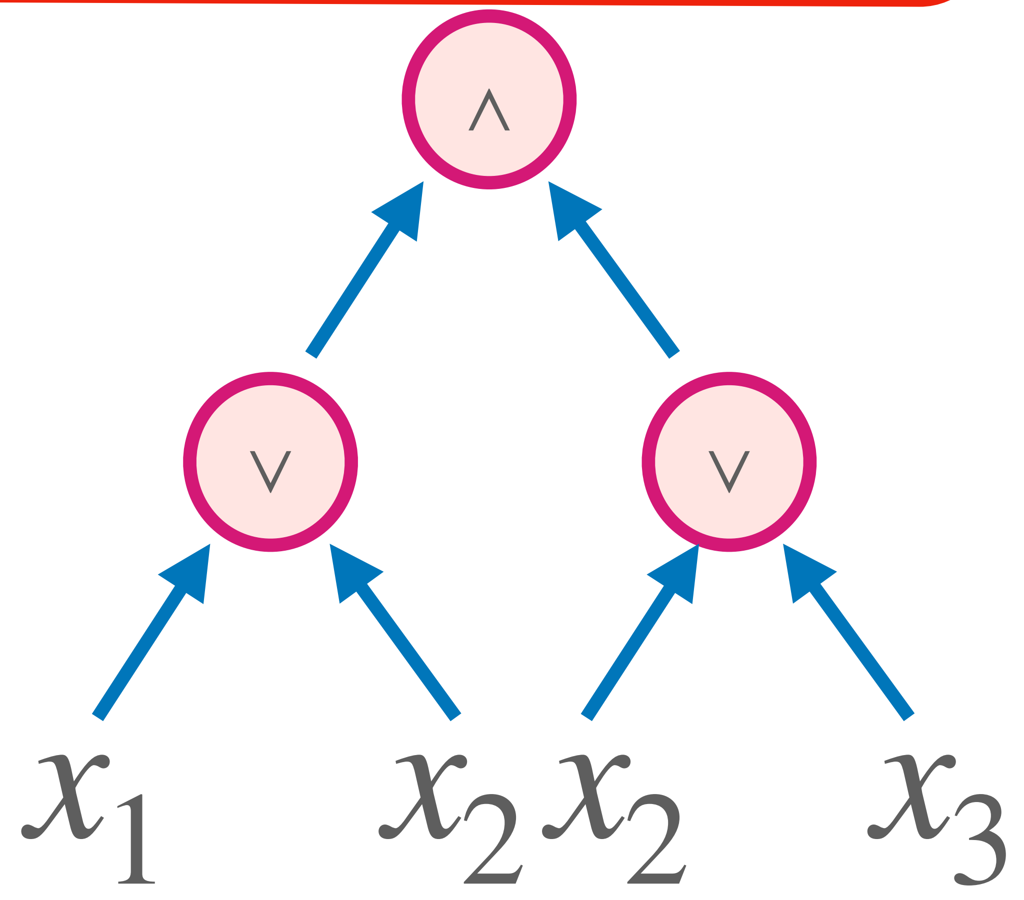


Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f: \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

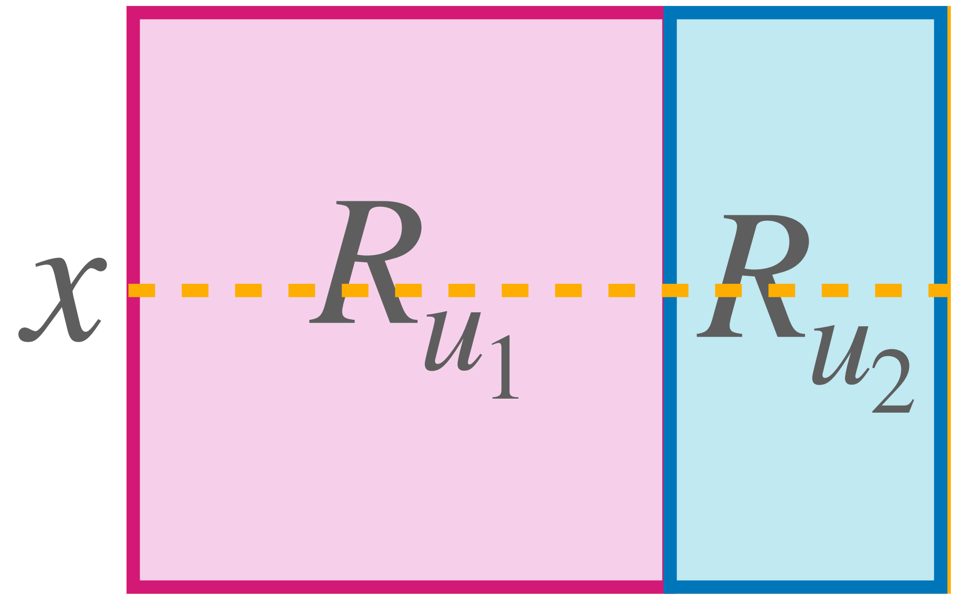
Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$



Correctness: Let R_v be the rectangle at node v.
→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$
By induction suppose this holds for C_{u_1}, C_{u_2}

- If $v = \wedge$ Bob speaks, partitioning R_v on columns. Let $(x, y) \in R_v$
  → $x \in R_{u_1}, R_{u_2} \implies C_{u_1}(x) = C_{u_2}(x) = 1$
  → $y \in R_{u_1}$ or $y \in R_{u_2} \implies C_{u_1}(y) = 0$ or $C_{u_2}(y) = 0$

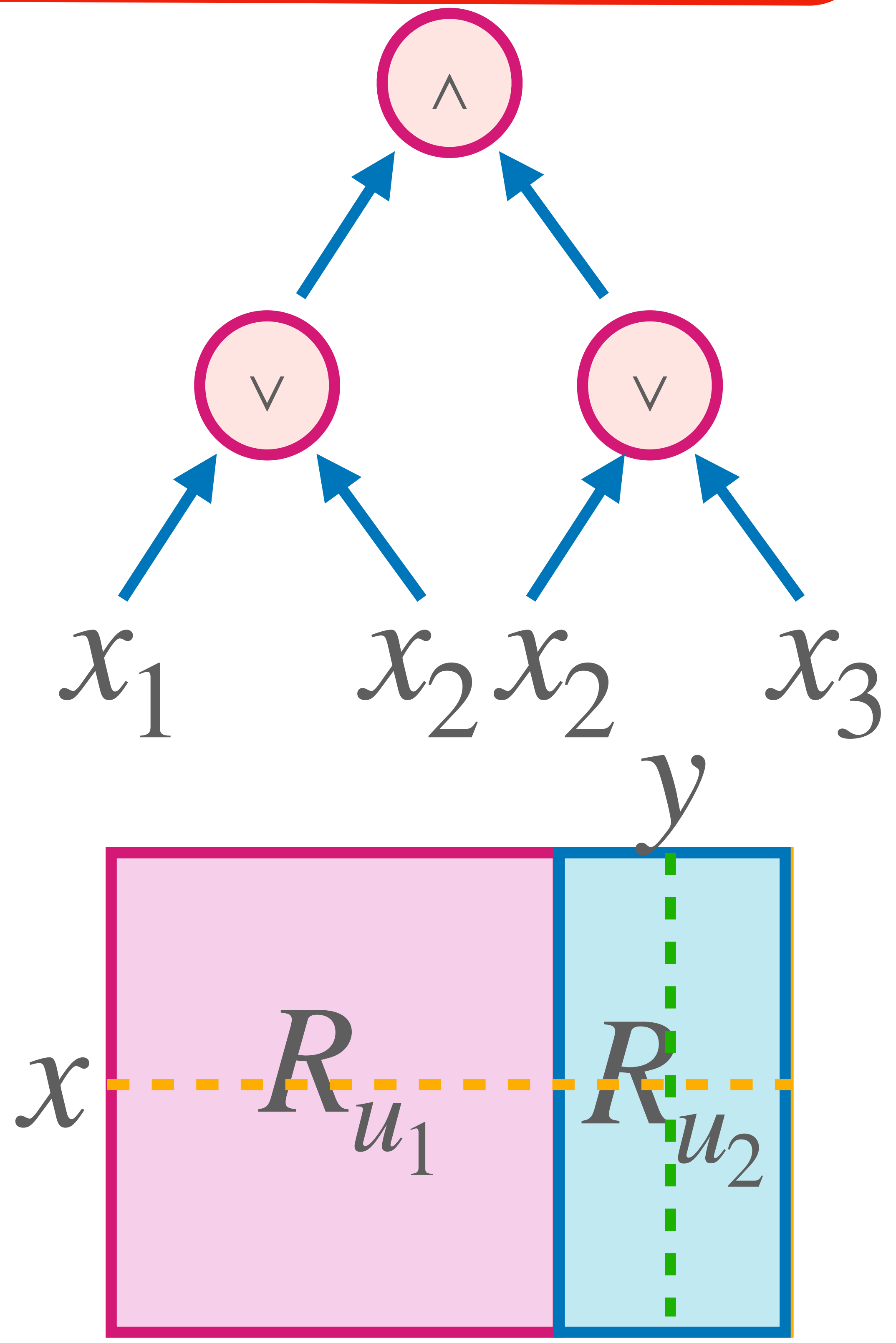


Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$



Correctness: Let R_v be the rectangle at node v.

→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$

By induction suppose this holds for C_{u_1}, C_{u_2}

- If $v = \wedge$ Bob speaks, partitioning R_v on columns. Let $(x, y) \in R_v$

  → $x \in R_{u_1}, R_{u_2} \implies C_{u_1}(x) = C_{u_2}(x) = 1$

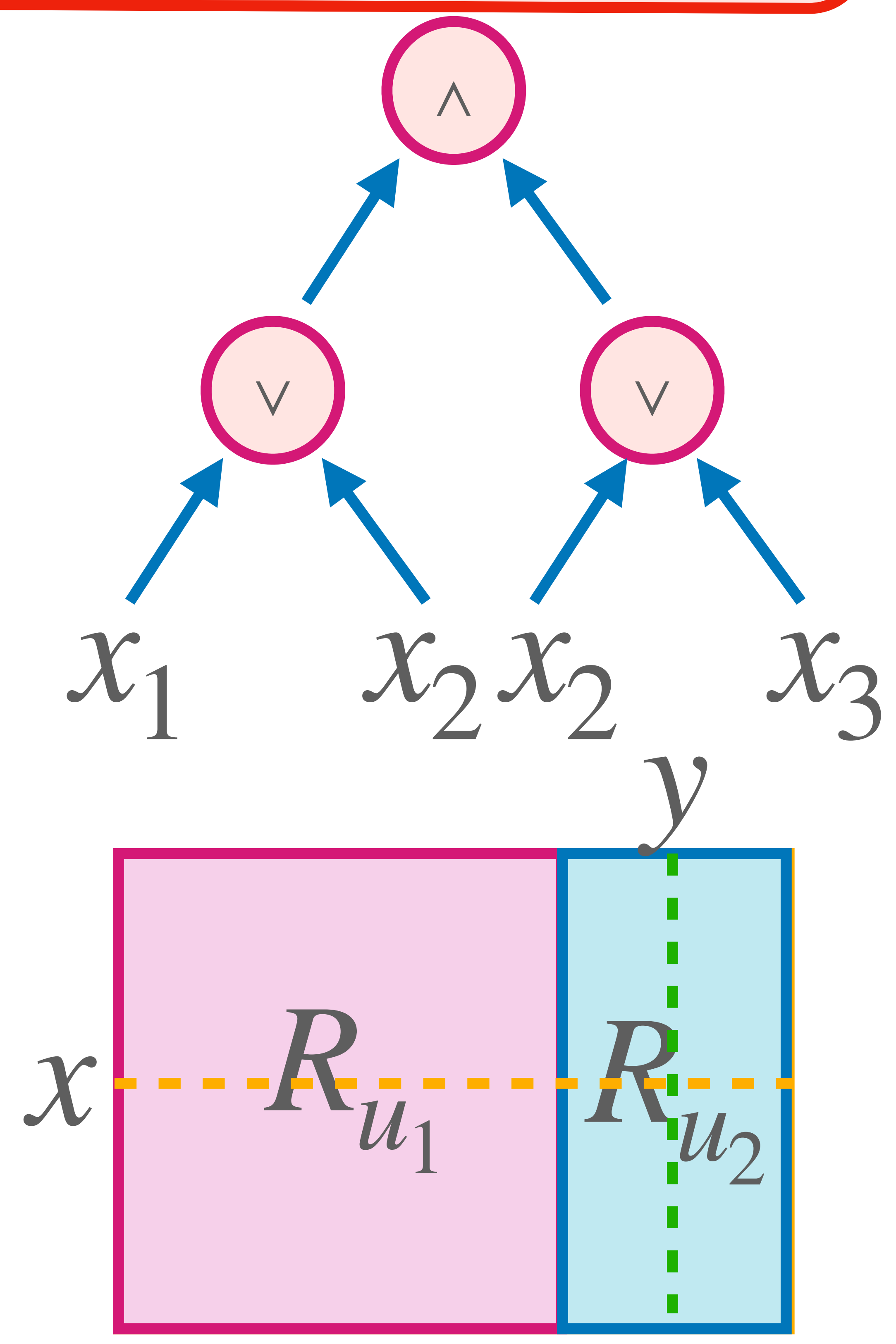
  → $y \in R_{u_1}$ or $y \in R_{u_2} \implies C_{u_1}(y) = 0$ or $C_{u_2}(y) = 0$

Karchmer-Wigderson and Circuits

Karchmer-Wigderson:
Let $f : \{0,1\}^n \rightarrow \{0,1\}$ be monotone. A cc-protocol for mKW$_f$ is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$



Correctness: Let R_v be the rectangle at node v.
→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$
By induction suppose this holds for C_{u_1}, C_{u_2}

- If $v = \wedge$ Bob speaks, partitioning R_v on columns. Let $(x, y) \in R_v$
  → $x \in R_{u_1}, R_{u_2} \implies C_{u_1}(x) \wedge C_{u_2}(x) = 1$
  → $y \in R_{u_1}$ or $y \in R_{u_2} \implies C_{u_1}(y) \wedge C_{u_2}(y) = 0$

Karchmer-Wigderson and Circuits

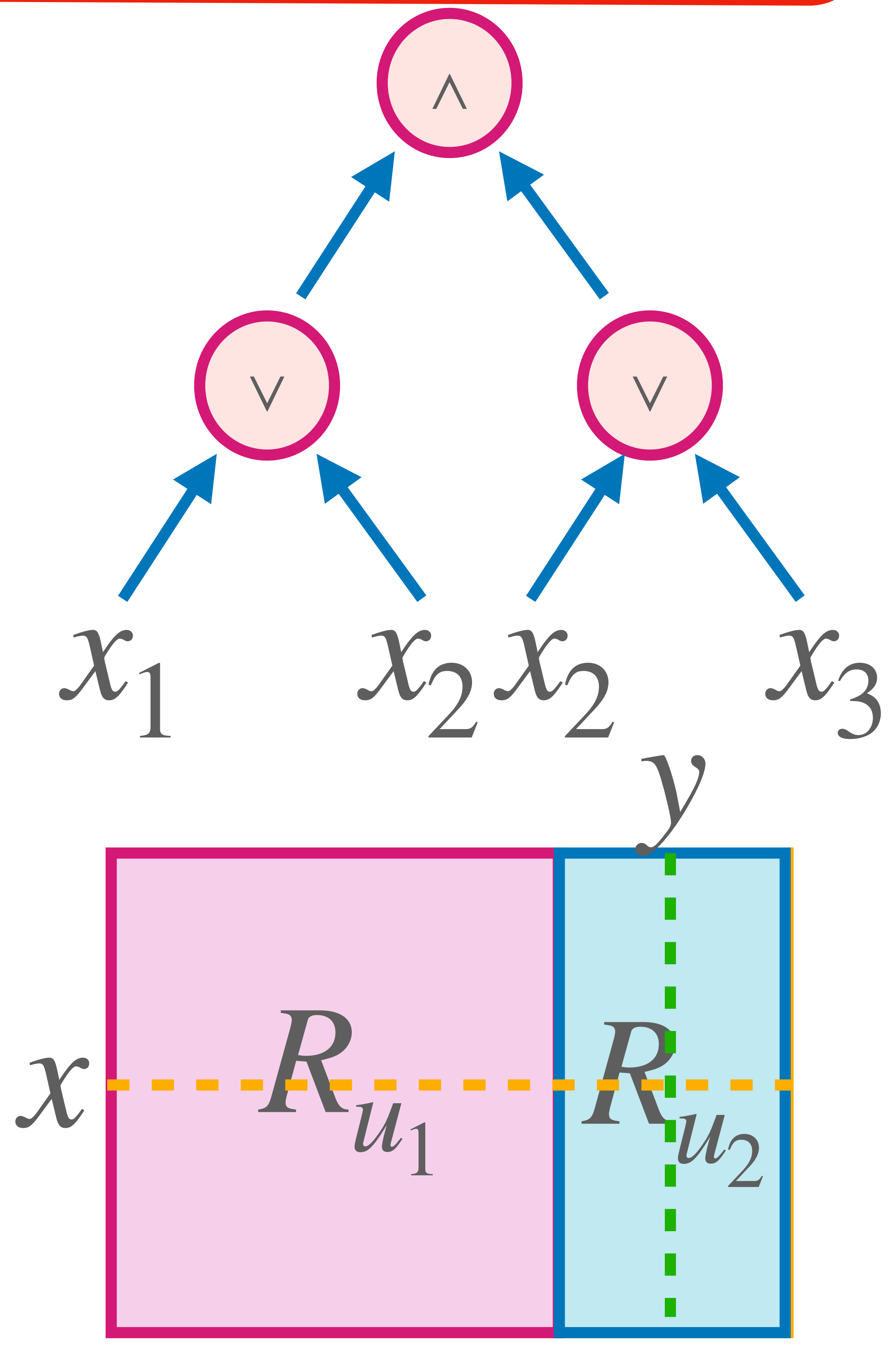
Karchmer-Wigderson:
Let $f : \{0,1\}^n \to \{0,1\}$ be monotone. A cc-protocol for mKW_f is equivalent a monotone formula computing f.

Proof→: From a protocol construct a monotone formula

- If Alice speaks at node v label it with $\vee$
- If Bob speaks at node v label it with $\wedge$

Correctness: Let R_v be the rectangle at node v.
→ Show sub-circuit C_v has $C_v(x) = 1$, $C_v(y) = 0$ for $(x, y) \in R_v$
By induction suppose this holds for C_{u_1}, C_{u_2}

- If $v = \wedge$ Bob speaks, partitioning R_v on columns.
- If $v = \vee$ Alice speaks partitioning R_v on rows; symmetric argument.



The False Clause Search Problem

The False Clause search problem characterizes the complexity of proof systems

The False Clause Search Problem

The False Clause search problem characterizes the complexity of proof systems

False Clause Search Problem: Let $F = C_1 \wedge \ldots \wedge C_m$ be an unsat CNF.
Let $X \times Y$ be a partition of $[n]$. $\text{Search}_F^{X,Y} \subseteq \{0,1\}^X \times \{0,1\}^Y \times [m]$:
Given $(x, y) \in \{0,1\}^{X \times Y}$ output $i \in [m]$ such that $C_i(x, y) = 0$

The False Clause Search Problem

The False Clause search problem characterizes the complexity of proof systems

False Clause Search Problem: Let $F = C_1 \wedge \ldots \wedge C_m$ be an unsat CNF.
Let $X \times Y$ be a partition of $[n]$. $\text{Search}_F^{X,Y} \subseteq \{0,1\}^X \times \{0,1\}^Y \times [m]$:
Given $(x, y) \in \{0,1\}^{X \times Y}$ output $i \in [m]$ such that $C_i(x, y) = 0$

Intuition: Every refutation must prove that for every assignment x to F there is a falsified clause of F

The False Clause Search Problem

Thm:
A a tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

Resolution:

$(x_2 \vee y)$ $(\bar{x}_1 \vee \bar{y})$ $(\bar{x}_2)$ $(x_1 \vee \bar{y})$

The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\mathrm{Search}_F^{X,Y}$

Resolution:

Derive new clauses from old ones using:

Resolution rule:

$$\frac{C_1 \vee x, \quad C_2 \vee \neg x}{C_1 \vee C_2}$$

$(x_2 \vee y)\ (\bar{x}_1 \vee \bar{y})\ (\bar{x}_2)\ (x_1 \vee \bar{y})$

The False Clause Search Problem

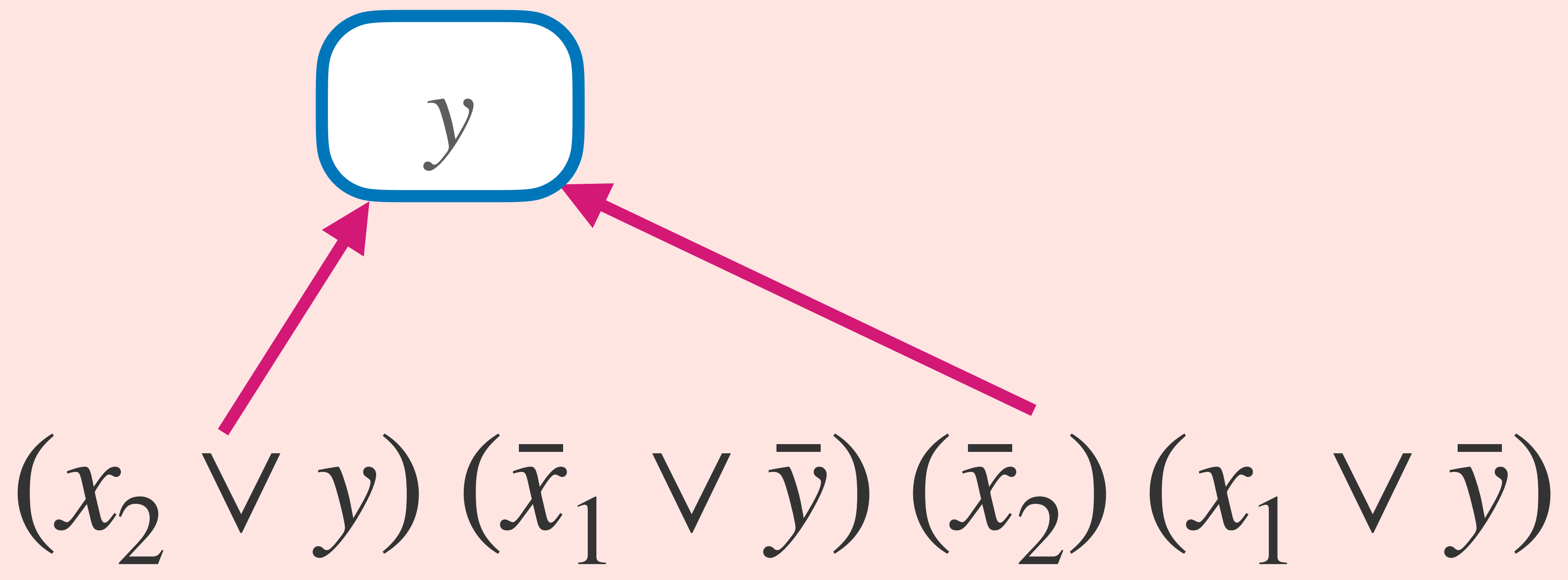
Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

Resolution:

Derive new clauses from old ones using:

Resolution rule:

$$\frac{C_1 \vee x, \quad C_2 \vee \neg x}{C_1 \vee C_2}$$



The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

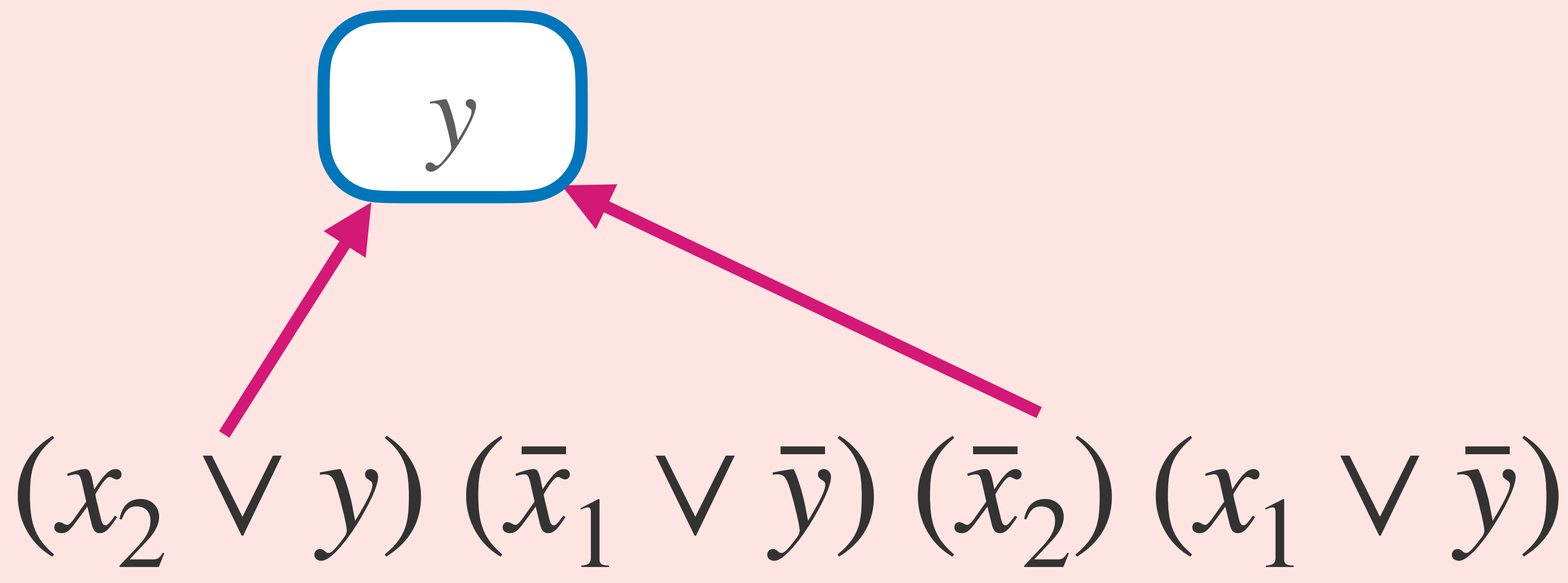
Resolution:

Derive new clauses from old ones using:

Resolution rule:

$$\frac{C_1 \vee x, \quad C_2 \vee \neg x}{C_1 \vee C_2}$$

Goal: derive empty clause Λ



The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

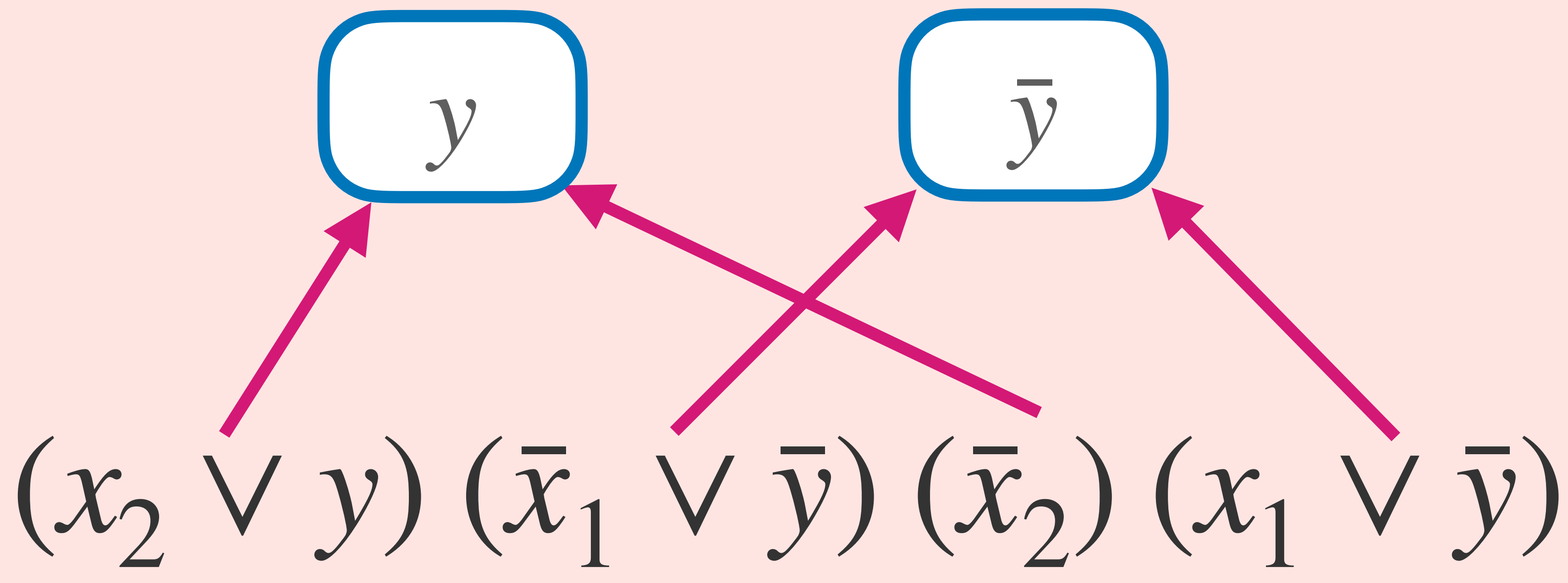
Resolution:

Derive new clauses from old ones using:

Resolution rule:

$$\frac{C_1 \vee x, \quad C_2 \vee \neg x}{C_1 \vee C_2}$$

Goal: derive empty clause Λ



The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

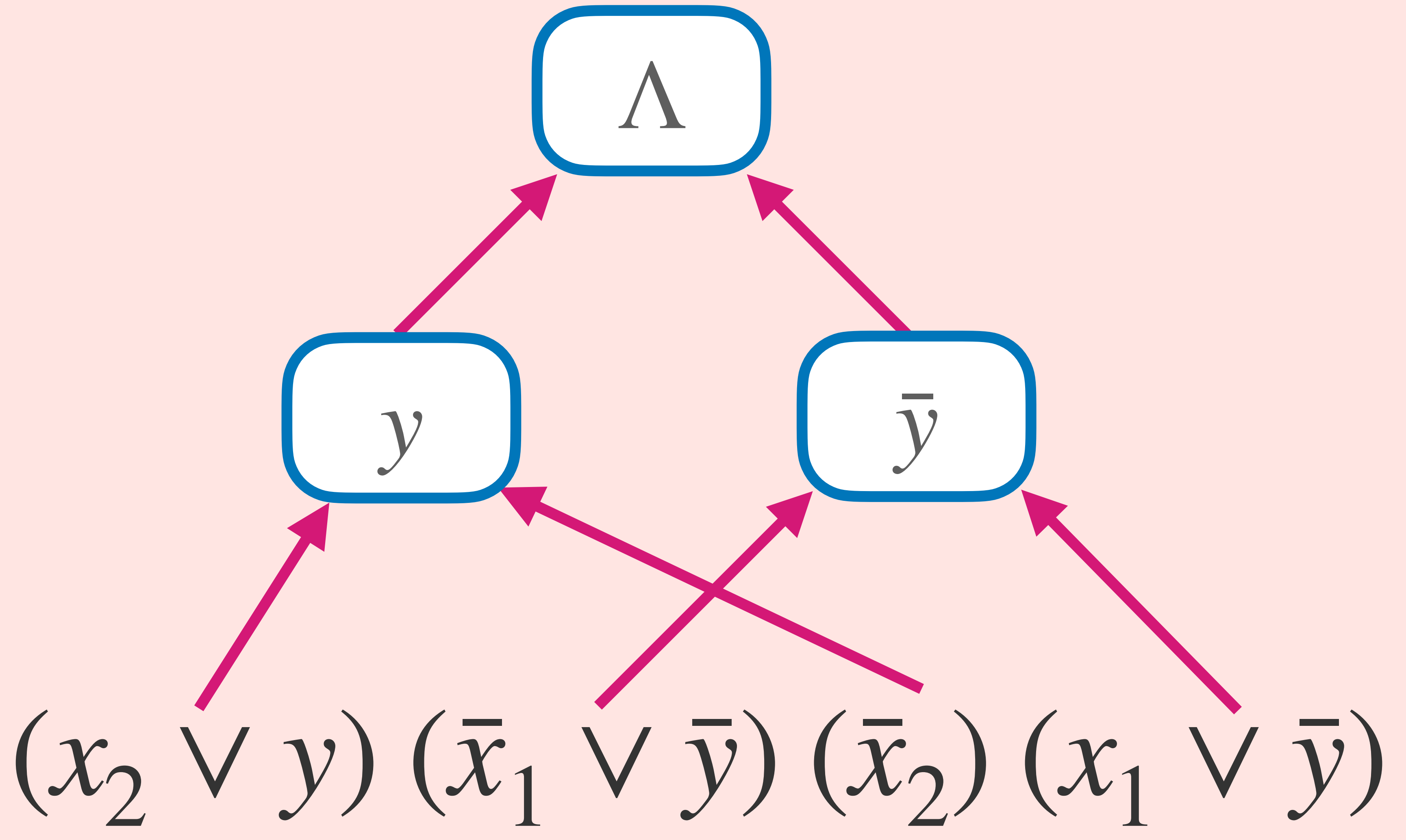
Resolution:

Derive new clauses from old ones using:

Resolution rule:

$$\frac{C_1 \vee x, \quad C_2 \vee \neg x}{C_1 \vee C_2}$$

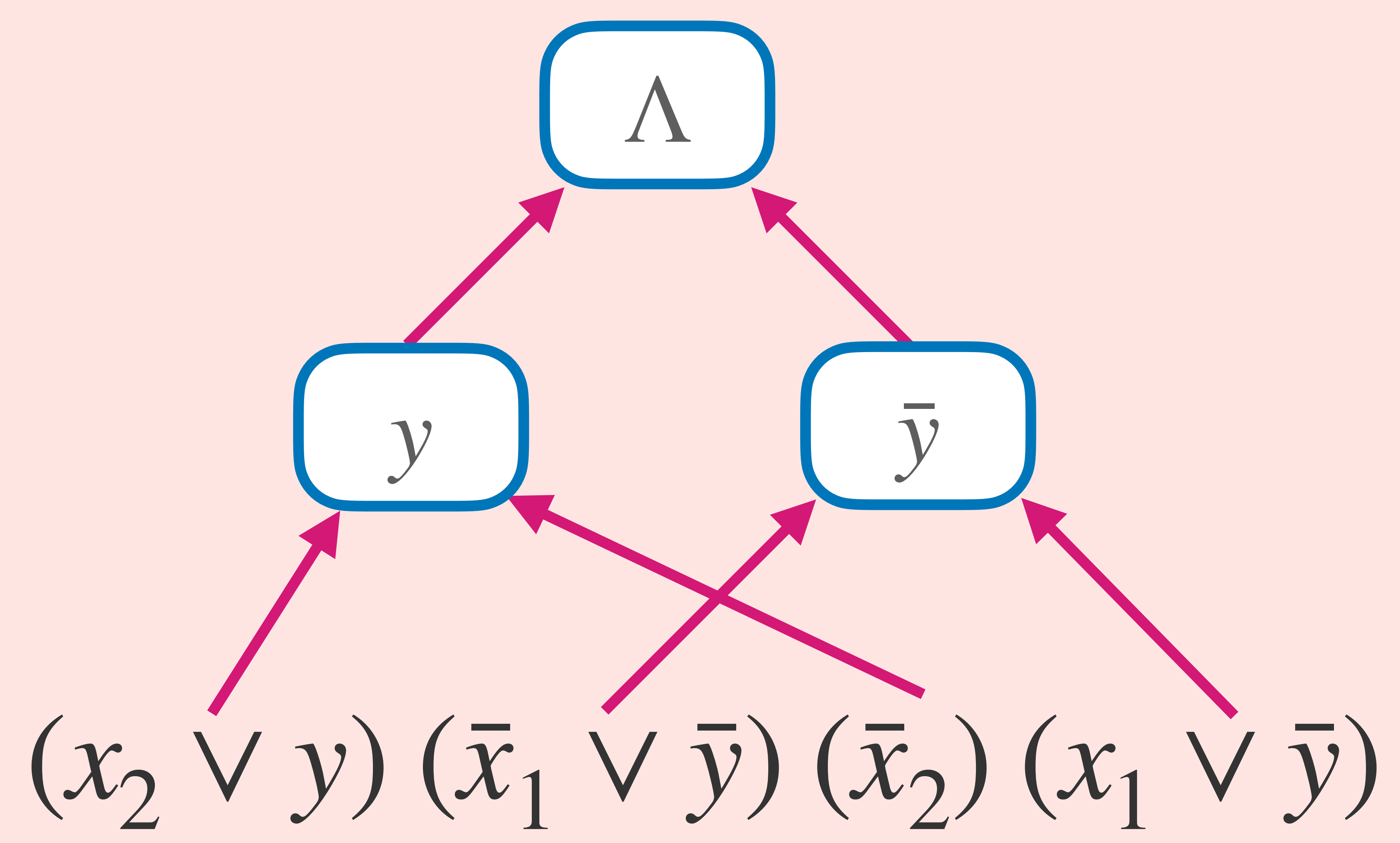
Goal: derive empty clause Λ



The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

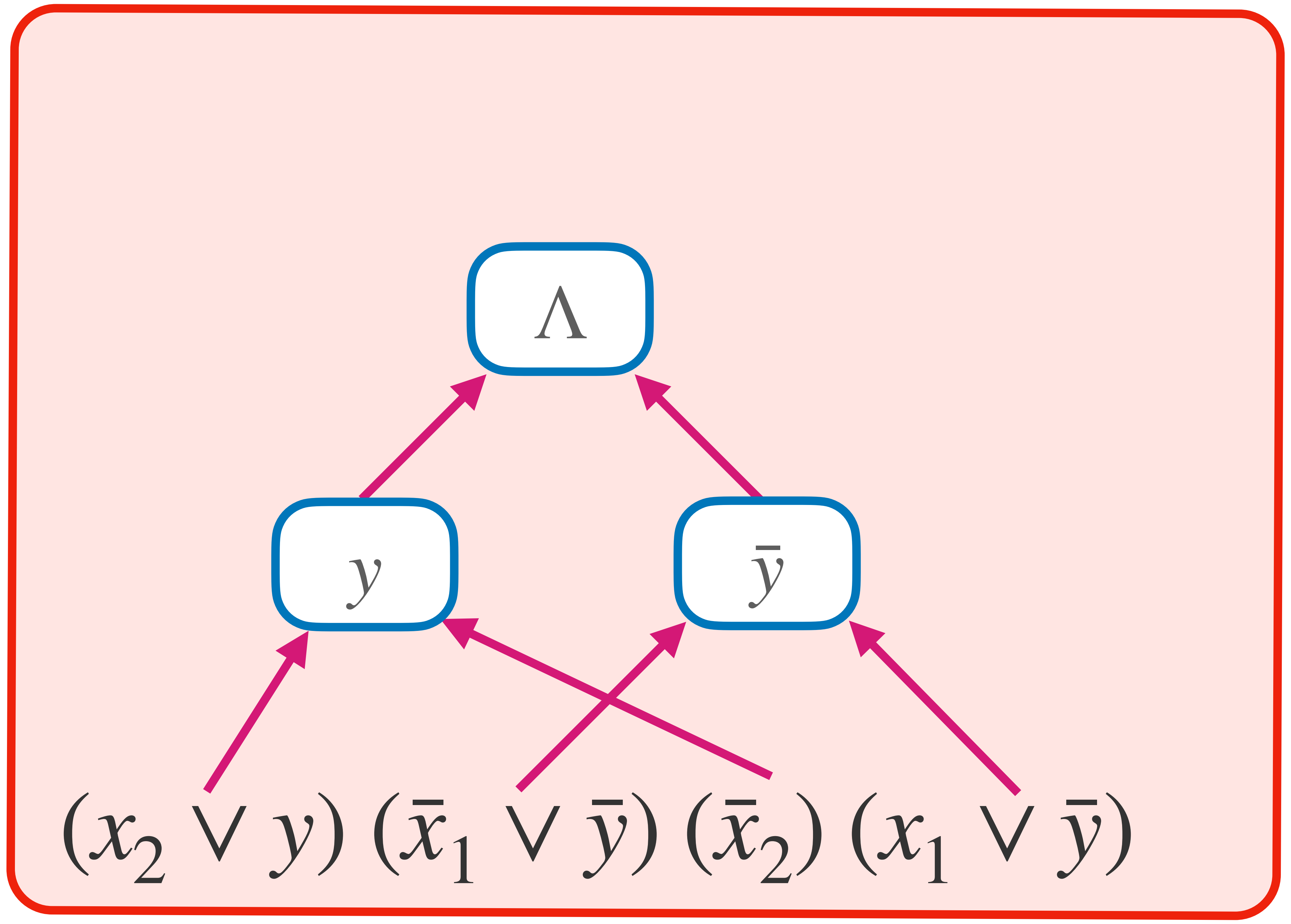
Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf



The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf

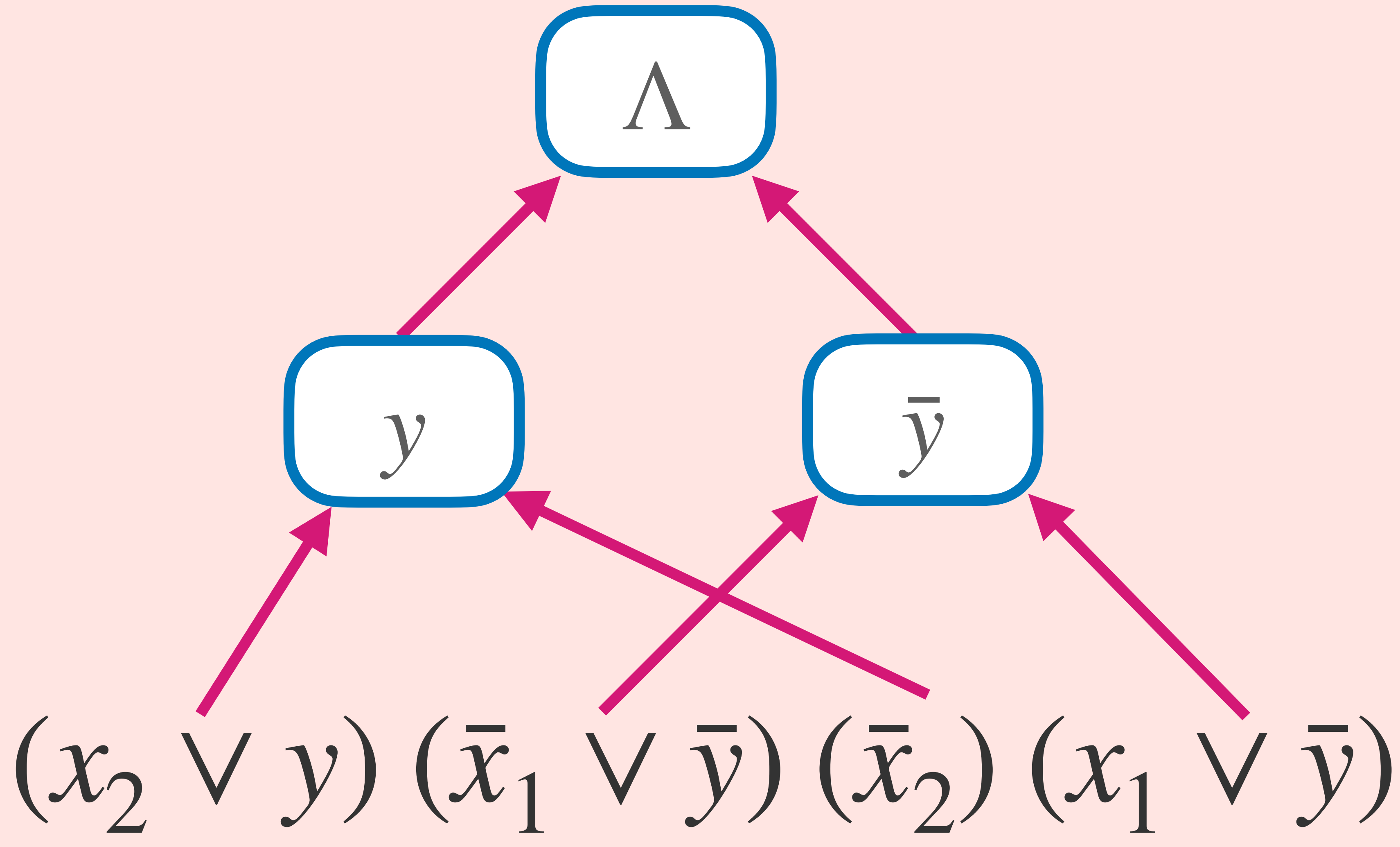


The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf

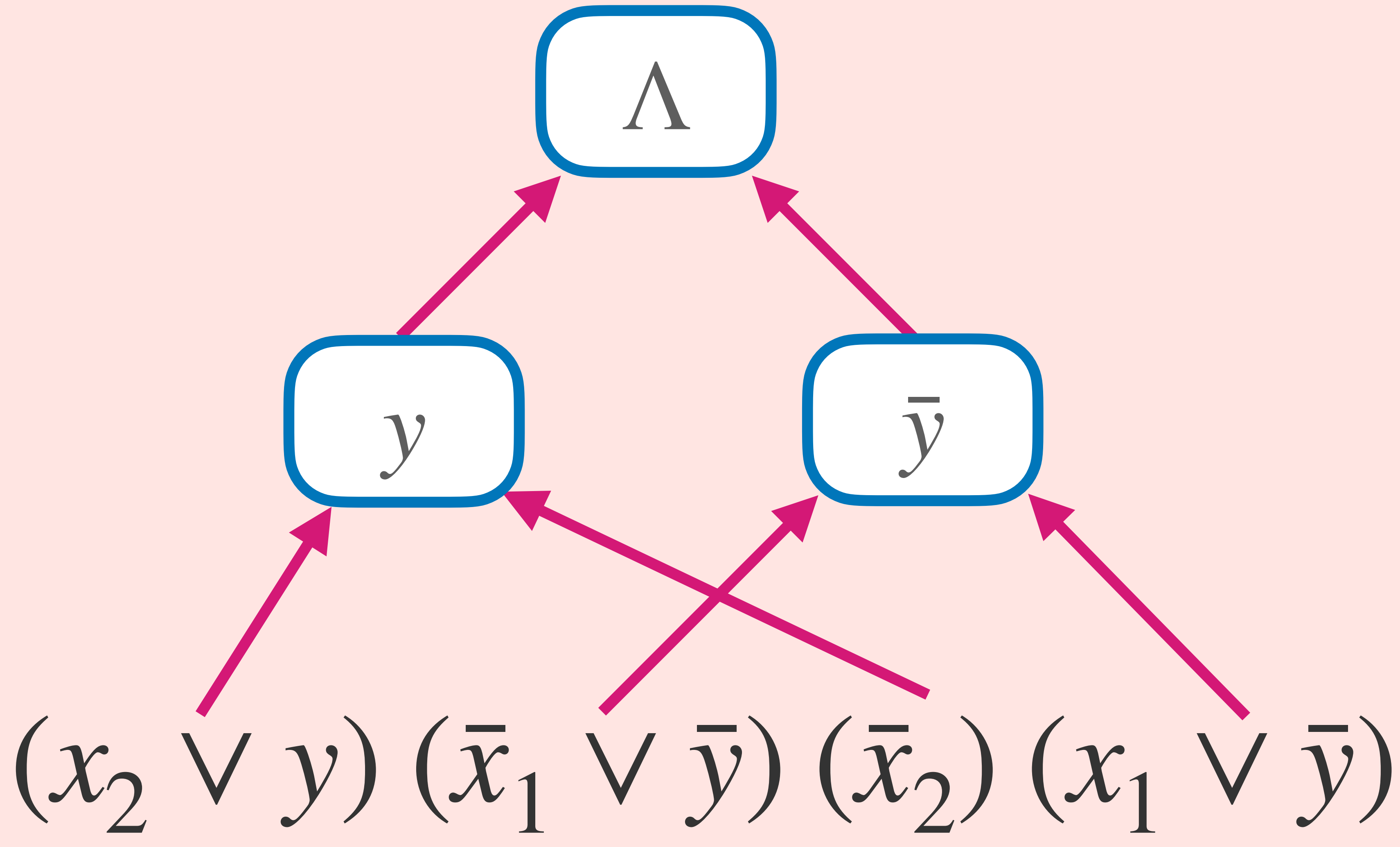
Invariant: If C is the current clause $C(x, y) = 0$



The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf



Invariant: If C is the current clause $C(x, y) = 0$

If C was derived from C', C'' by resolving on

- x_i then Alice sends x_i to Bob
- y_i then Bob sends y_i to Alice

The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

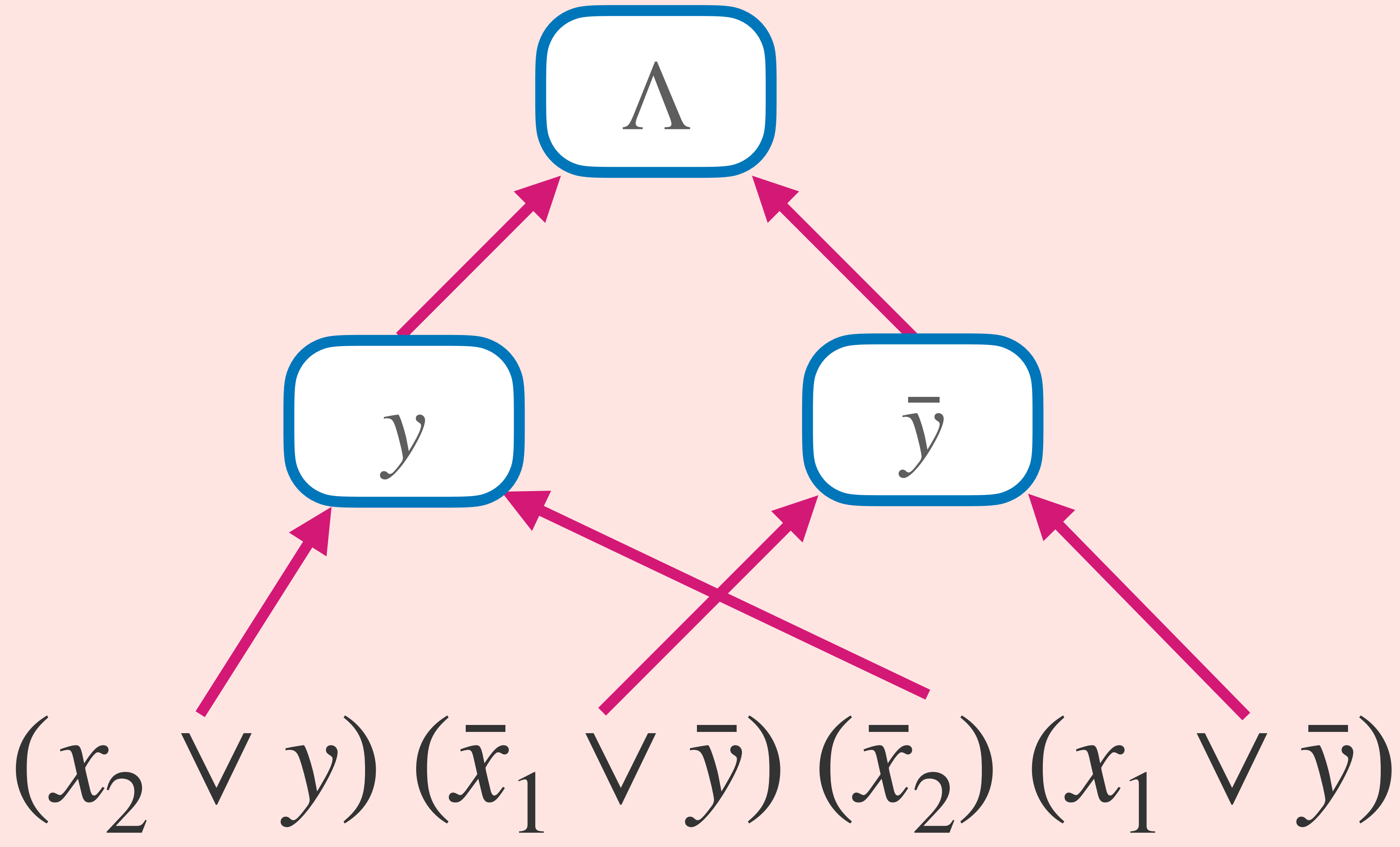
Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf

Invariant: If C is the current clause $C(x, y) = 0$

If C was derived from C', C'' by resolving on

- x_i then Alice sends x_i to Bob
- y_i then Bob sends y_i to Alice

Move to the child C' or C'' containing the falsified literal

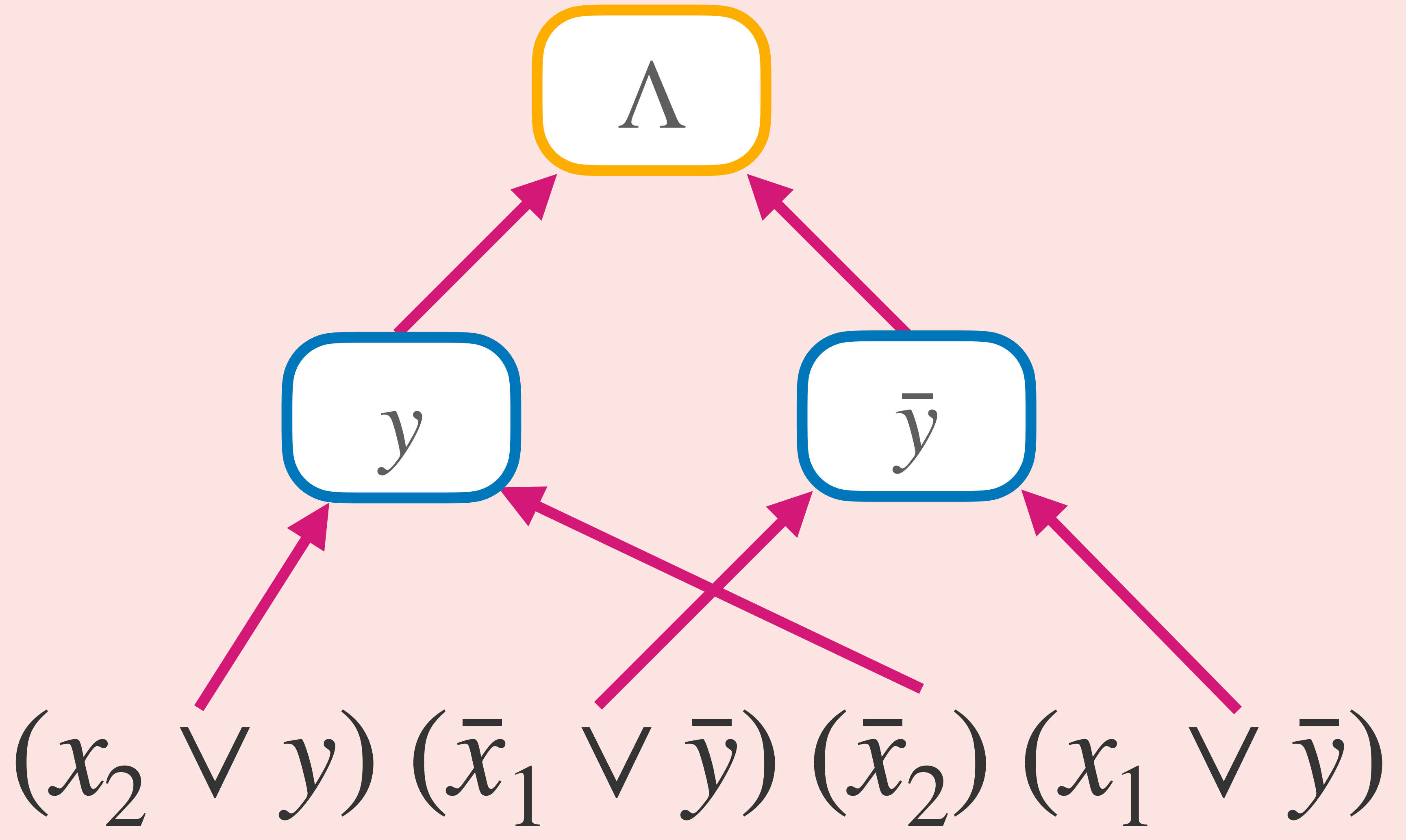


The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf

$x = [0,1], y = [0]$



Invariant: If C is the current clause $C(x, y) = 0$

If C was derived from C', C'' by resolving on

- x_i then Alice sends x_i to Bob
- y_i then Bob sends y_i to Alice

Move to the child C' or C'' containing the falsified literal

The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf

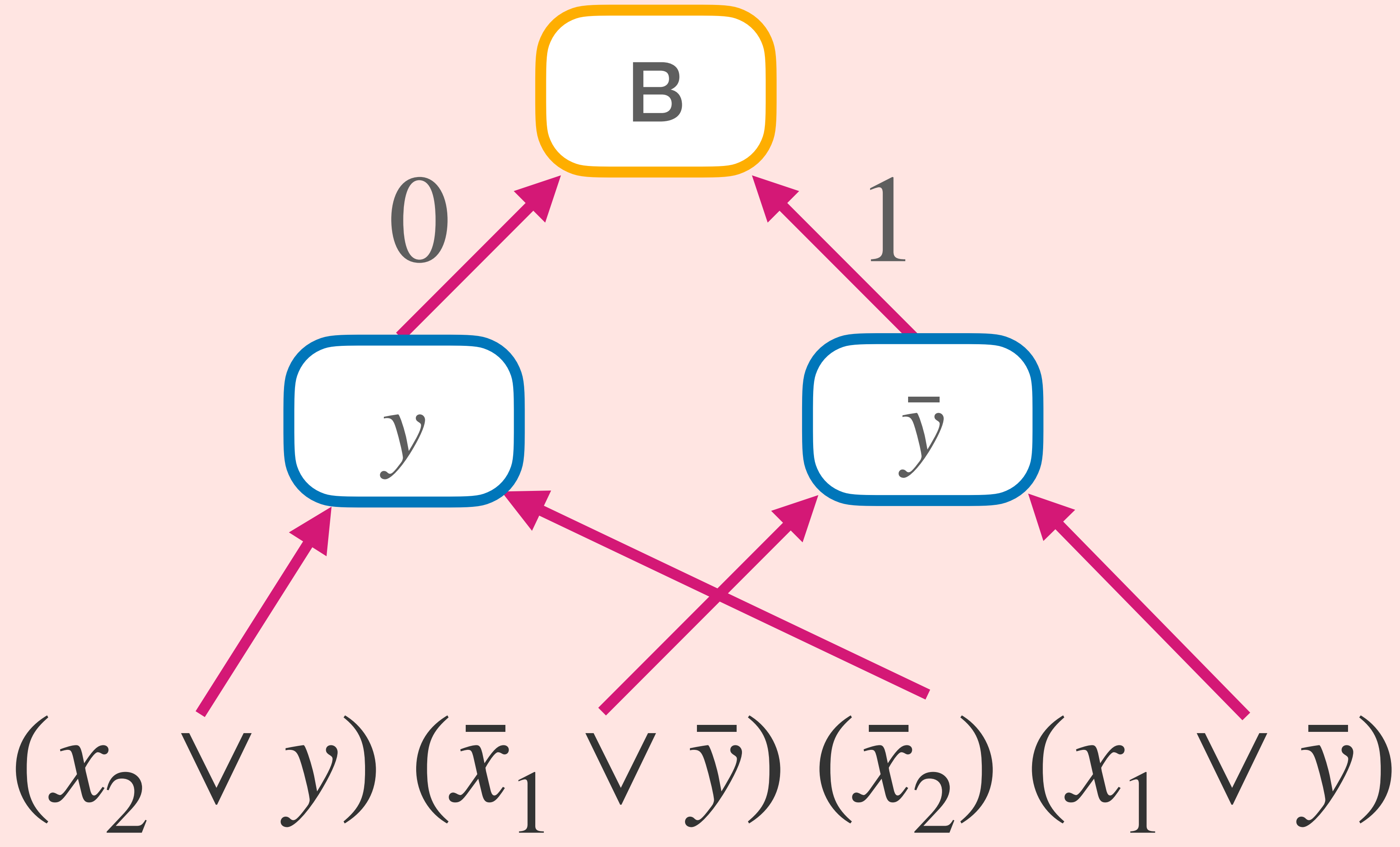
Invariant: If C is the current clause $C(x, y) = 0$

If C was derived from C', C'' by resolving on

- x_i then Alice sends x_i to Bob
- y_i then Bob sends y_i to Alice

Move to the child C' or C'' containing the falsified literal

$x = [0,1], y = [0]$



The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

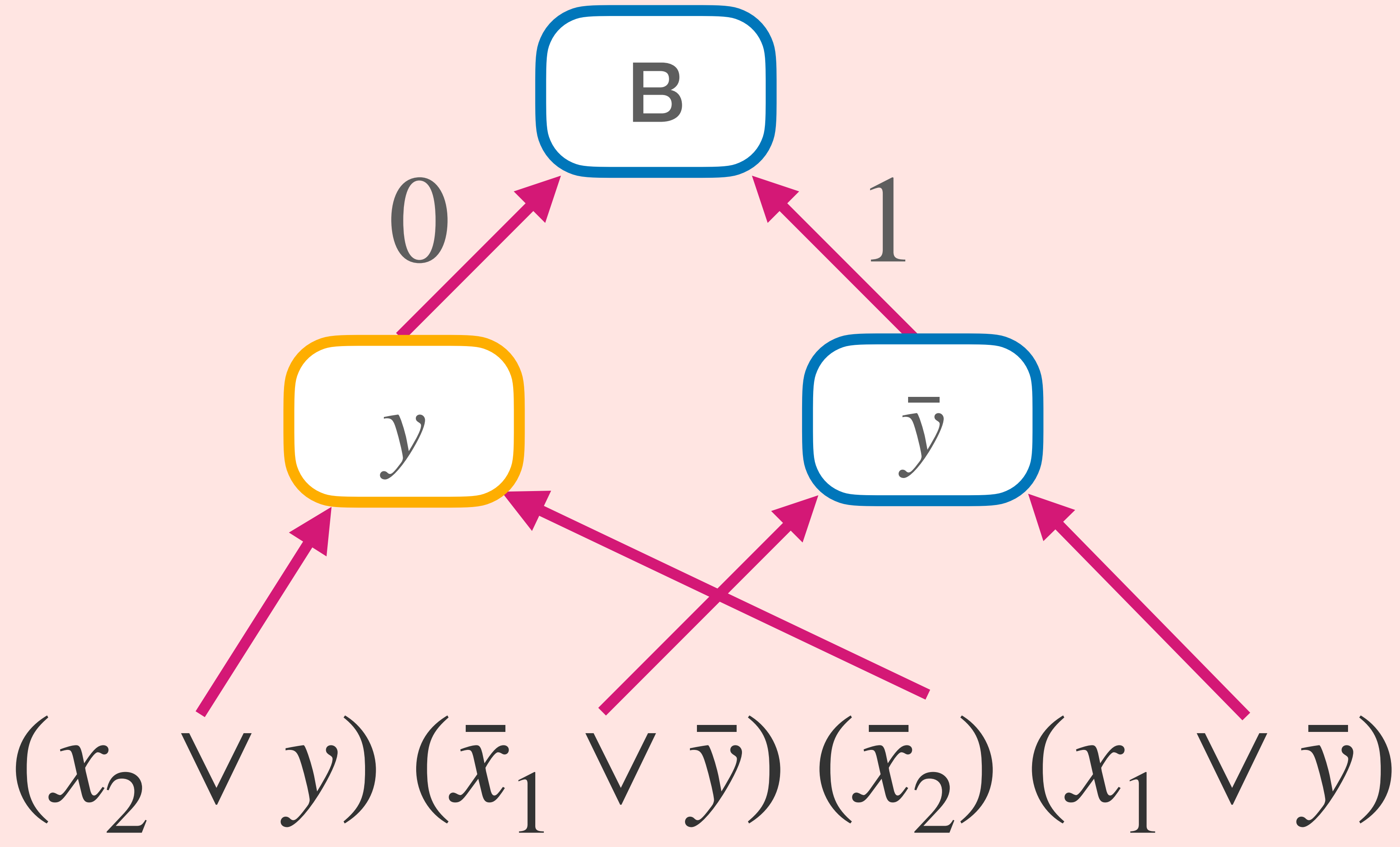
Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf

Invariant: If C is the current clause $C(x, y) = 0$

If C was derived from C', C'' by resolving on
- x_i then Alice sends x_i to Bob
- y_i then Bob sends y_i to Alice

Move to the child C' or C'' containing the falsified literal

$x = [0,1], y = [0]$



The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf

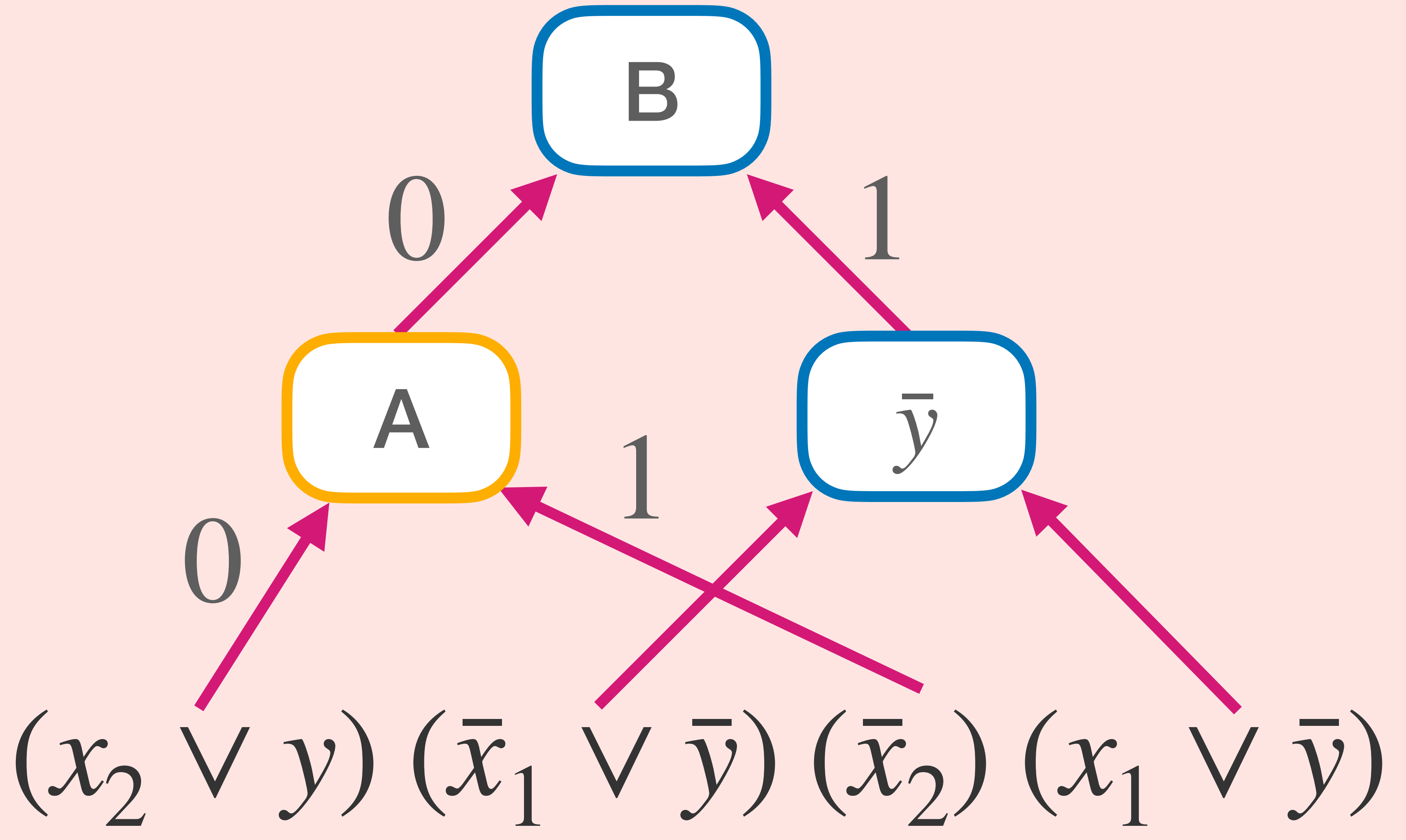
Invariant: If C is the current clause $C(x, y) = 0$

If C was derived from C', C'' by resolving on

- x_i then Alice sends x_i to Bob
- y_i then Bob sends y_i to Alice

Move to the child C' or C'' containing the falsified literal

$x = [0,1], y = [0]$



The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

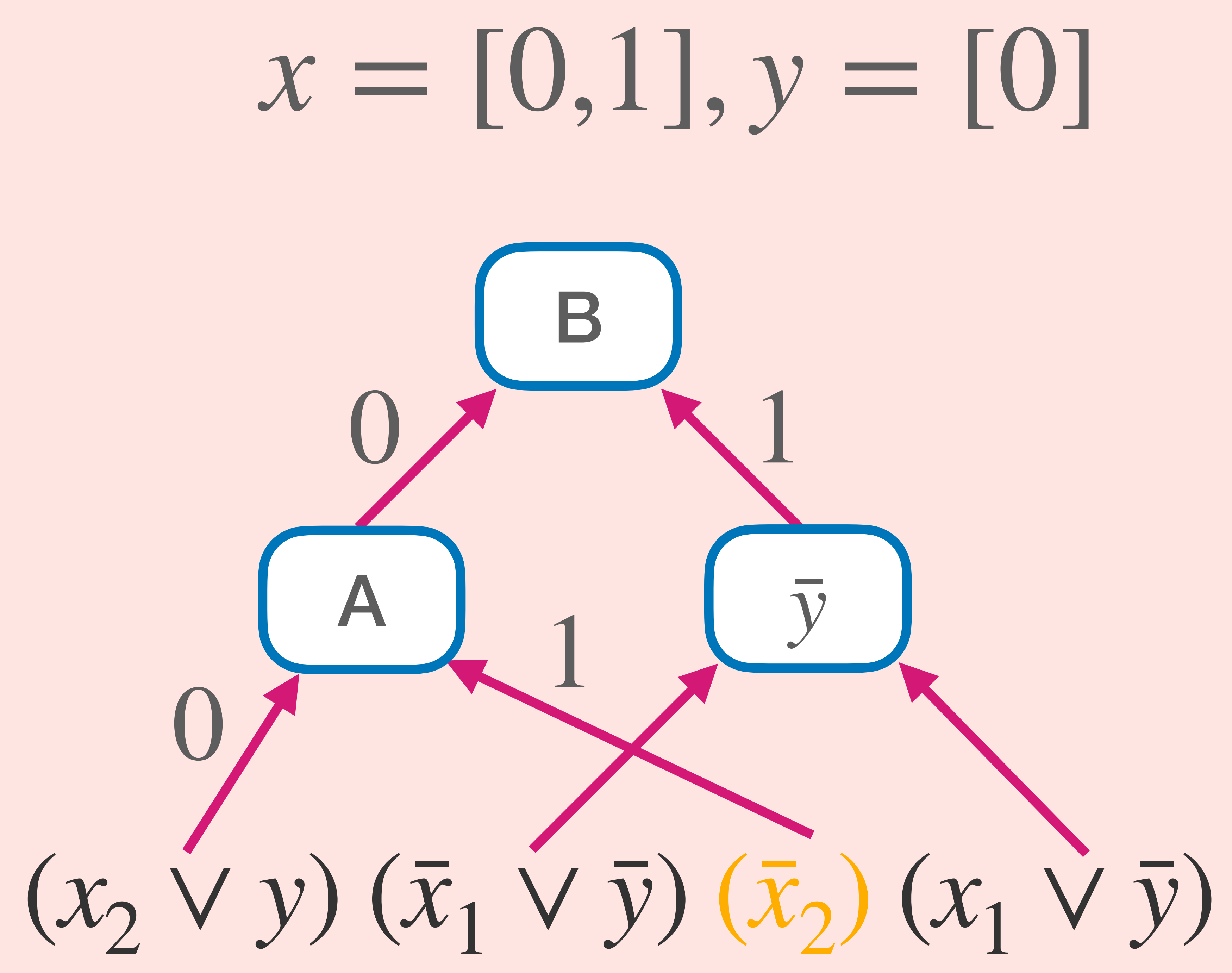
Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf

Invariant: If C is the current clause $C(x, y) = 0$

If C was derived from C', C'' by resolving on

- x_i then Alice sends x_i to Bob
- y_i then Bob sends y_i to Alice

Move to the child C' or C'' containing the falsified literal



The False Clause Search Problem

Thm:
A tree-like resolution proof implies a cc-protocol for $\text{Search}_F^{X,Y}$

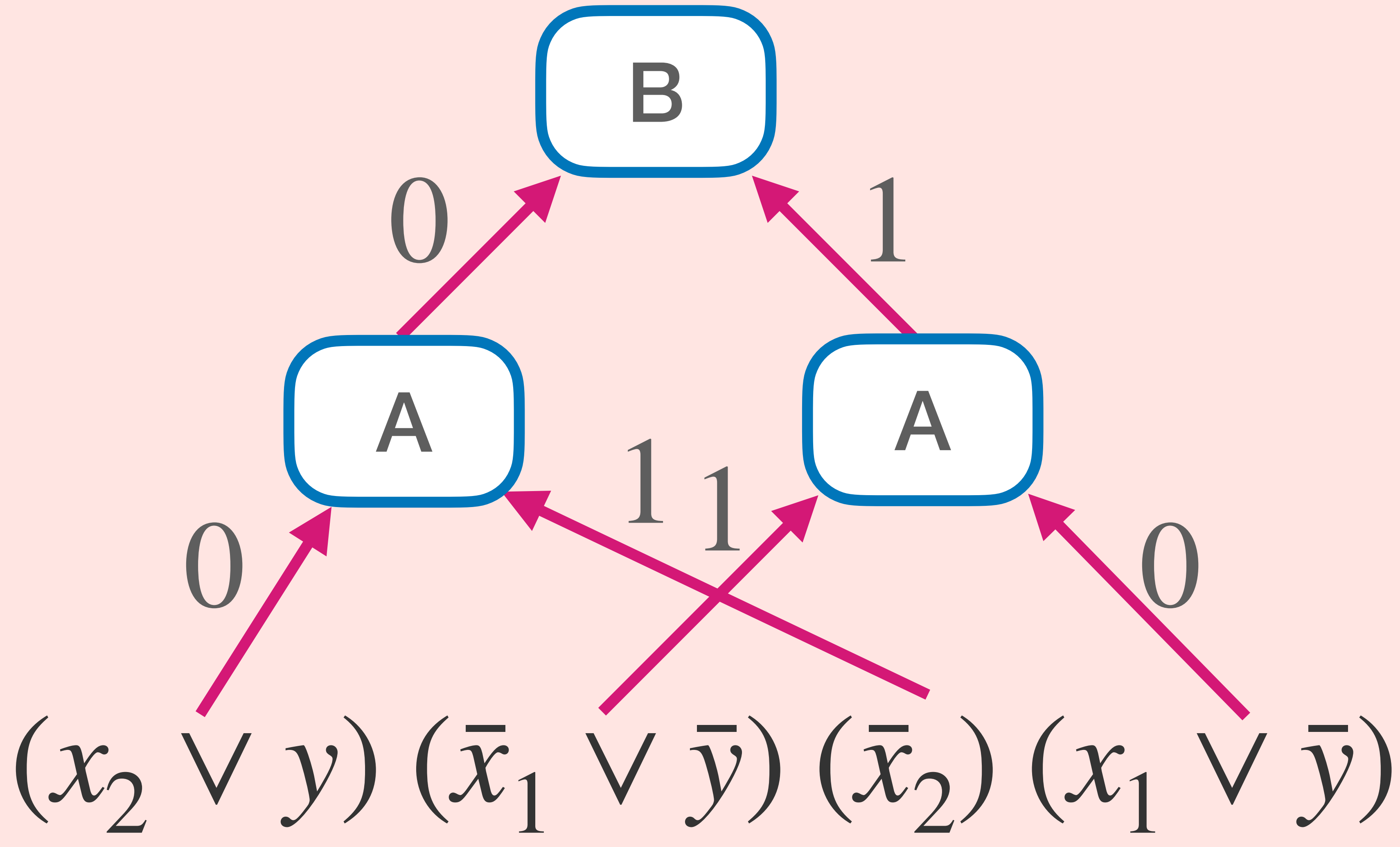
Proof: On input $(x, y) \in \{0,1\}^X \times \{0,1\}^Y$ walk from the root to a leaf

Invariant: If C is the current clause $C(x, y) = 0$

If C was derived from C', C'' by resolving on

- x_i then Alice sends x_i to Bob
- y_i then Bob sends y_i to Alice

Move to the child C' or C'' containing the falsified literal



Tale of Two Search Problems

So far we have seen

- Monotone circuit complexity $\approx \text{mKW}_f$
- Proof complexity $\approx \text{search}_F^{X,Y}$

Tale of Two Search Problems

So far we have seen

- Monotone circuit complexity $\approx \mathrm{mKW}_f$
- Proof complexity $\approx \mathrm{search}_F^{X,Y}$

We can prove a generic interpolation theorem by relating the two problems

Tale of Two Search Problems

So far we have seen

- Monotone circuit complexity $\approx$ mKW$_f$
- Proof complexity $\approx$ search$_F^{X,Y}$

We can prove a generic interpolation theorem by relating the two problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1)$, $R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1)$, $R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \mathsf{search}_F^{X,Y} \iff (A(x), R(y), i) \in \mathsf{mKW}_f$$

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \mathsf{search}_F^{X,Y} \iff (A(x), R(y), i) \in \mathsf{mKW}_f$$

Upshot: Solving $\mathsf{search}_F^{X,Y}$ is equivalent to solving mKW_f under on inputs $A(x), R(y)$ for $(x, y) \in \{0,1\}^{X \times Y}$

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Upshot: Solving $\text{search}_F^{X,Y}$ is equivalent to solving mKW_f under on inputs $A(x), R(y)$ for $(x, y) \in \{0,1\}^{X \times Y}$

tree-Res proof of $F \Longrightarrow$ cc-protocol for $\text{search}_F^{X,Y}$

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1)$, $R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Upshot: Solving $\text{search}_F^{X,Y}$ is equivalent to solving mKW_f under on inputs $A(x), R(y)$ for $(x, y) \in \{0,1\}^{X \times Y}$

tree-Res proof of $F \implies$ cc-protocol for $\text{search}_F^{X,Y}$

$\implies$ cc-protocol for mKW_f

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1)$, $R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Upshot: Solving $\text{search}_F^{X,Y}$ is equivalent to solving mKW_f under on inputs $A(x), R(y)$ for $(x, y) \in \{0,1\}^{X \times Y}$

tree-Res proof of F $\Longrightarrow$ cc-protocol for $\text{search}_F^{X,Y}$

$\Longrightarrow$ cc-protocol for mKW_f

$\Longrightarrow$ monotone circuit for f

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \land \ldots \land C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1)$, $R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1)$, $R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

$\to$ $C_i(x, y) = 0$ and $C_i(x', y') = 0$ then $C_i(x, y') = C_i(x', y) = 0$

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

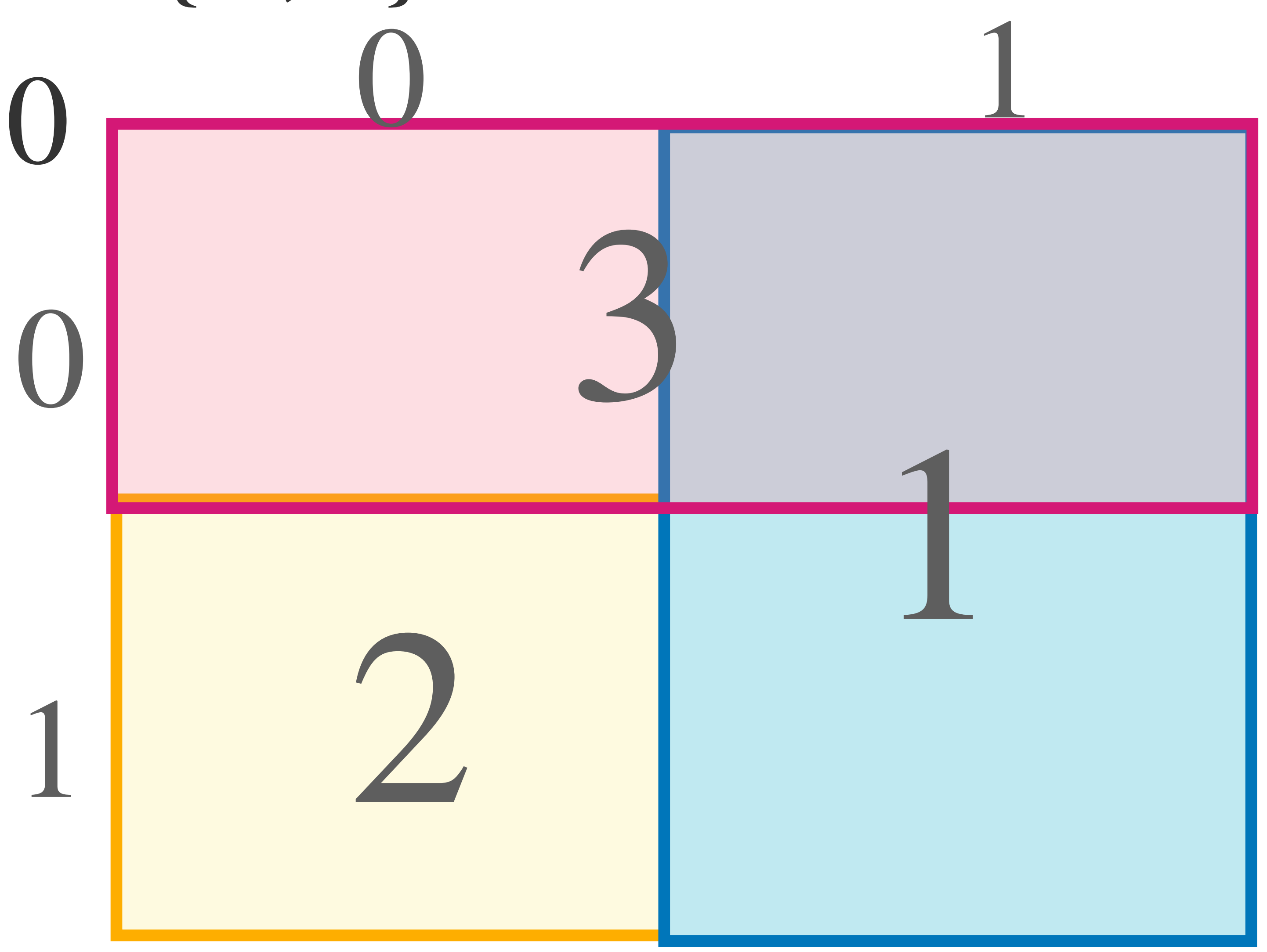
$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

$\rightarrow C_i(x, y) = 0$ and $C_i(x', y') = 0$ then $C_i(x, y') = C_i(x', y) = 0$

e.g. $\bar{y} \wedge (y \vee \bar{x}) \wedge x$



Tale of Two Search Problems

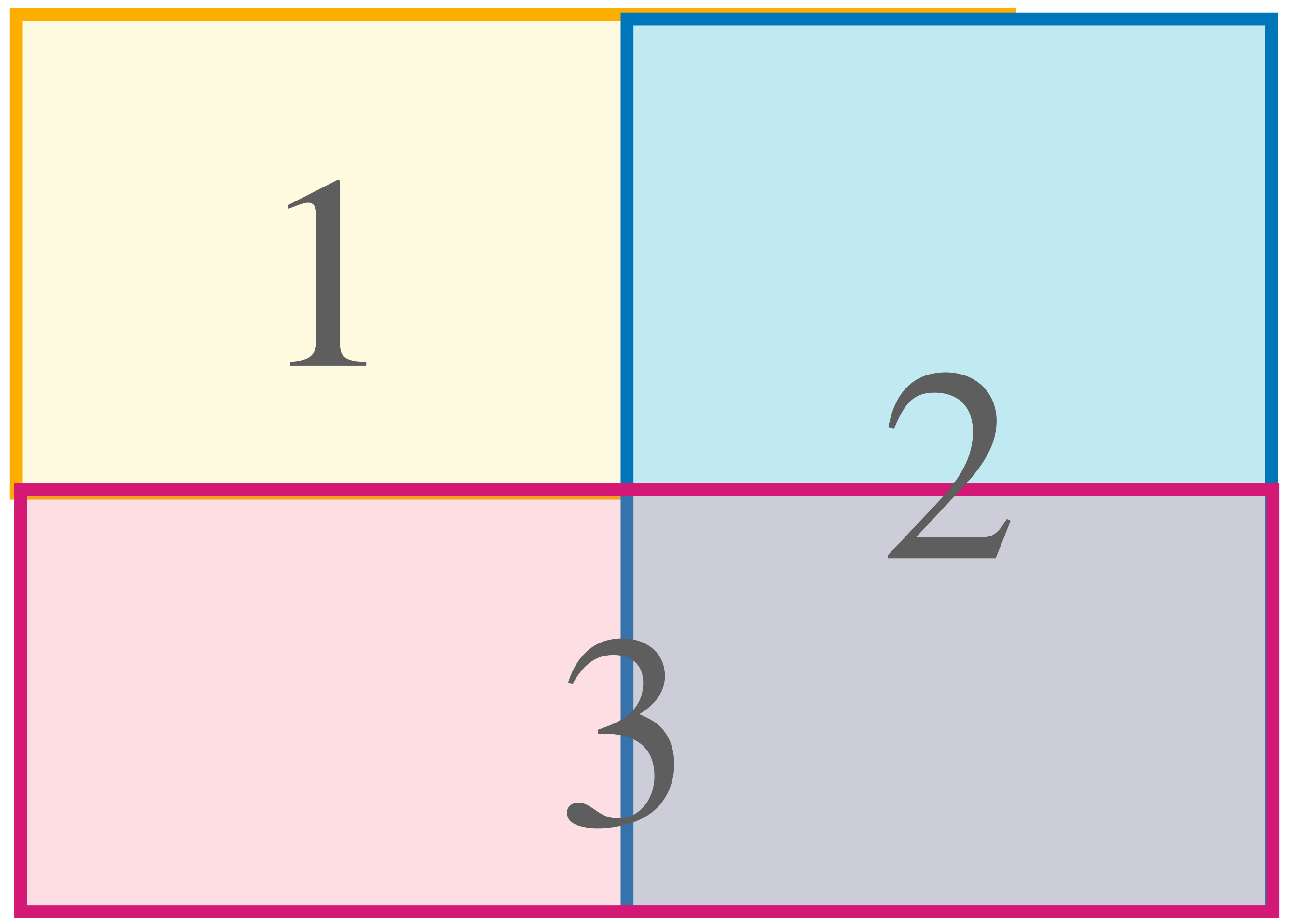
Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1)$, $R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff x intersects rectangle i



Tale of Two Search Problems

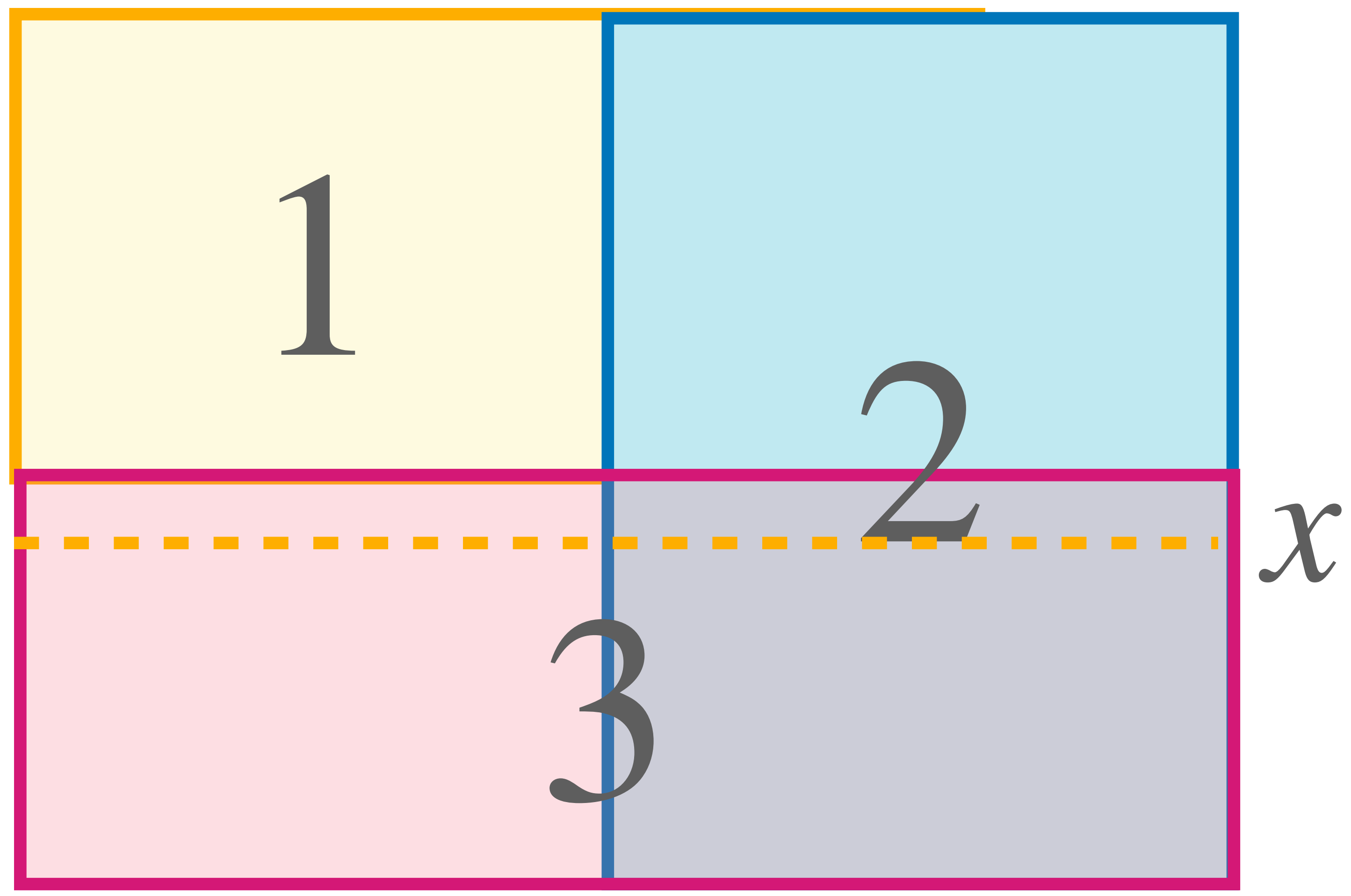
Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1)$, $R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff x intersects rectangle i



Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

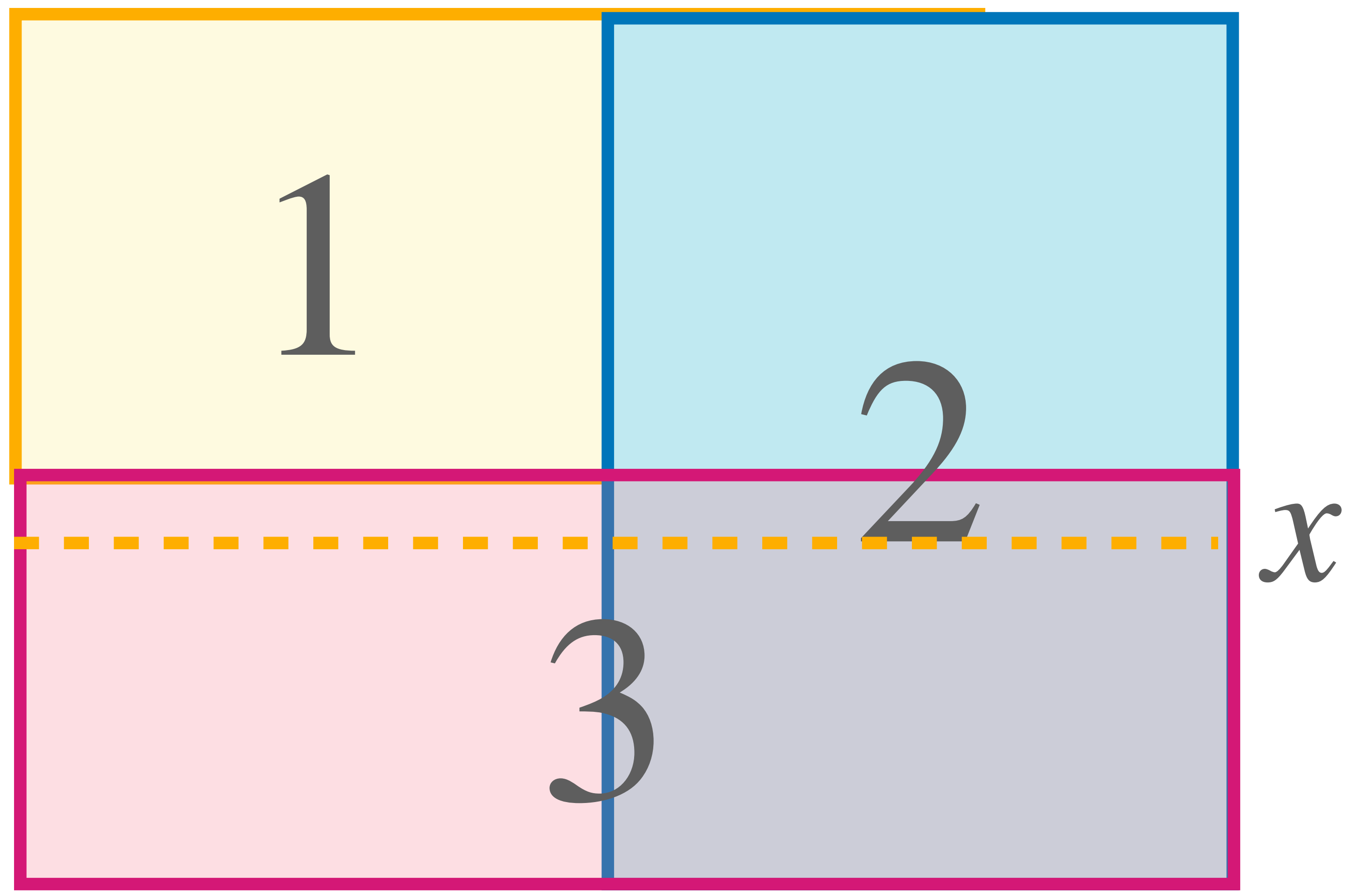
$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

$A(x) = [0,1,1]$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff x intersects rectangle i



Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

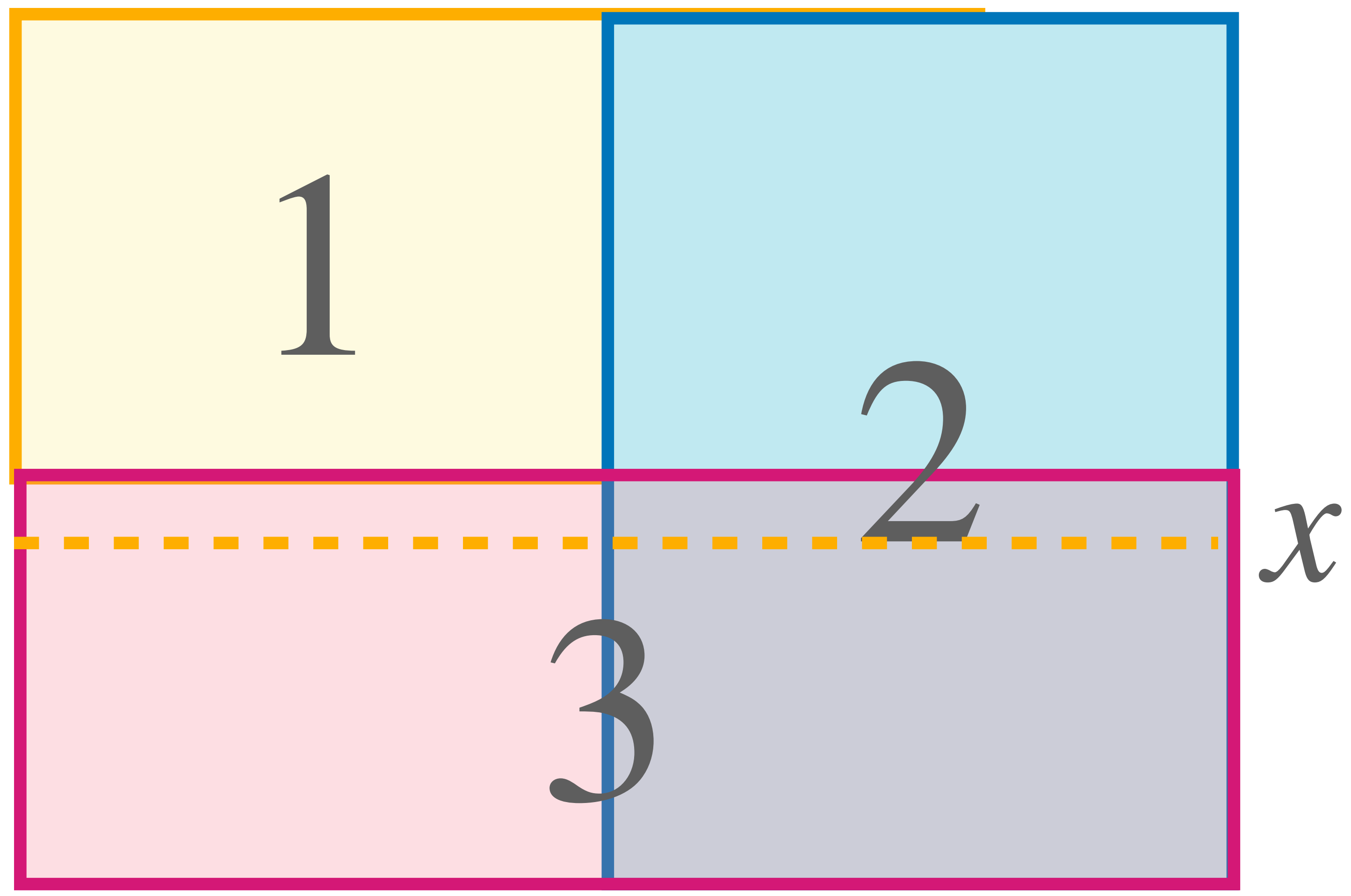
$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

$A(x) = [0,1,1]$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff x intersects rectangle i
  i.e. iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$



Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

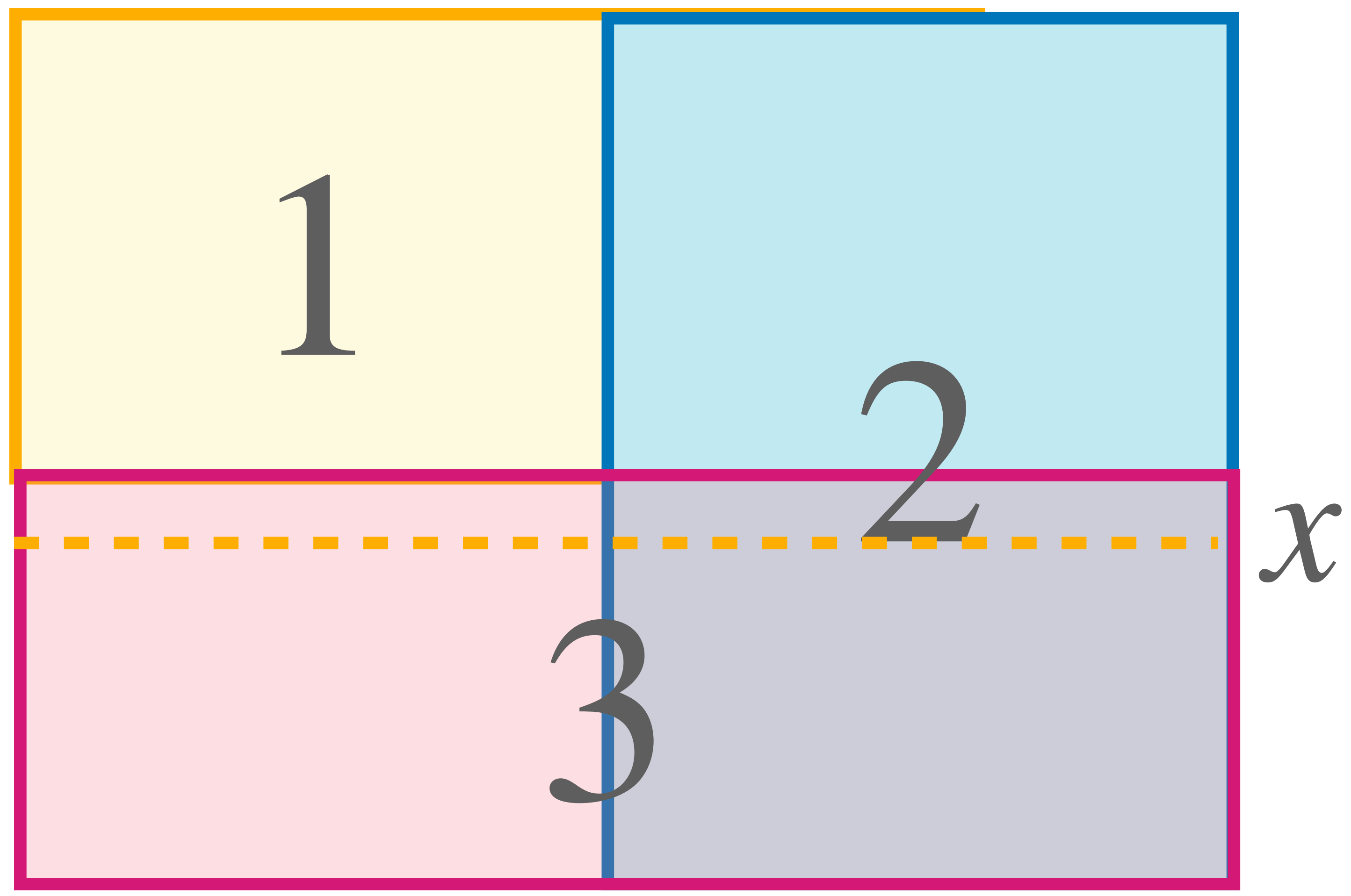
$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

$A(x) = [0,1,1]$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff x intersects rectangle i
  i.e. iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$
- $R(y) \in \{0,1\}^m$: set $R(y)_i = 0$ iff y intersects rectangle i
  i.e. iff exists $x \in \{0,1\}^X$ s.t. $C_i(x, y) = 0$



Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

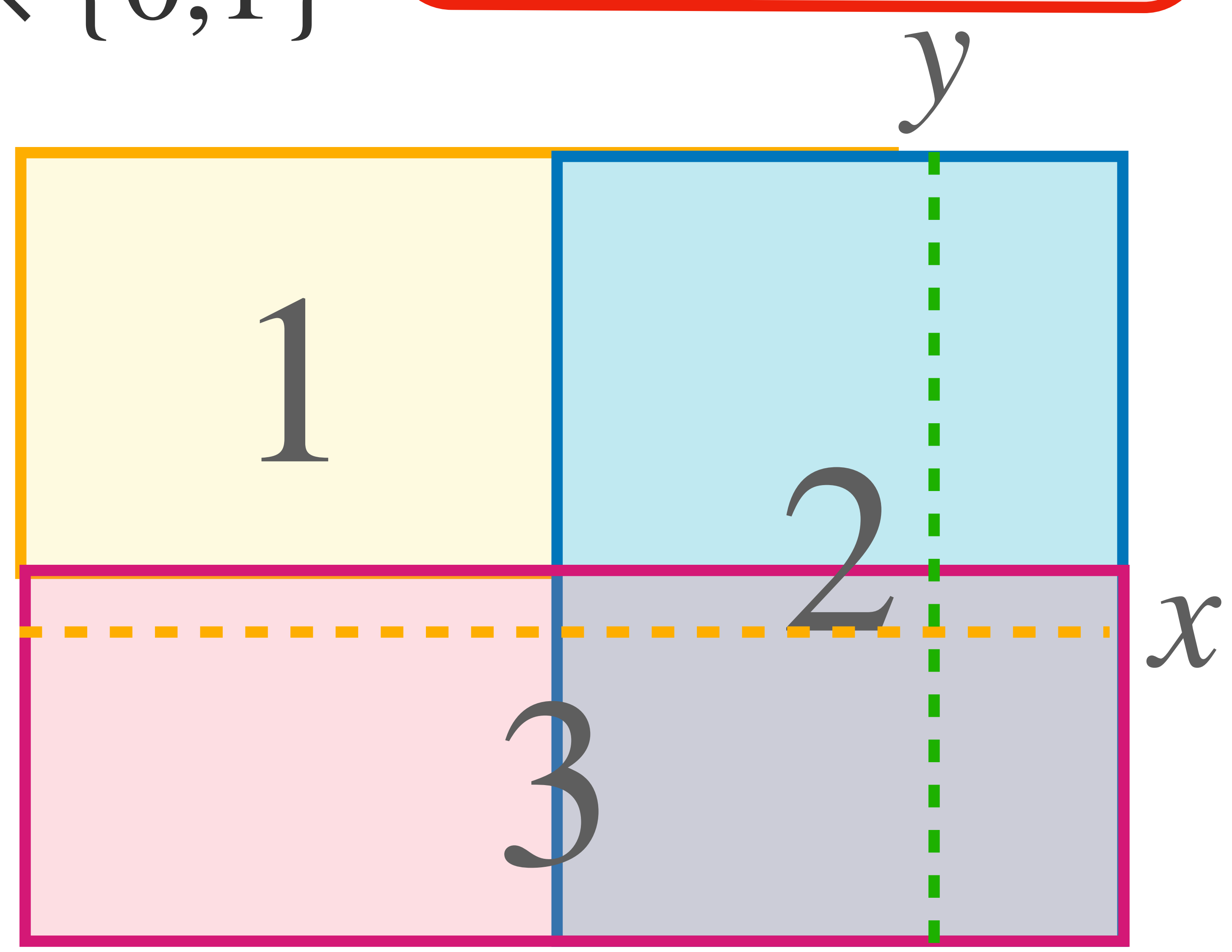
$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

$A(x) = [0,1,1]$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff x intersects rectangle i
  i.e. iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$
- $R(y) \in \{0,1\}^m$: set $R(y)_i = 0$ iff y intersects rectangle i
  i.e. iff exists $x \in \{0,1\}^X$ s.t. $C_i(x, y) = 0$



Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1)$, $R : \{0,1\}^Y \to f^{-1}(0)$ such that

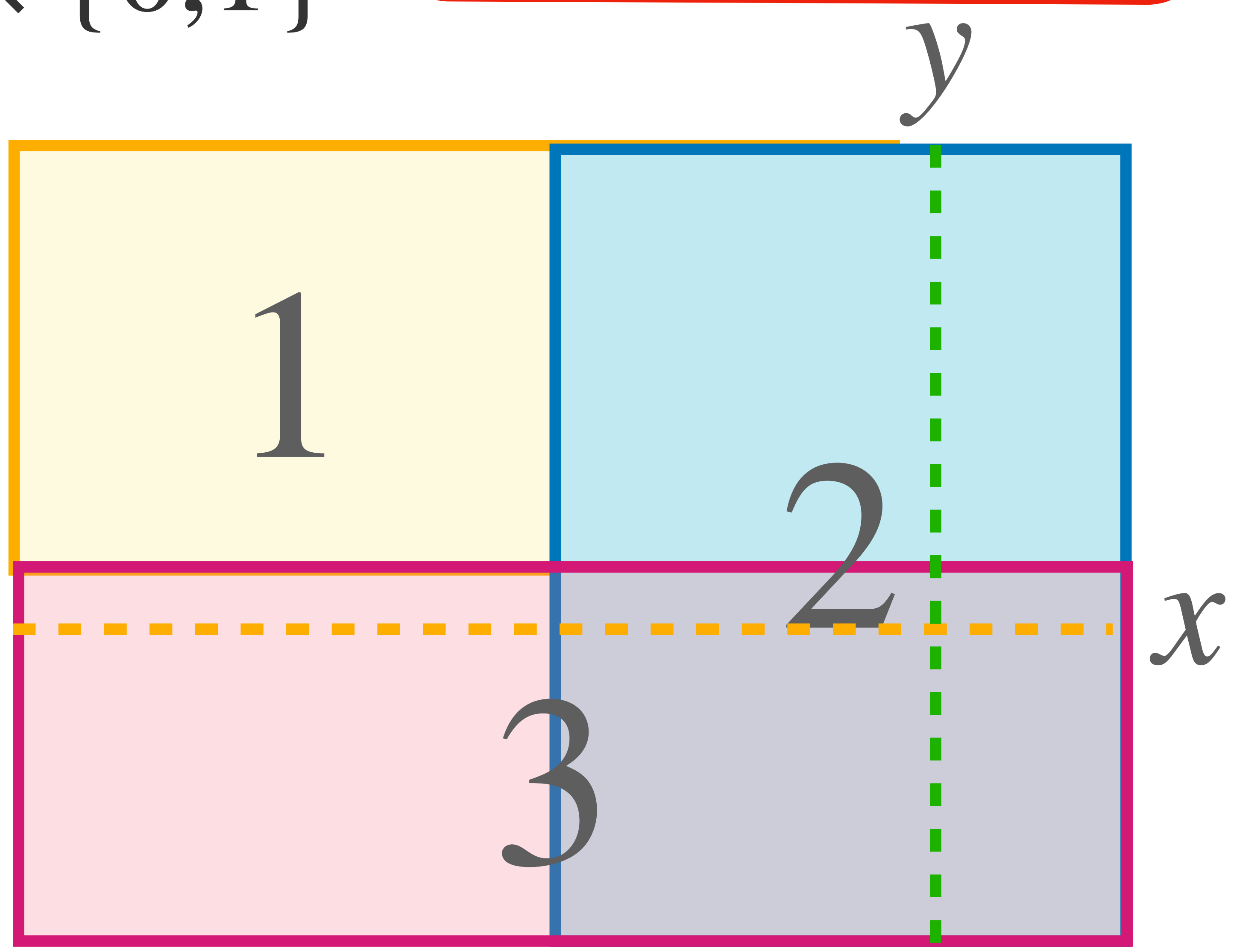
$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

$A(x) = [0,1,1]$
$R(y) = [1,0,0]$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff x intersects rectangle i
  i.e. iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$
- $R(y) \in \{0,1\}^m$: set $R(y)_i = 0$ iff y intersects rectangle i
  i.e. iff exists $x \in \{0,1\}^X$ s.t. $C_i(x, y) = 0$



Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

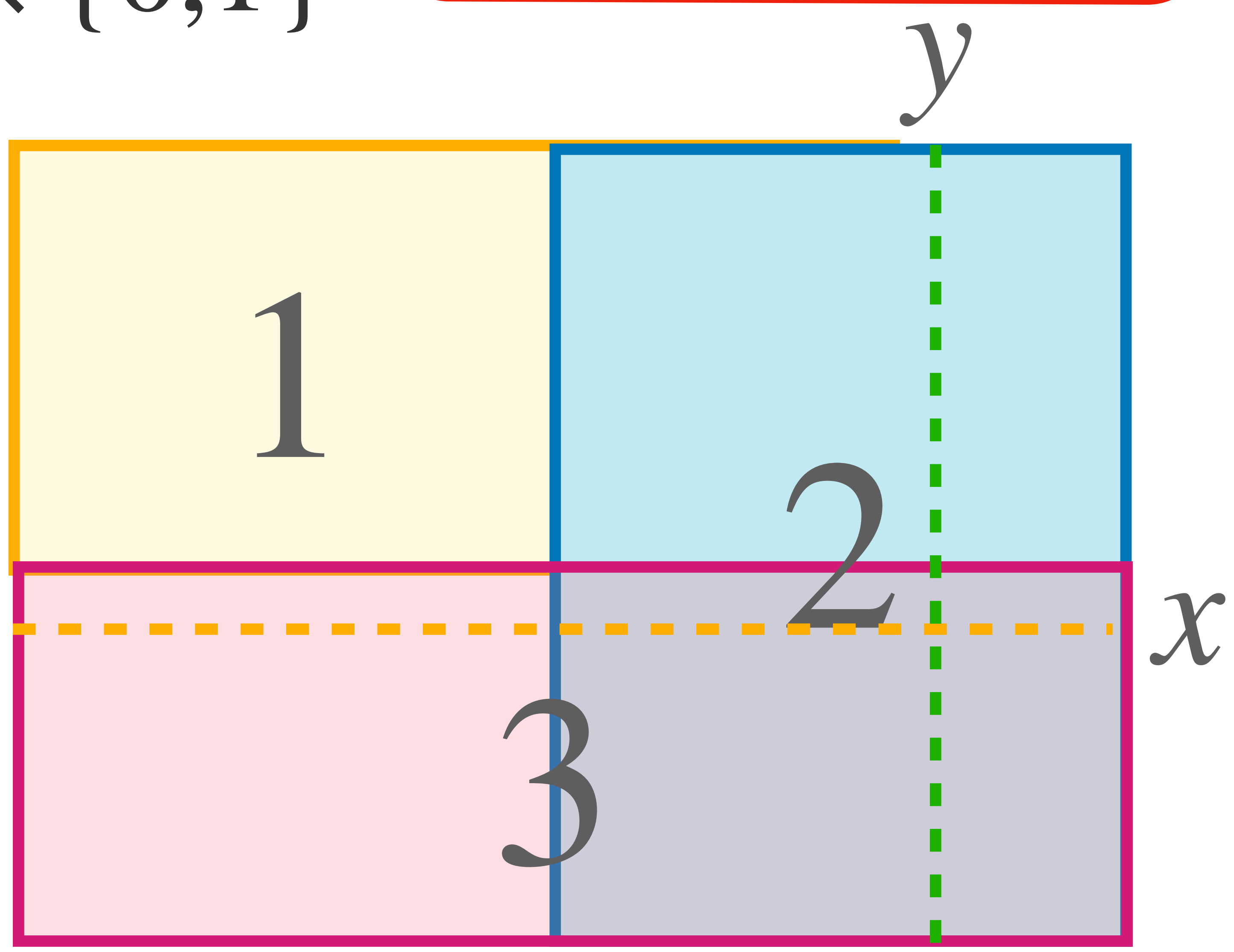
$A(x) = [0,1,1]$
$R(y) = [1,0,0]$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff x intersects rectangle i
  i.e. iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$
- $R(y) \in \{0,1\}^m$: set $R(y)_i = 0$ iff y intersects rectangle i
  i.e. iff exists $x \in \{0,1\}^X$ s.t. $C_i(x, y) = 0$

$$(x, y, i) \in \text{search}_F^{X,Y} \iff C_i(x, y) = 0$$



Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

$A(x) = [0,1,1]$

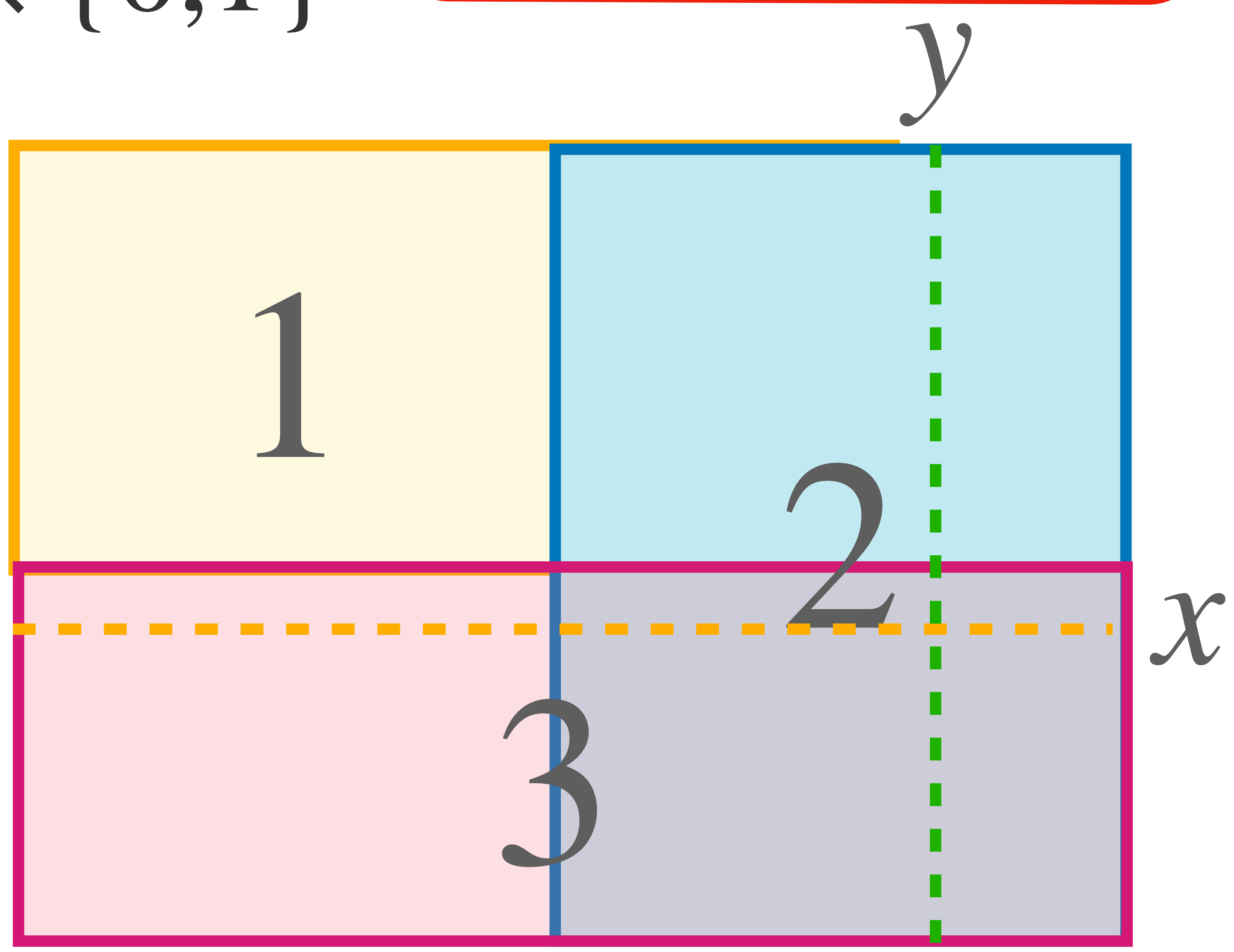
$R(y) = [1,0,0]$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff x intersects rectangle i
  i.e. iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$
- $R(y) \in \{0,1\}^m$: set $R(y)_i = 0$ iff y intersects rectangle i
  i.e. iff exists $x \in \{0,1\}^X$ s.t. $C_i(x, y) = 0$

$$(x, y, i) \in \text{search}_F^{X,Y} \iff C_i(x, y) = 0 \iff (x, y) \in R_i$$



Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

$A(x) = [0,1,1]$

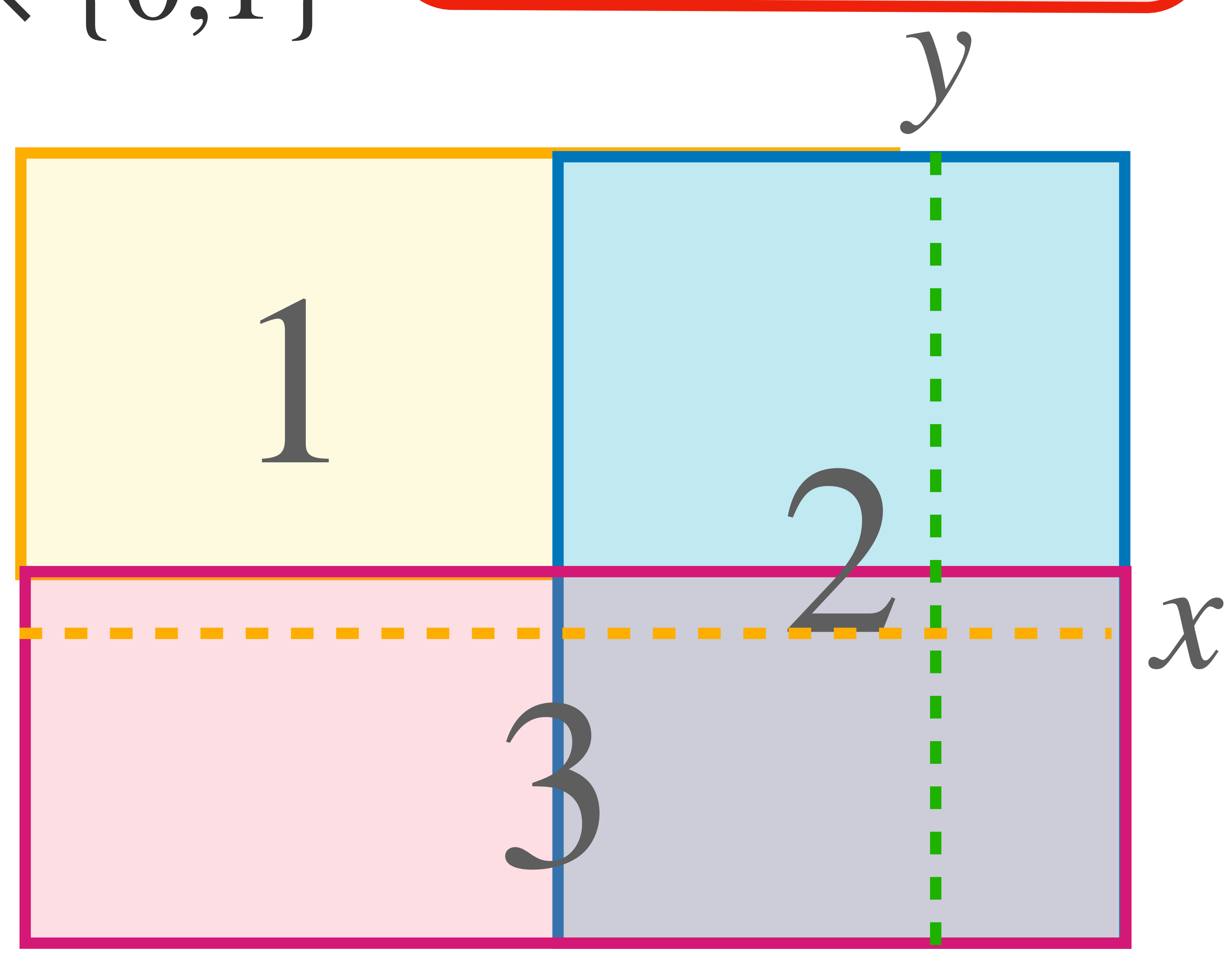
$R(y) = [1,0,0]$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The solutions of $\text{Search}_F^{X,Y}$ define a rectangle covering of $\{0,1\}^x \times \{0,1\}^Y$

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff x intersects rectangle i
  i.e. iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$
- $R(y) \in \{0,1\}^m$: set $R(y)_i = 0$ iff y intersects rectangle i
  i.e. iff exists $x \in \{0,1\}^X$ s.t. $C_i(x, y) = 0$

$$(x, y, i) \in \text{search}_F^{X,Y} \iff C_i(x, y) = 0 \iff (x, y) \in R_i$$
$$\iff A_i(x) > R_i(y)$$



Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Proof: Specify $A(x)$ of accepting inputs and $R(y)$ of rejecting inputs

The function: Let f be a monotone function which accepts $A(x)$ for all $x \in \{0,1\}^X$ and rejects $R(y)$ for all $y \in \{0,1\}^Y$

Tale of Two Search Problems

Q. What is this function explicitly?

Tale of Two Search Problems

Q. What is this function explicitly?

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$

Tale of Two Search Problems

Q. What is this function explicitly?

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$
- $R(y) \in \{0,1\}^m$: set $R(y)_i = 0$ iff exists $x \in \{0,1\}^X$ s.t. $C_i(x, y) = 0$

Tale of Two Search Problems

Q. What is this function explicitly?

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$
- $R(y) \in \{0,1\}^m$: set $R(y)_i = 0$ iff exists $x \in \{0,1\}^X$ s.t. $C_i(x, y) = 0$

Let C_i^X be obtained by discarding the Y-variables from C_i; C_i^Y analogous

Tale of Two Search Problems

Q. What is this function explicitly?

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$
- $R(y) \in \{0,1\}^m$: set $R(y)_i = 0$ iff exists $x \in \{0,1\}^X$ s.t. $C_i(x, y) = 0$

Let C_i^X be obtained by discarding the Y-variables from C_i; C_i^Y analogous

[HP17] **unsat certificate**: $\text{cert}_F^{X,Y}(\alpha) = \begin{cases} 0 & \text{if } C_i^X : \alpha_i = 1 \text{ is satisfiable} \\ 1 & \text{if } C_i^Y : \alpha_i = 0 \text{ is satisfiable} \end{cases}$

Tale of Two Search Problems

Q. What is this function explicitly?

- $A(x) \in \{0,1\}^m$: set $A(x)_i = 1$ iff exists $y \in \{0,1\}^Y$ s.t. $C_i(x, y) = 0$
- $R(y) \in \{0,1\}^m$: set $R(y)_i = 0$ iff exists $x \in \{0,1\}^X$ s.t. $C_i(x, y) = 0$

Let C_i^X be obtained by discarding the Y-variables from C_i; C_i^Y analogous

[HP17] **unsat certificate**: $\text{cert}_F^{X,Y}(\alpha) = \begin{cases} 0 & \text{if } C_i^X : \alpha_i = 1 \text{ is satisfiable} \\ 1 & \text{if } C_i^Y : \alpha_i = 0 \text{ is satisfiable} \end{cases}$

[FPPR17] proved same theorem using a function containing cert as a projection

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \mathsf{search}_F^{X,Y} \iff (A(x), R(y), i) \in \mathsf{mKW}_f$$

Changing the underlying model of computation from cc-protocols to other models changes the different interpolation theorems we get!

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \mathsf{search}_F^{X,Y} \iff (A(x), R(y), i) \in \mathsf{mKW}_f$$

Changing the underlying model of computation from cc-protocols to other models changes the different interpolation theorems we get!

- Communication dags/PLS^{cc} protocols = Resolution, monotone circuits

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \mathsf{search}_F^{X,Y} \iff (A(x), R(y), i) \in \mathsf{mKW}_f$$

Changing the underlying model of computation from cc-protocols to other models changes the different interpolation theorems we get!

- Communication dags/PLS^{cc} protocols = Resolution, monotone circuits
- Real communication dags = Cutting Planes, monotone real circuits

Tale of Two Search Problems

Thm: For any unsatisfiable CNF formula $F = C_1 \wedge \ldots \wedge C_m$ and any partition $X \times Y$ of $[n]$ there is a monotone function $f : \{0,1\}^m \to \{0,1\}$ and mappings $A : \{0,1\}^X \to f^{-1}(1),\ R : \{0,1\}^Y \to f^{-1}(0)$ such that

$$(x, y, i) \in \text{search}_F^{X,Y} \iff (A(x), R(y), i) \in \text{mKW}_f$$

Changing the underlying model of computation from cc-protocols to other models changes the different interpolation theorems we get!

- Communication dags/PLS^{cc} protocols = Resolution, monotone circuits
- Real communication dags = Cutting Planes, monotone real circuits
- Communication in expectation = Sherali-Adams, extended formulations

Tale of Two Search Problems

Q. How does cert relate to the interpolant function for split formulas?

Tale of Two Search Problems

Q . How does cert relate to the interpolant function for split formulas?

Unsat Certificate:

$$\text{cert}_F^{X,Y}(\alpha) = \begin{cases} 0 & \text{if } C_i^X : \alpha_i = 1 \text{ is satisfiable} \\ 1 & \text{if } C_i^Y : \alpha_i = 0 \text{ is satisfiable} \end{cases}$$

Tale of Two Search Problems

Q. How does cert relate to the interpolant function for split formulas?

Unsat Certificate: $\text{cert}_F^{X,Y}(\alpha) = \begin{cases} 0 & \text{if } C_i^X : \alpha_i = 1 \text{ is satisfiable} \\ 1 & \text{if } C_i^Y : \alpha_i = 0 \text{ is satisfiable} \end{cases}$

Split formula: $F(x, y, z) = A(x, z) \wedge B(y, z)$

Where A, B are CNF and all z variables occur positively in A

Monotone Interpolant function: $I_F(\alpha) = \begin{cases} 0 & \text{if } A(x, \alpha) \text{ is satisfiable} \\ 1 & \text{if } B(\alpha, y) \text{ is satisfiable} \end{cases}$

Tale of Two Search Problems

Q. How does cert relate to the interpolant function for split formulas?

Unsat Certificate: $\text{cert}_F^{X,Y}(\alpha) = \begin{cases} 0 & \text{if } C_i^X : \alpha_i = 1 \text{ is satisfiable} \\ 1 & \text{if } C_i^Y : \alpha_i = 0 \text{ is satisfiable} \end{cases}$

Split formula: $F(x, y, z) = A(x, z) \wedge B(y, z)$

Where A, B are CNF and all z variables occur positively in A

Monotone Interpolant function: $I_F(\alpha) = \begin{cases} 0 & \text{if } A(x, \alpha) \text{ is satisfiable} \\ 1 & \text{if } B(\alpha, y) \text{ is satisfiable} \end{cases}$

Any formula $F = C_1 \wedge \ldots \wedge C_m$ can be made split!

Tale of Two Search Problems

Q. How does cert relate to the interpolant function for split formulas?

Unsat Certificate: $\text{cert}_F^{X,Y}(\alpha) = \begin{cases} 0 & \text{if } C_i^X : \alpha_i = 1 \text{ is satisfiable} \\ 1 & \text{if } C_i^Y : \alpha_i = 0 \text{ is satisfiable} \end{cases}$

Split formula: $F(x, y, z) = A(x, z) \wedge B(y, z)$

Where A, B are CNF and all z variables occur positively in A

Monotone Interpolant function: $I_F(\alpha) = \begin{cases} 0 & \text{if } A(x, \alpha) \text{ is satisfiable} \\ 1 & \text{if } B(\alpha, y) \text{ is satisfiable} \end{cases}$

Any formula $F = C_1 \wedge \ldots \wedge C_m$ can be made split!

Introduce new variables $z_1, \ldots, z_m$.

Tale of Two Search Problems

Q. How does cert relate to the interpolant function for split formulas?

Unsat Certificate: $\text{cert}_F^{X,Y}(\alpha) = \begin{cases} 0 & \text{if } C_i^X : \alpha_i = 1 \text{ is satisfiable} \\ 1 & \text{if } C_i^Y : \alpha_i = 0 \text{ is satisfiable} \end{cases}$

Split formula: $F(x, y, z) = A(x, z) \wedge B(y, z)$

Where A, B are CNF and all z variables occur positively in A

Monotone Interpolant function: $I_F(\alpha) = \begin{cases} 0 & \text{if } A(x, \alpha) \text{ is satisfiable} \\ 1 & \text{if } B(\alpha, y) \text{ is satisfiable} \end{cases}$

Any formula $F = C_1 \wedge \ldots \wedge C_m$ can be made split!

Introduce new variables $z_1, \ldots, z_m$. Define

$$F_{\text{split}} = (C_1^Y \vee z_1) \wedge \ldots \wedge (C_m^Y \vee z_m) \wedge (C_1^X \vee \bar{z}_1) \wedge \ldots \wedge (C_m^X \vee \bar{z}_m)$$

Tale of Two Search Problems

Q. How does cert relate to the interpolant function for split formulas?

Unsat Certificate: $\text{cert}_F^{X,Y}(\alpha) = \begin{cases} 0 & \text{if } C_i^X : \alpha_i = 1 \text{ is satisfiable} \\ 1 & \text{if } C_i^Y : \alpha_i = 0 \text{ is satisfiable} \end{cases}$

Split formula: $F(x, y, z) = A(x, z) \wedge B(y, z)$

Where A, B are CNF and all z variables occur positively in A

Monotone Interpolant function: $I_F(\alpha) = \begin{cases} 0 & \text{if } A(x, \alpha) \text{ is satisfiable} \\ 1 & \text{if } B(\alpha, y) \text{ is satisfiable} \end{cases}$

Any formula $F = C_1 \wedge \ldots \wedge C_m$ can be made split!

Introduce new variables $z_1, \ldots, z_m$. Define

$$F_{\text{split}} = (C_1^Y \vee z_1) \wedge \ldots \wedge (C_m^Y \vee z_m) \wedge (C_1^X \vee \bar{z}_1) \wedge \ldots \wedge (C_m^X \vee \bar{z}_m)$$

Then $I_{F_{\text{split}}}$ is $\text{cert}_F^{X,Y}$!

Tale of Two Search Problems

We saw that for every F there is f_F such that $\mathsf{search}_F = \mathsf{mKW}_f$

Tale of Two Search Problems

We saw that for every F there is f_F such that $\mathsf{search}_F = \mathsf{mKW}_f$

And cc-protocols for $\mathsf{mKW}(f_F)$ are equivalent to monotone formulas for f_F

Tale of Two Search Problems

We saw that for every F there is f_F such that $\mathsf{search}_F = \mathsf{mKW}_f$

And cc-protocols for $\mathsf{mKW}(f_F)$ are equivalent to monotone formulas for f_F

Q. Is there a proof system that is equivalent (on f_F) to monotone formulas?

Tale of Two Search Problems

We saw that for every F there is f_F such that $\mathsf{search}_F = \mathsf{mKW}_f$

And cc-protocols for $\mathsf{mKW}(f_F)$ are equivalent to monotone formulas for f_F

Q . Is there a proof system that is equivalent (on f_F) to monotone formulas? Yes!

Tale of Two Search Problems

We saw that for every F there is f_F such that $\mathsf{search}_F = \mathsf{mKW}_f$

And cc-protocols for $\mathsf{mKW}(f_F)$ are equivalent to monotone formulas for f_F

Q. Is there a proof system that is equivalent (on f_F) to monotone formulas? Yes!

CC_1-proofs

Let $X \times Y$ be a partition of $[n]$

Tale of Two Search Problems

We saw that for every F there is f_F such that $\text{search}_F = \text{mKW}_f$

And cc-protocols for $\text{mKW}(f_F)$ are equivalent to monotone formulas for f_F

Q. Is there a proof system that is equivalent (on f_F) to monotone formulas? Yes!

CC_1-proofs

Let $X \times Y$ be a partition of $[n]$

Lines: any function $L : \{0,1\}^{X \times Y} \to \{0,1\}$ such that $f(x, y)$ can be evaluated with a single bit of communication

Tale of Two Search Problems

We saw that for every F there is f_F such that $\text{search}_F = \text{mKW}_f$

And cc-protocols for $\text{mKW}(f_F)$ are equivalent to monotone formulas for f_F

Q. Is there a proof system that is equivalent (on f_F) to monotone formulas? Yes!

CC_1-proofs

Let $X \times Y$ be a partition of $[n]$

Lines: any function $L : \{0,1\}^{X \times Y} \to \{0,1\}$ such that $f(x, y)$ can be evaluated with a single bit of communication

Semantic Inference rule: From L_1, L_2 derive any L satisfying

Soundness: $L_1(x, y) \wedge L_2(x, y) \implies L(x, y)$ for every $(x, y) \in \{0,1\}^{X \times Y}$

Tale of Two Search Problems

We saw that for every F there is f_F such that $\text{search}_F = \text{mKW}_f$

And cc-protocols for $\text{mKW}(f_F)$ are equivalent to monotone formulas for f_F

Q. Is there a proof system that is equivalent (on f_F) to monotone formulas? Yes!

CC_1-proofs

Let $X \times Y$ be a partition of $[n]$

Lines: any function $L : \{0,1\}^{X \times Y} \to \{0,1\}$ such that $f(x, y)$ can be evaluated with a single bit of communication

Semantic Inference rule: From L_1, L_2 derive any L satisfying

Soundness: $L_1(x, y) \wedge L_2(x, y) \implies L(x, y)$ for every $(x, y) \in \{0,1\}^{X \times Y}$

CC_1 proofs are equivalent to cc-protocols computing $\text{search}_F^{X,Y}$