Game semantics for multi-player logics of imperfect information

Pietro Galliani
pgallian@gmail.com

LINT ILLC
Universiteit van Amsterdam

Game semantics for multi-player logics of imperfect information

Logic...

Game semantics for multi-player logics of imperfect information

Logic...

To describe and study games (Game Logic, Coalition Logic, ...);

Game semantics for multi-player logics of imperfect information

Logic...

To describe and study games (Game Logic, Coalition Logic, ...);

Studied through game-theoretical techniques (Ehrenfeucht-Fraïssé games, forcing, ...);

Game semantics for multi-player logics of imperfect information

Logic...

To describe and study games (Game Logic, Coalition Logic, ...);

Studied through game-theoretical techniques (Ehrenfeucht-Fraïssé games, forcing, ...);

Embodying games (Game Semantics, Game Theoretic Semantics)

Game semantics for multi-player logics of imperfect information

Logic...

To describe and study games (Game Logic, Coalition Logic, ...);

Studied through game-theoretical techniques (Ehrenfeucht-Fraïssé games, forcing, ...);

Embodying games (**Game Semantics**, Game Theoretic Semantics)

Game semantics for FOL

$$\phi ::= Rx_1.,.x_n \mid \neg \phi \mid \phi \vee \phi \mid \exists x \phi$$

Game semantics for FOL

$$\phi ::= Rx_1.,.x_n \mid \neg \phi \mid \phi \vee \phi \mid \exists x \phi$$

Associate to each sentence ϕ a game structure $||\phi||_G$, as follows:

Game semantics for FOL

$$\phi ::= Rx_1.,.x_n \mid \neg \phi \mid \phi \vee \phi \mid \exists x \phi$$

Associate to each sentence ϕ a game structure $||\phi||_G$, as follows:

Two Players, **I** and **II** (or Abelard and Eloise);

Positions of $||\phi||_G$ are tuples (ψ, s, t), where $\psi \in \mathrm{Sub}(\phi)$, s set of *annotations*, t $\in$ {**I**, **II**};

Starting position = (ϕ, $\emptyset$, **II**);

At position (ψ, s, t), Player t chooses the next position;

Game semantics for FOL

Position	**Successors**
$(\neg\psi, s, t)$	(ψ, s, t')

Game semantics for FOL

Position	**Successors**
$(\neg\psi, s, t)$	(ψ, s, t')
$(\psi \vee \theta, s, t)$	$(\psi, s, t), (\theta, s, t)$

Game semantics for FOL

Position	**Successors**
$(\neg\psi, s, t)$	(ψ, s, t')
$(\psi \vee \theta, s, t)$	$(\psi, s, t), (\theta, s, t)$
$(\exists x\ \psi, s, t)$	$(\psi, s + \{x:e\}, t)$ *$e \in E$ (domain of quantification)*

Game semantics for FOL

Position	**Successors**
$(\neg\psi, s, t)$	(ψ, s, t')
$(\psi \vee \theta, s, t)$	$(\psi, s, t), (\theta, s, t)$
$(\exists x\ \psi, s, t)$	$(\psi, s + \{x{:}e\}, t)$ *$e \in E$ (domain of quantification)*
$(Rx_1.,.x_n, s, t)$	none

Game semantics for FOL

Given a game structure $||\phi||_G$ and a model M, we can *instantiate* $||\phi||_G$ with M.

The *domain of quantification* becomes Dom(M);

A position (ϕ, s, t), ϕ atomic, is *winning* for t iff

M, s |= ϕ

Game semantics for FOL

(Van Benthem, 2003): **Logic games are complete for game logic**

If a power statement (of game logic) is falsifiable by abstract sequential games then it is falsifiable by first-order evaluation games.

Game semantics for FOL

If now we consider winning strategies in these games, we move from Game Semantics to Game Theoretic Semantics.

Game semantics for FOL

If now we consider winning strategies in these games, we move from Game Semantics to Game Theoretic Semantics.

$$V(||\phi||_G(M)) = \begin{cases} 1 & \text{if } \mathbf{II} \text{ has a winning strategy;} \\ 0 & \text{if } \mathbf{I} \text{ has a winning strategy.} \end{cases}$$

Game semantics for FOL

If now we consider winning strategies in these games, we move from Game Semantics to Game Theoretic Semantics.

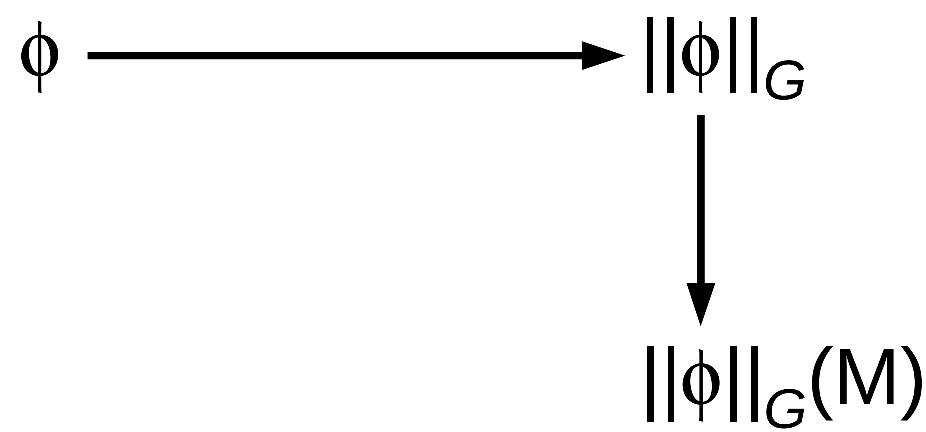
$$V(||\phi||_G(M)) = \begin{cases} 1 & \text{if } \mathbf{II} \text{ has a winning strategy;} \\ 0 & \text{if } \mathbf{I} \text{ has a winning strategy.} \end{cases}$$

$$\phi \longrightarrow ||\phi||_G$$

Game semantics for FOL

If now we consider winning strategies in these games, we move from Game Semantics to Game Theoretic Semantics.

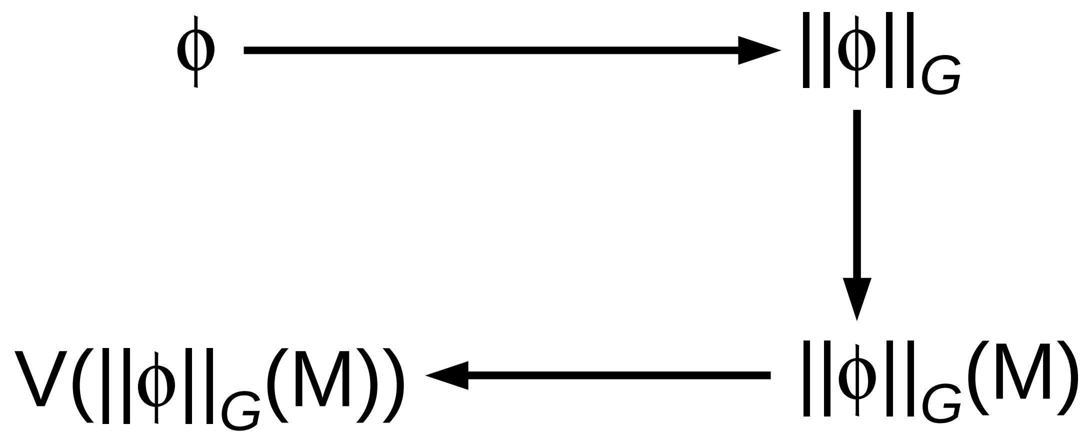
$$V(||\phi||_G(M)) = \begin{cases} 1 & \text{if } \mathbf{II} \text{ has a winning strategy;} \\ 0 & \text{if } \mathbf{I} \text{ has a winning strategy.} \end{cases}$$



Game semantics for FOL

If now we consider winning strategies in these games, we move from Game Semantics to Game Theoretic Semantics.

$$V(||\phi||_G(M)) = \begin{cases} 1 & \text{if } \mathbf{II} \text{ has a winning strategy;} \\ 0 & \text{if } \mathbf{I} \text{ has a winning strategy.} \end{cases}$$



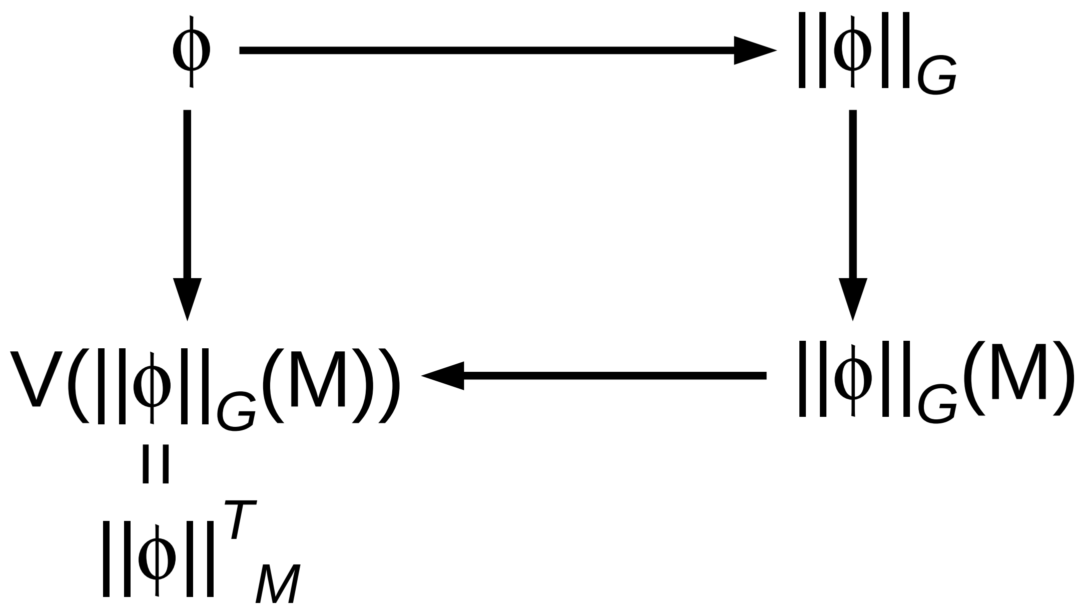
Game semantics for FOL

If now we consider winning strategies in these games, we move from Game Semantics to Game Theoretic Semantics.

$$V(||\phi||_G(M)) = \begin{cases} 1 & \text{if } \mathbf{II} \text{ has a winning strategy;} \\ 0 & \text{if } \mathbf{I} \text{ has a winning strategy.} \end{cases}$$

For First Order Logic,

GTS ≡ Tarski Semantics

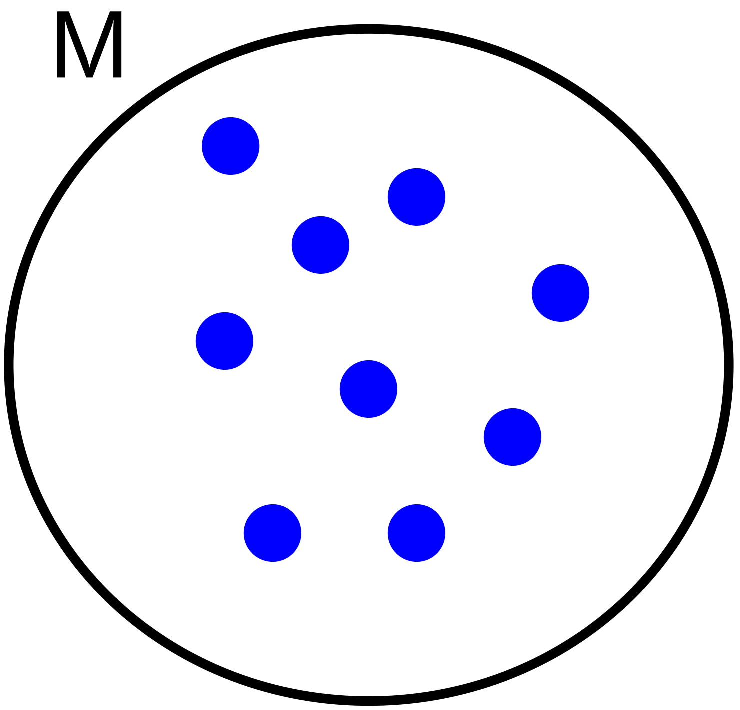


A very trivial example

$$\phi = \forall x\ \exists y\ (x=y)$$

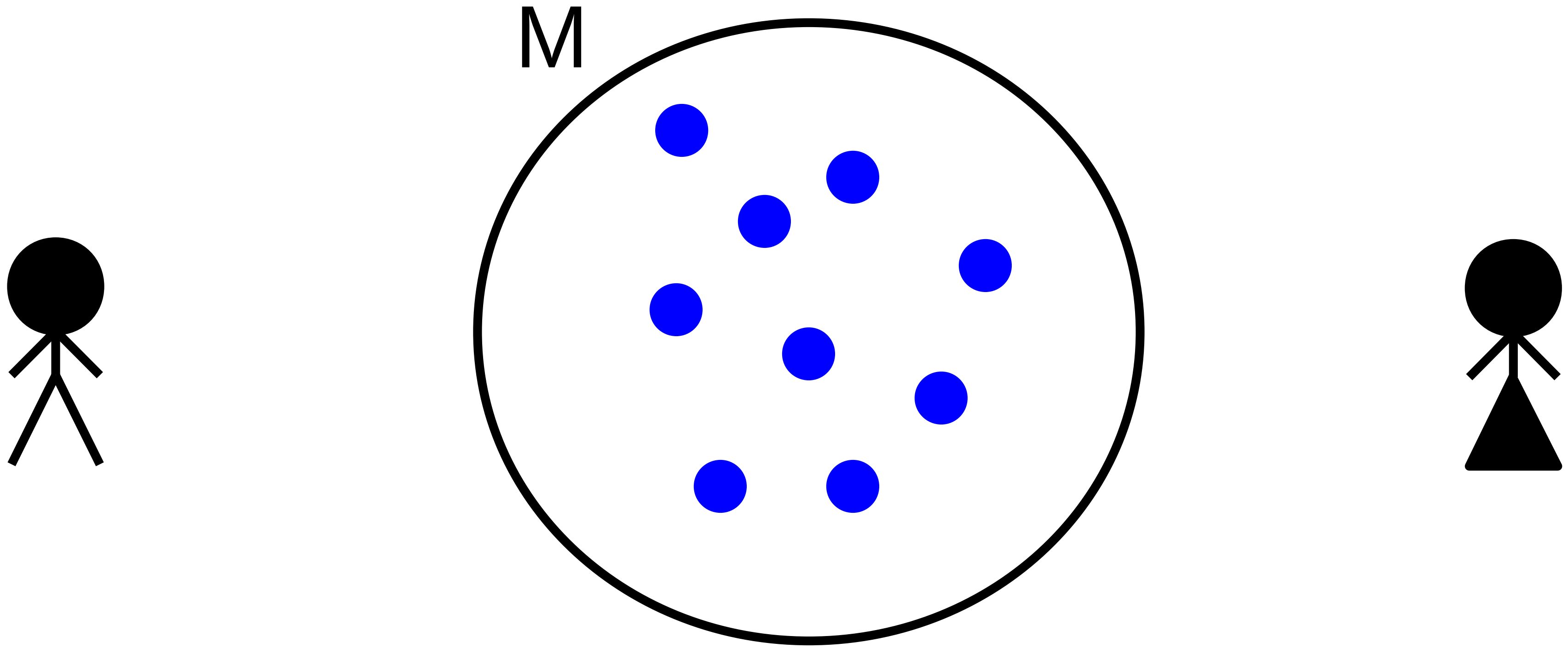
A very trivial example

$$\phi = \forall x\ \exists y\ (x=y)$$



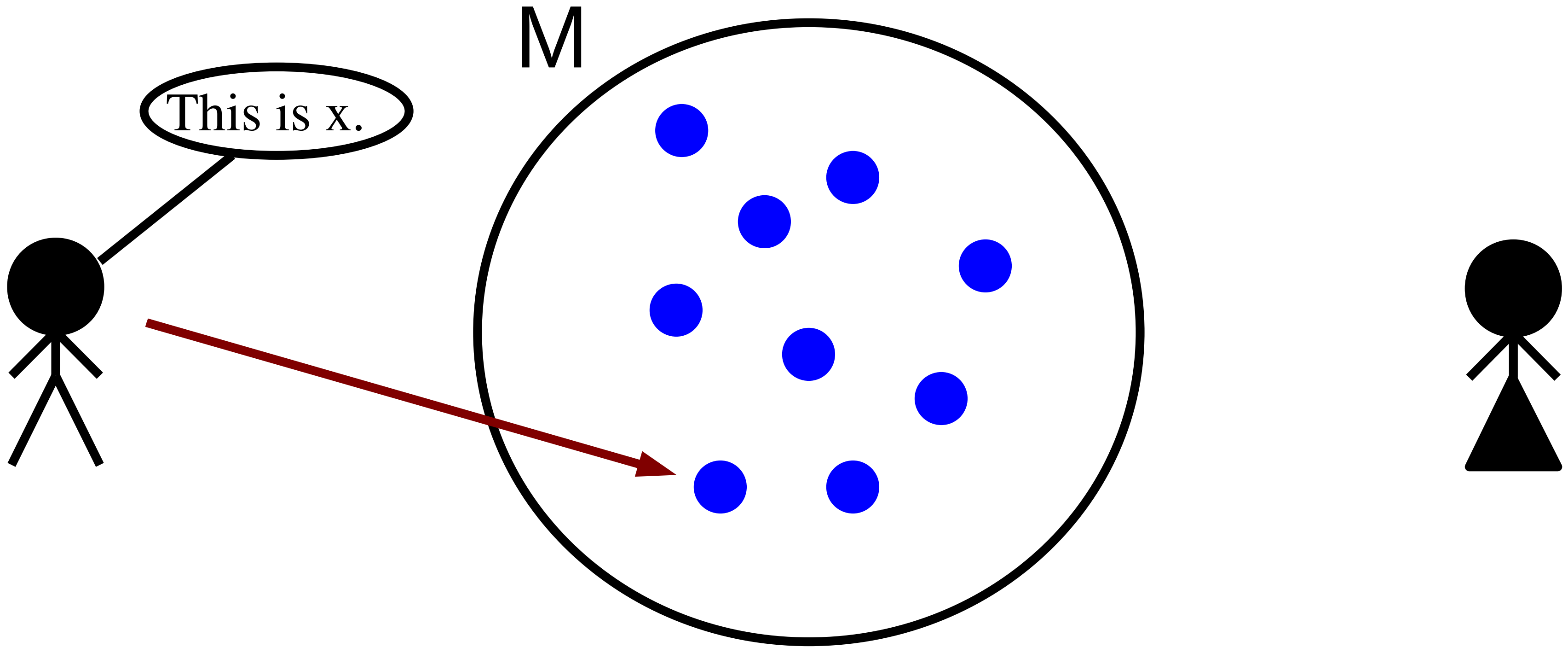
A very trivial example

$$\phi = \forall x\ \exists y\ (x=y)$$



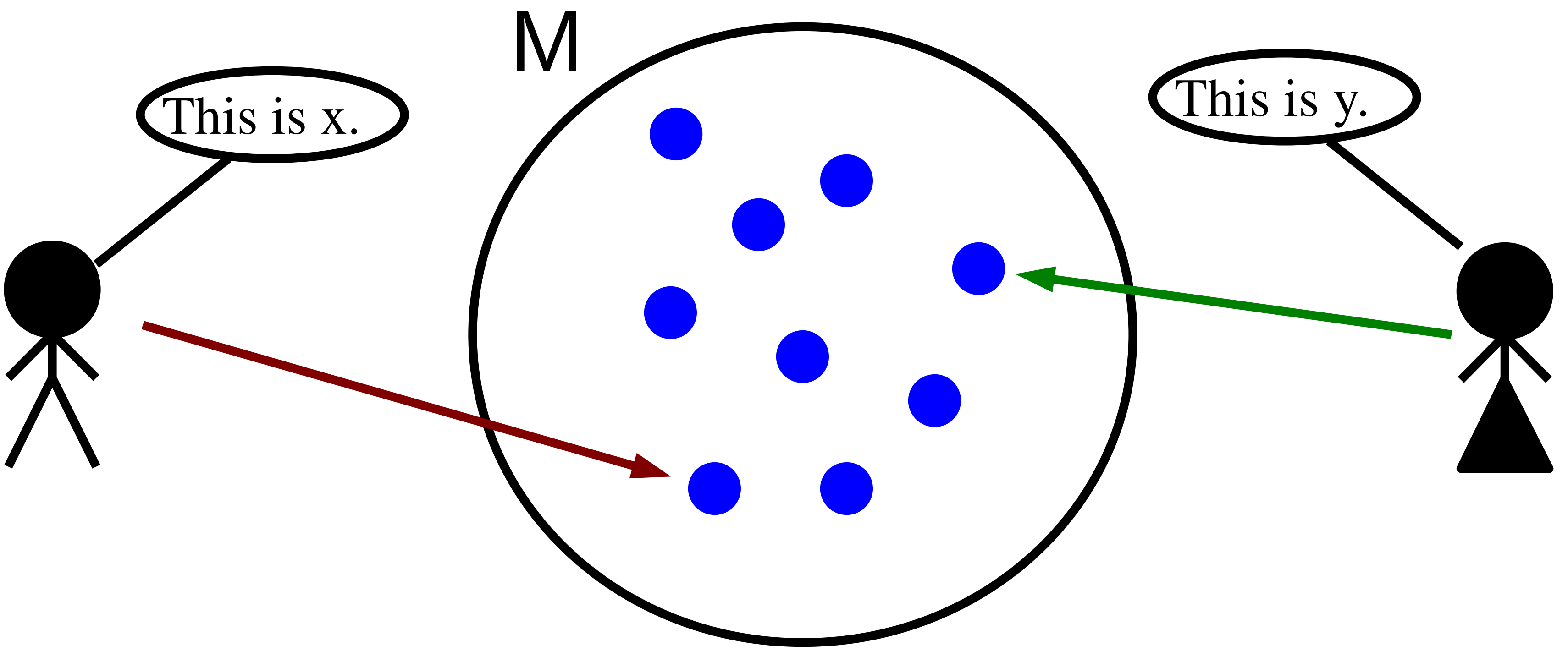
A very trivial example

$$\phi = \forall x\, \exists y\, (x=y)$$



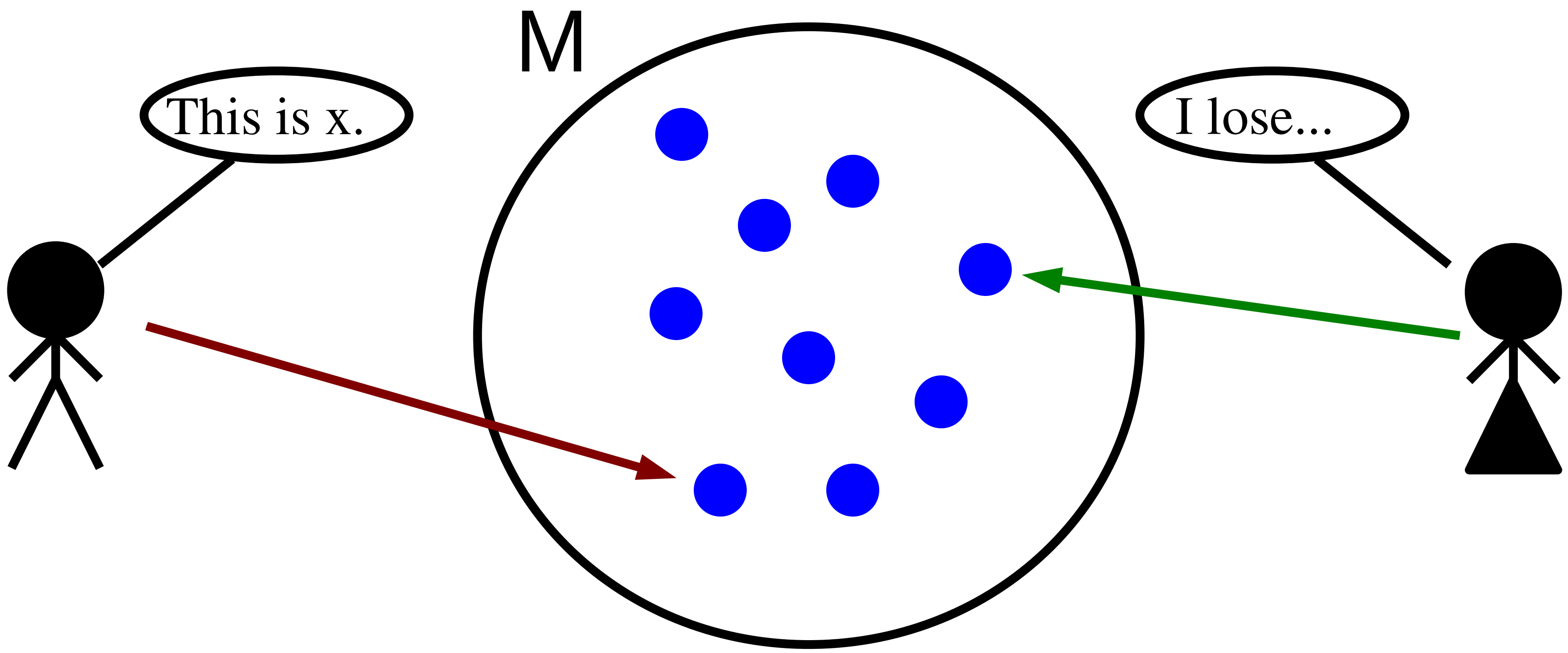
A very trivial example

$$\phi = \forall x \, \exists y \, (x=y)$$



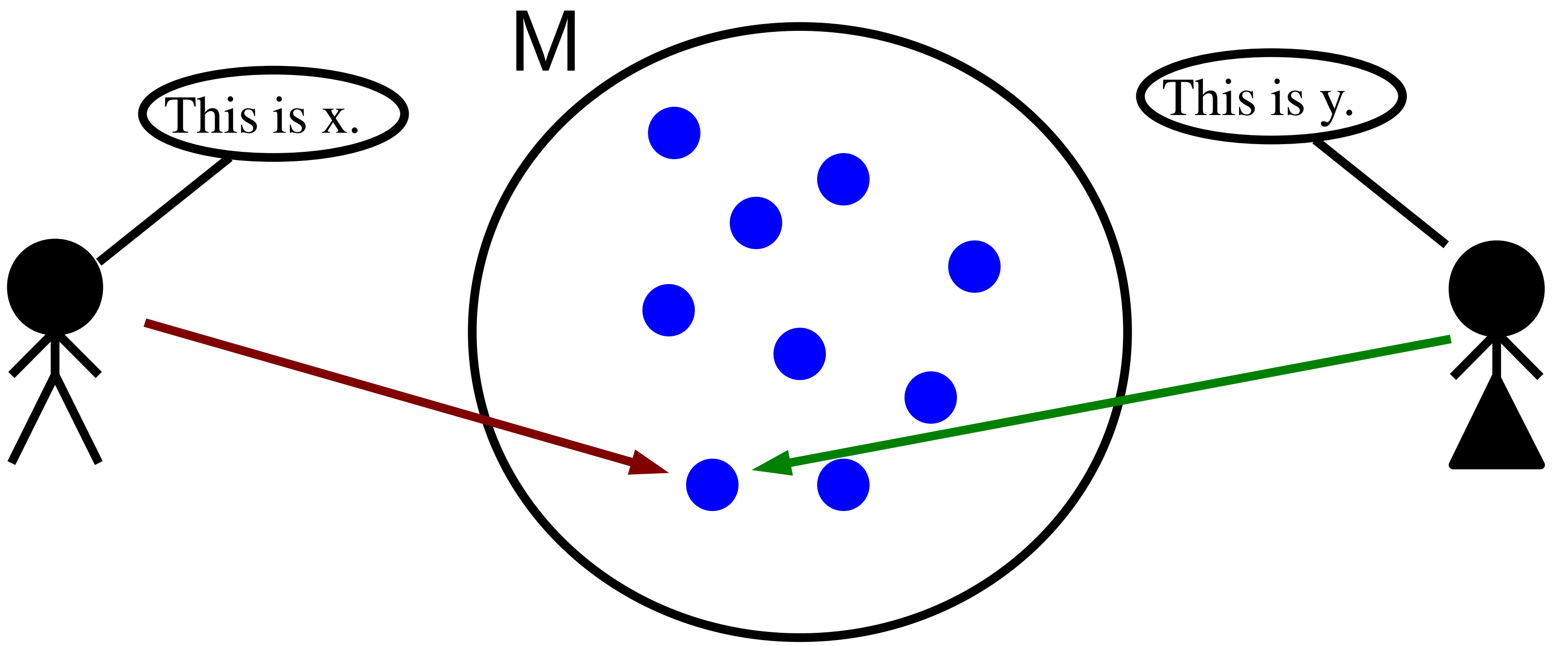
A very trivial example

$$\phi = \forall x \, \exists y \, (x = y)$$



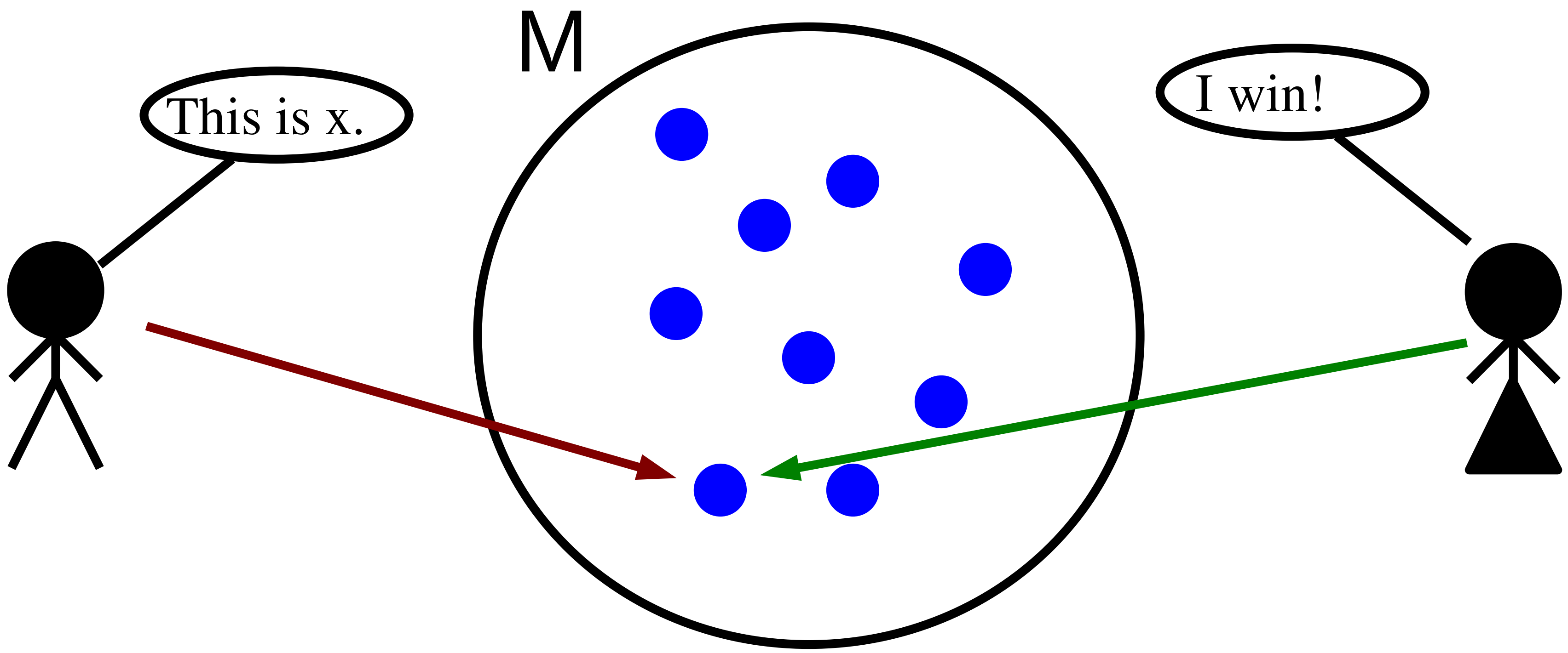
A very trivial example

$$\phi = \forall x \, \exists y \, (x=y)$$



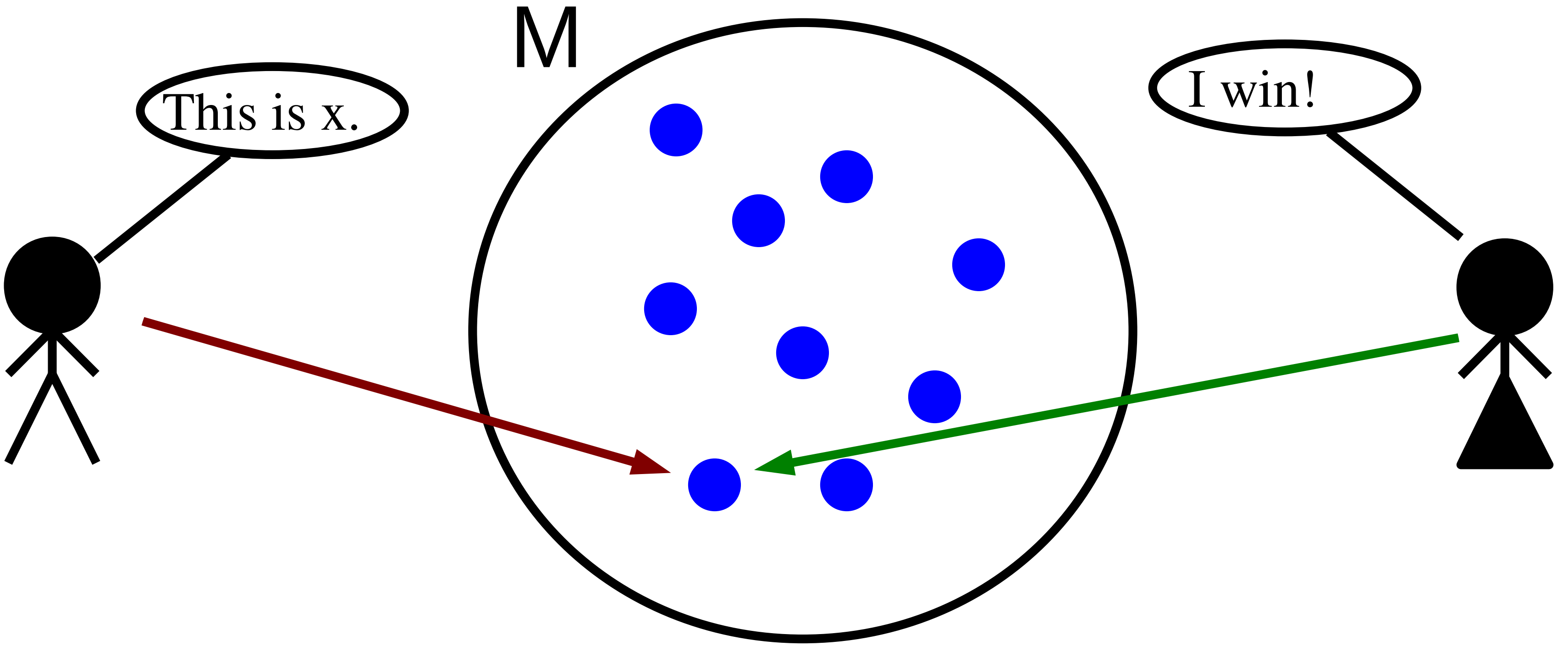
A very trivial example

$$\phi = \forall x\ \exists y\ (x=y)$$



A very trivial example

$$\phi = \forall x\ \exists y\ (x=y)$$



Eloise has a winning strategy; therefore, M |= ϕ

Multi Player Logics

1. Can we generalize these techniques?

2. Which sort of logics would correspond to multi-player games?

Multi Player Logics

0. Why would one be interested in multi-player logics?

Multi Player Logics

0. Why would one be interested in multi-player logics?

Communication protocols (also data compression):

$$\forall x_1 \ldots x_n \,(D(x_1 \ldots x_n) \rightarrow (\exists y_1 \ldots y_m \,(M(y_1 \ldots y_m) \wedge$$
$$\wedge\, (\exists (z_1 \ldots z_n) \backslash \{y_1 .. y_m\})\, (z_1 = x_1 \wedge \ldots \wedge z_n = x_n)))$$

Multi Player Logics

0. Why would one be interested in multi-player logics?

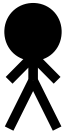
Communication protocols (also data compression):

$$\forall \mathbf{x_1 \ldots x_n} \, (\mathbf{D(x_1 \ldots x_n)}) \rightarrow (\exists y_1 \ldots y_m \, (M(y_1 \ldots . y_m) \wedge$$
$$\wedge \, (\exists (z_1 \ldots z_n) \backslash \{y_1 .. y_m\}) \, (z_1 = x_1 \wedge \ldots \wedge z_n = x_n)))$$

Information Source

$x_1 \ldots x_n$

$D(x_1 \ldots x_n)$

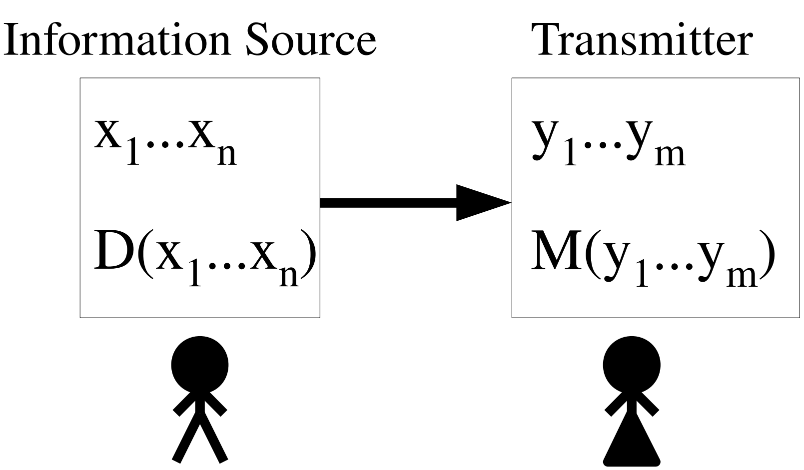


Multi Player Logics

0. Why would one be interested in multi-player logics?

Communication protocols (also data compression):

$$\forall x_1 \ldots x_n \,(D(x_1 \ldots x_n) \rightarrow (\exists \mathbf{y_1 \ldots y_m} \,(\mathbf{M(y_1 \ldots y_m)} \wedge$$
$$\wedge\, (\exists (z_1 \ldots z_n) \backslash \{y_1 .. y_m\})\,(z_1 = x_1 \wedge \ldots \wedge z_n = x_n)))$$

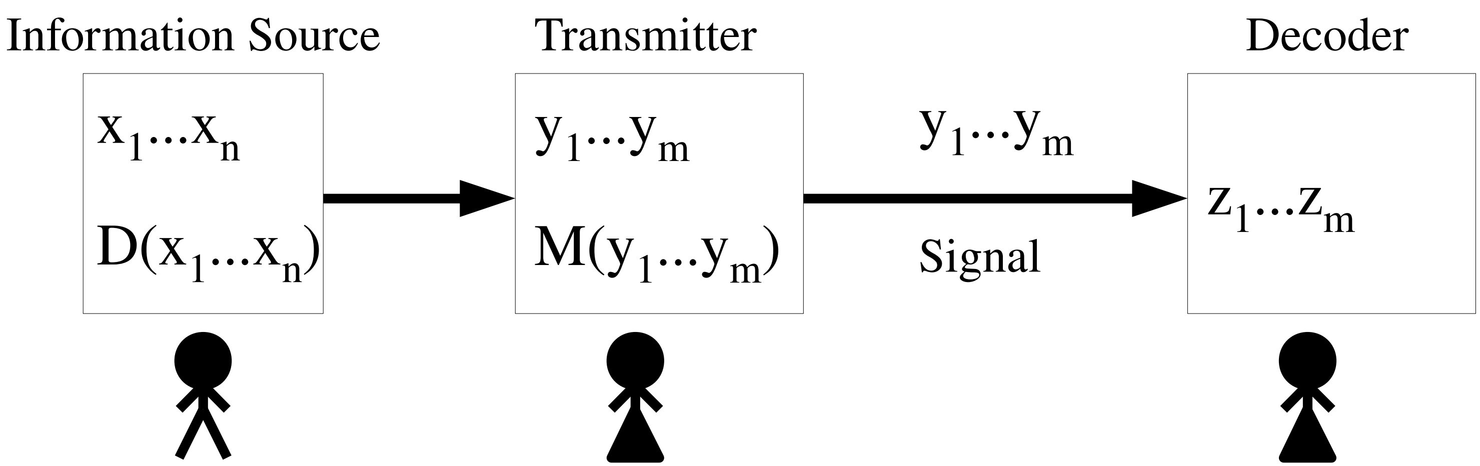


Multi Player Logics

0. Why would one be interested in multi-player logics?

Communication protocols (also data compression):

$$\forall x_1 \ldots x_n \,(D(x_1 \ldots x_n) \rightarrow (\exists y_1 \ldots y_m \,(M(y_1 \ldots y_m) \wedge$$
$$\wedge\, (\boldsymbol{\exists (z_1 \ldots z_n) \backslash \{y_1 .. y_m\}})\, (z_1 = x_1 \wedge \ldots \wedge z_n = x_n)))$$

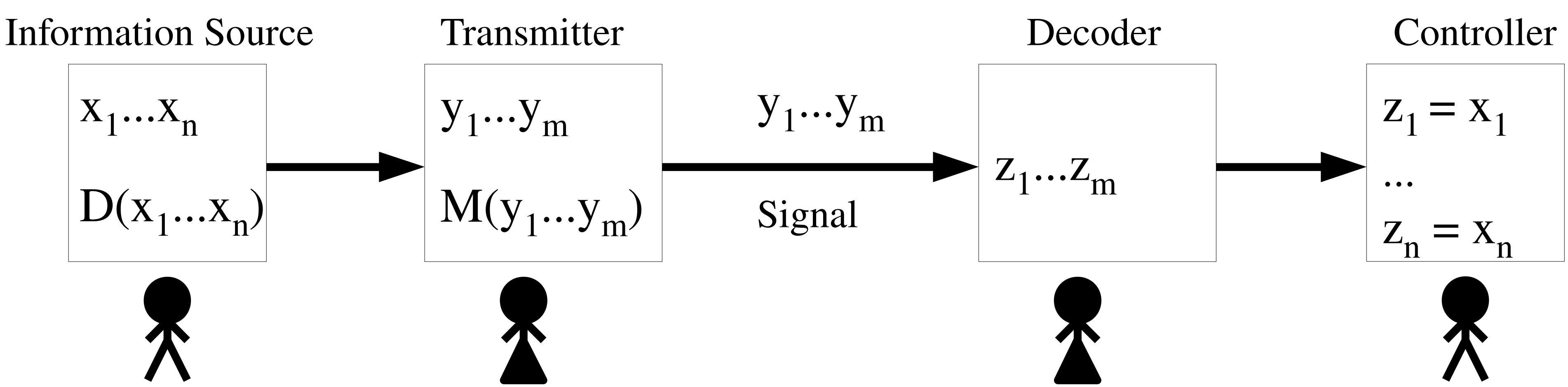


Multi Player Logics

0. Why would one be interested in multi-player logics?

Communication protocols (also data compression):

$$\forall x_1 \ldots x_n \, (D(x_1 \ldots x_n) \rightarrow (\exists y_1 \ldots y_m \, (M(y_1 \ldots . y_m) \wedge$$
$$\wedge \, (\exists (z_1 \ldots z_n) \backslash \{y_1 .. y_m\}) \, (\mathbf{z_1 = x_1 \wedge \ldots \wedge z_n = x_n})))$$



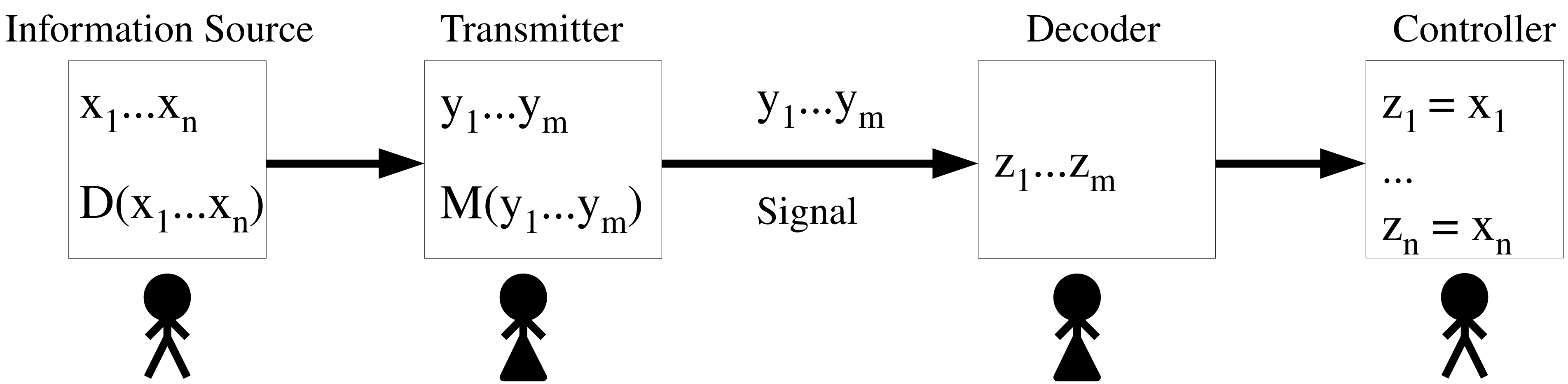
Multi Player Logics

0. Why would one be interested in multi-player logics?

Communication protocols (also data compression):

$$\forall x_1 \ldots x_n \, (D(x_1 \ldots x_n) \rightarrow (\exists y_1 \ldots y_m \, (M(y_1 \ldots . y_m) \wedge$$

$$\wedge \, (\exists (z_1 \ldots z_n) \backslash \{y_1 .. y_m\}) \, (z_1 = x_1 \wedge \ldots \wedge z_n = x_n)))$$



Multi Player Logics

Cryptography:

$\forall x_1 \ldots x_n \exists y_1 \ldots y_m((\exists(z_1 \ldots z_n)\backslash\{y_1 .,. y_m\}) (z_1{=}x_1 \wedge \ldots \wedge z_n{=}x_n) \wedge$

$\wedge (\forall(w_1 \ldots w_n)\backslash\{y_1 .,. y_m\})(w_1 \neq x_1 \vee \ldots \vee w_n \neq x_n))$

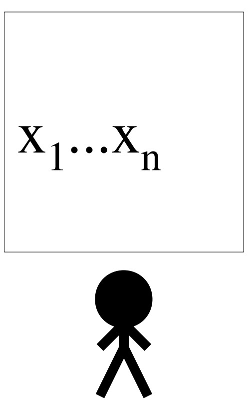
Multi Player Logics

Cryptography:

$$\forall \mathbf{x_1 \ldots x_n} \exists y_1 \ldots y_m ((\exists (z_1 \ldots z_n) \setminus \{y_1 .,. y_m\}) (z_1 = x_1 \wedge \ldots \wedge z_n = x_n) \wedge$$

$$\wedge (\forall (w_1 \ldots w_n) \setminus \{y_1 .,. y_m\}) (w_1 \neq x_1 \vee \ldots \vee w_n \neq x_n))$$

Information Source

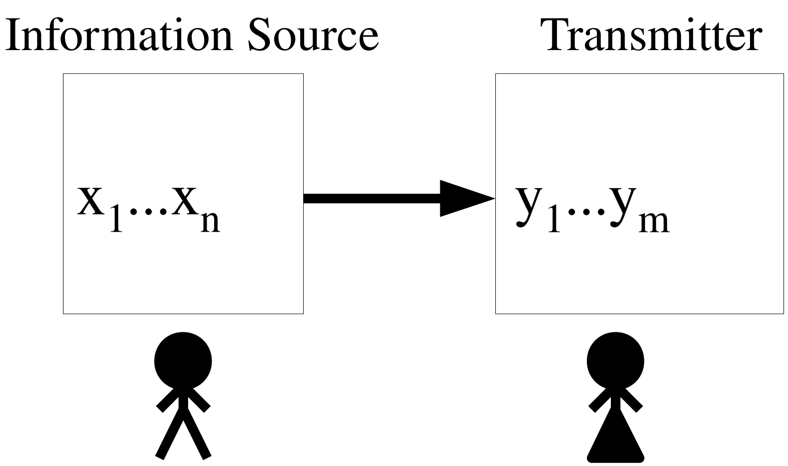


Multi Player Logics

Cryptography:

$\forall x_1 \ldots x_n \mathbf{\exists y_1 \ldots y_m}((\exists(z_1 \ldots z_n)\backslash\{y_1.,.y_m\})\ (z_1 = x_1 \wedge \ldots \wedge\ z_n = x_n) \wedge$

$\wedge\ (\forall(w_1 \ldots w_n)\backslash\{y_1.,.y_m\})(w_1 \neq x_1\ \vee \ldots \vee\ w_n \neq x_n))$

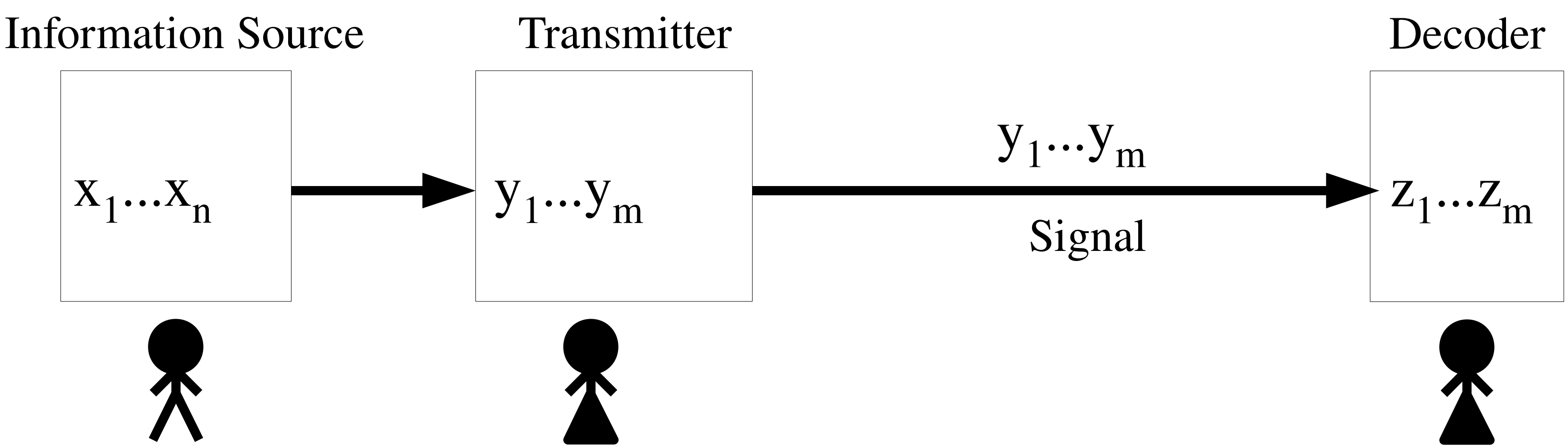


Multi Player Logics

Cryptography:

$\forall x_1 \ldots x_n \exists y_1 \ldots y_m ((\boldsymbol{\exists (z_1 \ldots z_n)} \backslash \boldsymbol{\{y_1 .,. y_m\}}) (z_1 = x_1 \wedge \ldots \wedge z_n = x_n) \wedge$

$\wedge (\forall (w_1 \ldots w_n) \backslash \{y_1 .,. y_m\})(w_1 \neq x_1 \vee \ldots \vee w_n \neq x_n))$

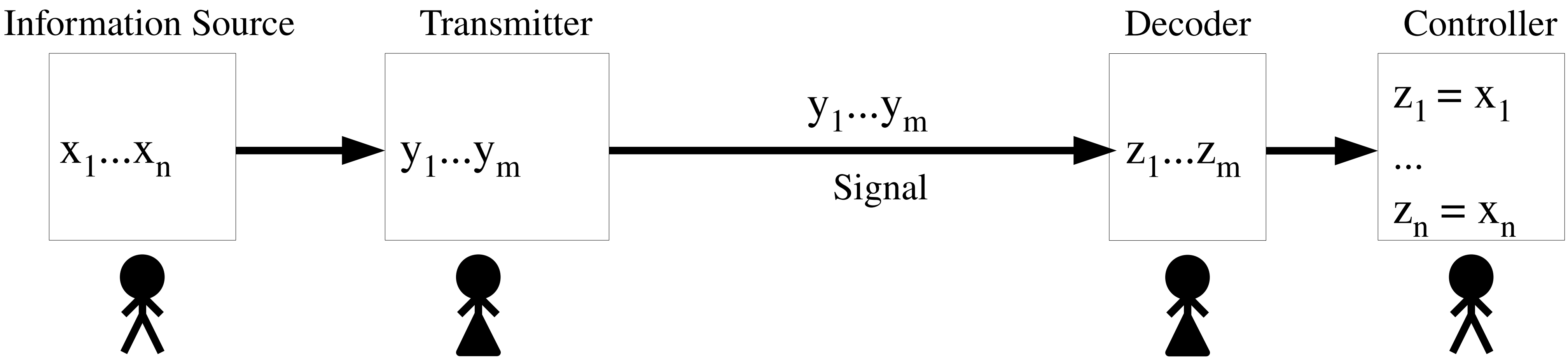


Multi Player Logics

Cryptography:

$$\forall x_1 \ldots x_n \exists y_1 \ldots y_m((\exists(z_1 \ldots z_n)\backslash\{y_1.,.y_m\})\ (\mathbf{z_1{=}x_1 \wedge \ldots \wedge z_n{=}x_n}) \wedge$$

$$\wedge\ (\forall(w_1 \ldots w_n)\backslash\{y_1.,.y_m\})(w_1 \neq x_1 \vee \ldots \vee w_n \neq x_n))$$

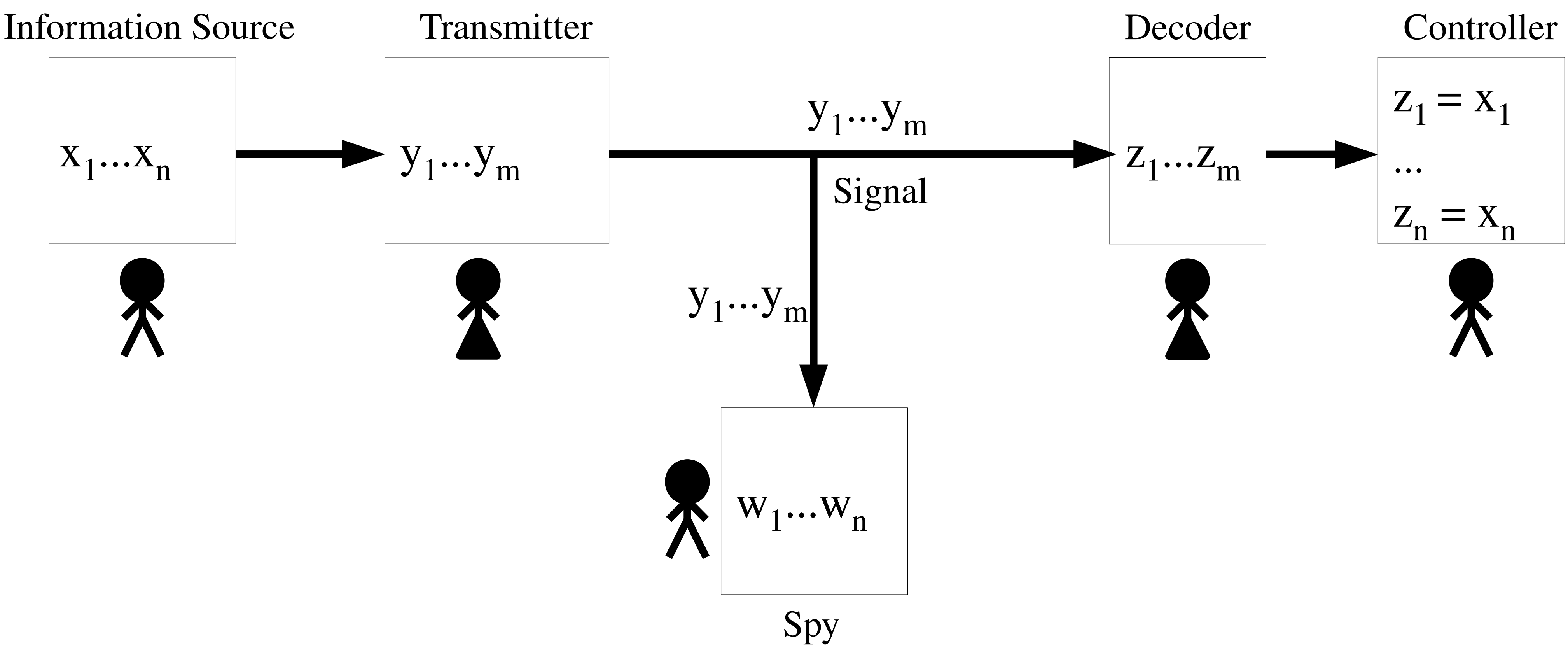


Multi Player Logics

Cryptography:

$\forall x_1 \ldots x_n \exists y_1 \ldots y_m ((\exists (z_1 \ldots z_n) \backslash \{y_1 .,. y_m\}) (z_1 = x_1 \wedge \ldots \wedge z_n = x_n) \wedge$

$\wedge \, \mathbf{(\forall (w_1 \ldots w_n) \backslash \{y_1 .,. y_m\})} (w_1 \neq x_1 \vee \ldots \vee w_n \neq x_n))$

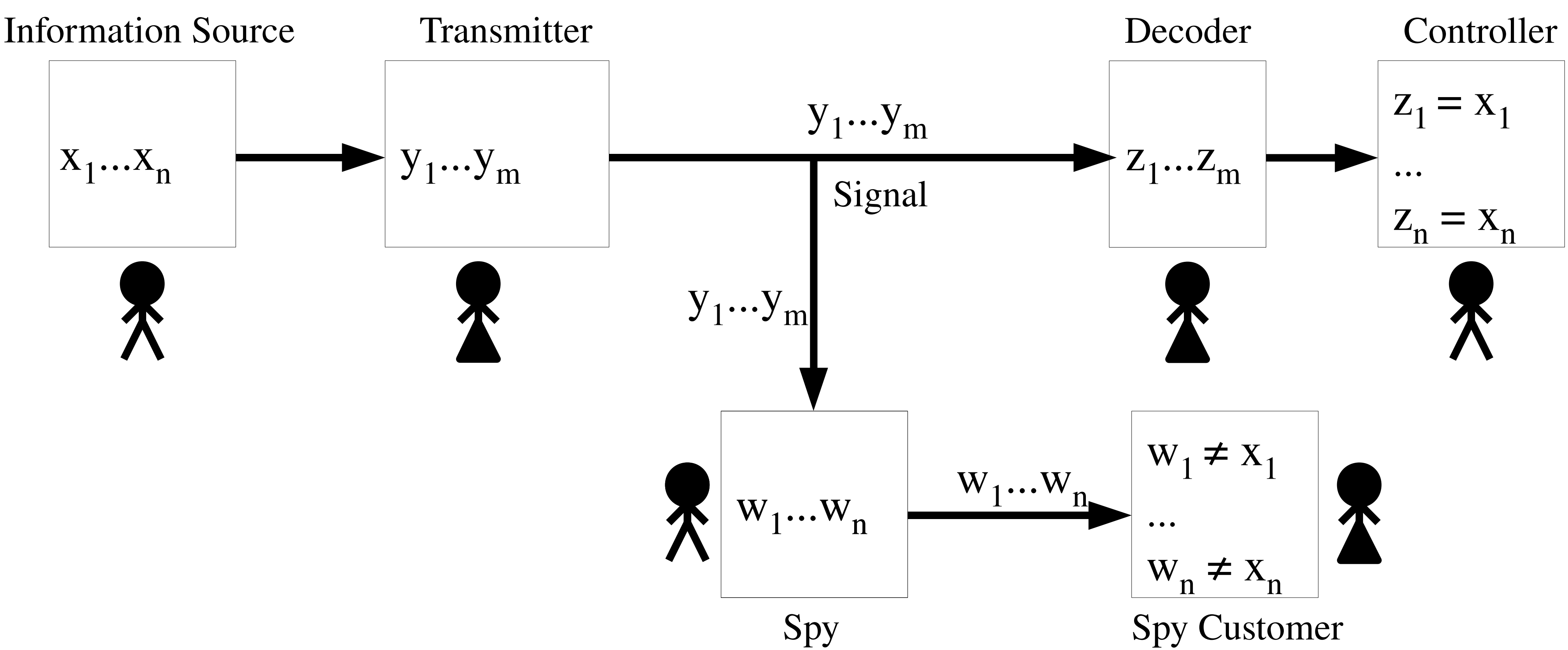


Multi Player Logics

Cryptography:

$$\forall x_1 \ldots x_n \exists y_1 \ldots y_m((\exists(z_1 \ldots z_n)\backslash\{y_1.,.y_m\})\ (z_1{=}x_1 \wedge \ldots \wedge z_n{=}x_n) \wedge$$

$$\wedge\ (\forall(w_1 \ldots w_n)\backslash\{y_1.,.y_m\})(\mathbf{w_1{\neq}x_1 \vee \ldots \vee\ w_n{\neq}x_n}))$$

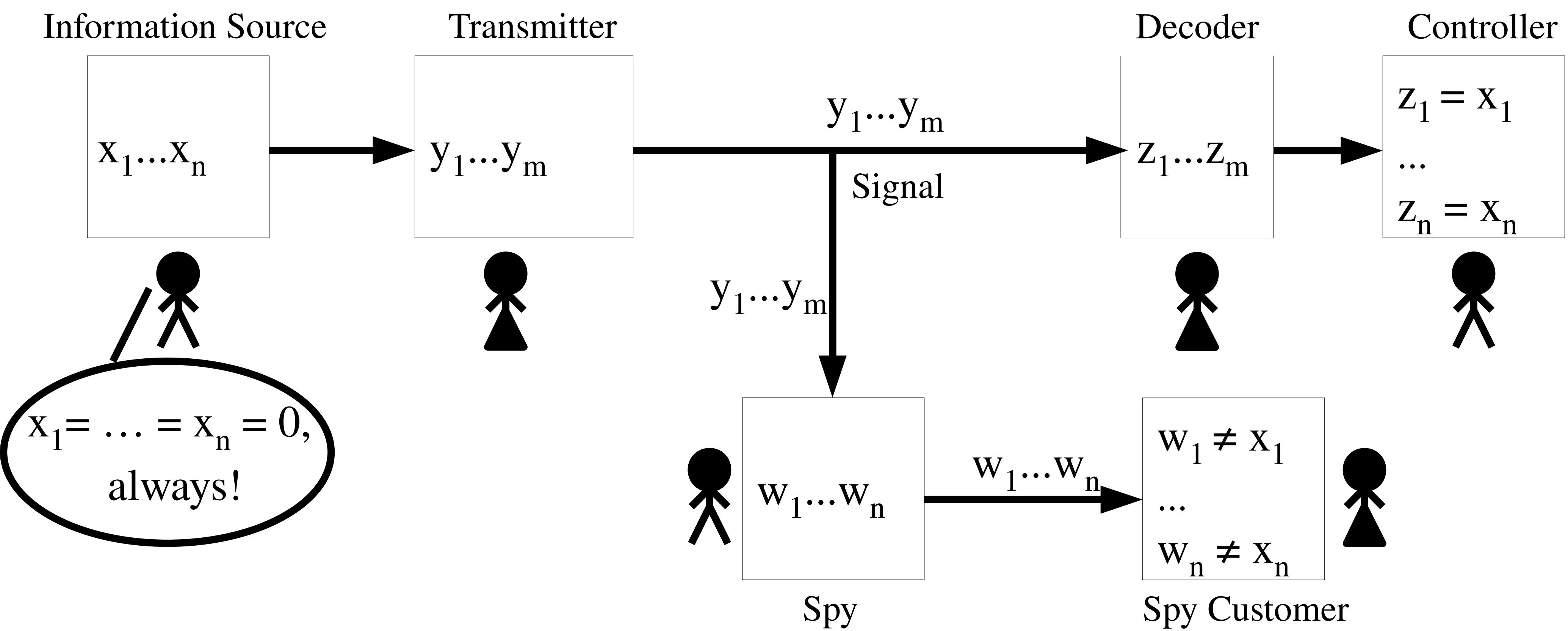


Multi Player Logics

Cryptography:

$$\forall x_1 \ldots x_n \exists y_1 \ldots y_m ((\exists (z_1 \ldots z_n) \backslash \{y_1 .,. y_m\})\ (z_1 = x_1 \wedge \ldots \wedge z_n = x_n) \wedge$$

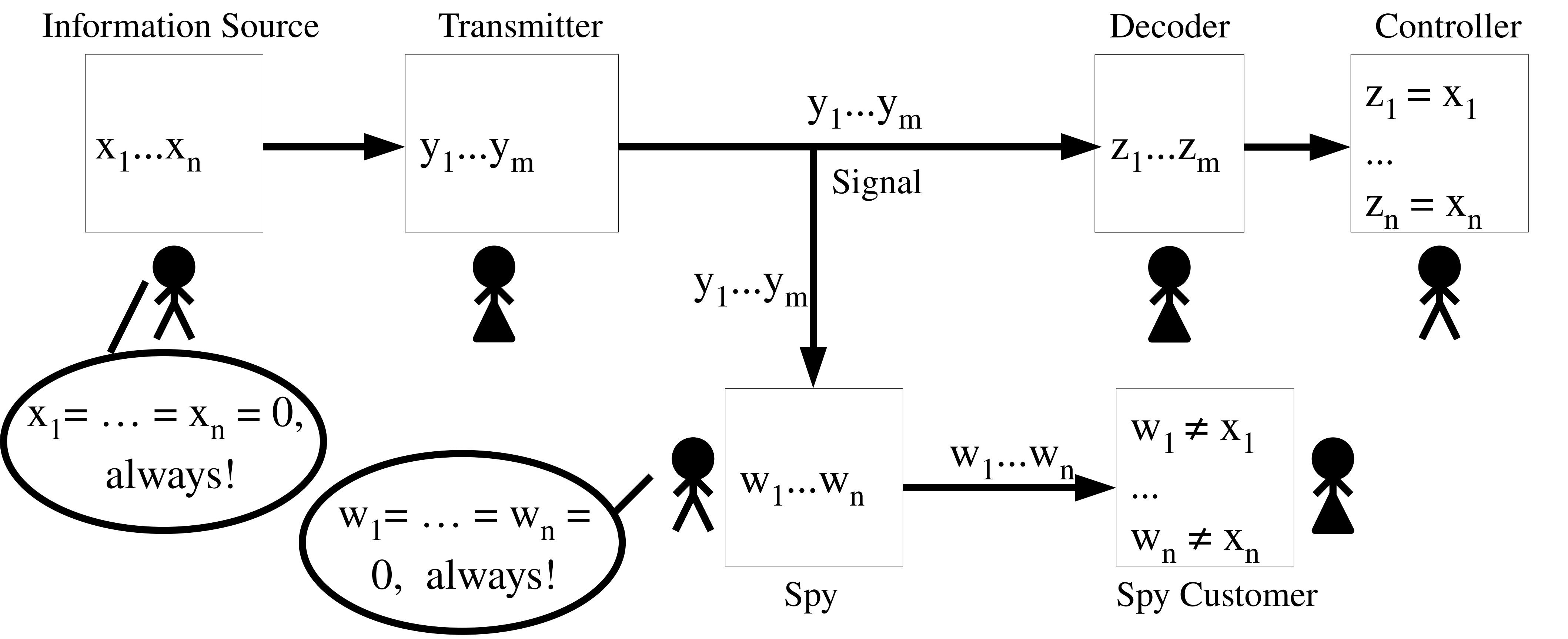
$$\wedge\ (\forall (w_1 \ldots w_n) \backslash \{y_1 .,. y_m\})(w_1 \neq x_1 \vee \ldots \vee w_n \neq x_n))$$



Multi Player Logics

Cryptography:

$$\forall x_1 \ldots x_n \exists y_1 \ldots y_m((\exists(z_1 \ldots z_n)\backslash\{y_1.,.y_m\})\ (z_1{=}x_1 \wedge \ldots \wedge z_n{=}x_n) \wedge$$
$$\wedge\ (\forall(w_1 \ldots w_n)\backslash\{y_1.,.y_m\})(w_1{\neq}x_1 \vee \ldots \vee w_n{\neq}x_n))$$

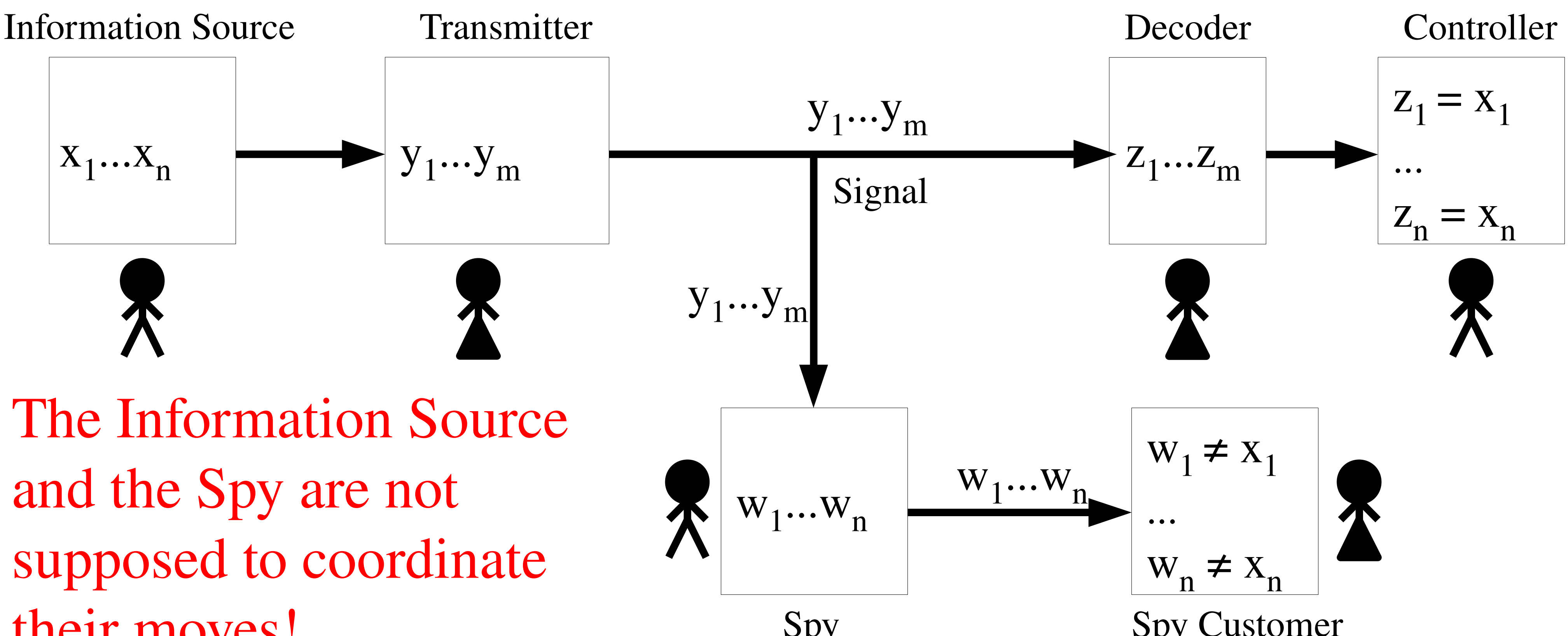


Multi Player Logics

Cryptography:

$$\forall x_1 \ldots x_n \exists y_1 \ldots y_m ((\exists (z_1 \ldots z_n) \backslash \{y_1 .,. y_m\}) \, (z_1 = x_1 \wedge \ldots \wedge z_n = x_n) \wedge$$

$$\wedge \, (\forall (w_1 \ldots w_n) \backslash \{y_1 .,. y_m\})(w_1 \neq x_1 \vee \ldots \vee w_n \neq x_n))$$



The Information Source and the Spy are not supposed to coordinate their moves!

Multi Player Logics

Abramsky, 2006: Socially Responsive, Environmentally Frienly Logic

Abramsky, 2007: A Compositional Game Semantics for Multi-Agent Logics of Partial Information

See also:

Tulenheimo, Venema, 2007: Propositional Logic for Three;
Tulenheimo, 2007: Satisfiability and Validity from a 3-player Perspective;
Loohuis, 2008: Multi-player Logics;
Loohuis, Venema, 2009: Logics and Algebras for Multiple Players;

Concrete Data Structures

Concrete Data Structures (Kahn, Plotkin, 1975):

$$M = (C, V, D, \vdash)$$

Concrete Data Structures

Concrete Data Structures (Kahn, Plotkin, 1975):

$$M = (C, V, D, \vdash)$$

C = *cells* (loci of decisions);

V = *values* (that a cell can take);

D = *decisions* ($\subseteq C \times V$);

|- = *enabling relation* ($\subseteq \mathcal{P}_{\mathbf{f}}(D) \times C$)

Concrete Data Structures

A state *s* over M is a subset of D such that

$(c, v_1), (c, v_2) \in s \Rightarrow v_1 = v_2$;

If $(c, v) \in s$ then $\exists\ (c_1, v_1) \ldots (c_k, v_k) = (c,v)$ s.t., $\forall\ i \in 1 \ldots k$,

$(c_i, v_i) \in s$ and $\Gamma_i \vdash c_i$ for some $\Gamma_i \subseteq \{(c_1, v_1)..(c_{i-1}, v_{i-1})\}$

Concrete Data Structures

$\lambda : C \rightarrow A$ *labeling function*, A = the set of players

Concrete Data Structures

$\lambda : C \rightarrow A$ *labeling function*, A = the set of players

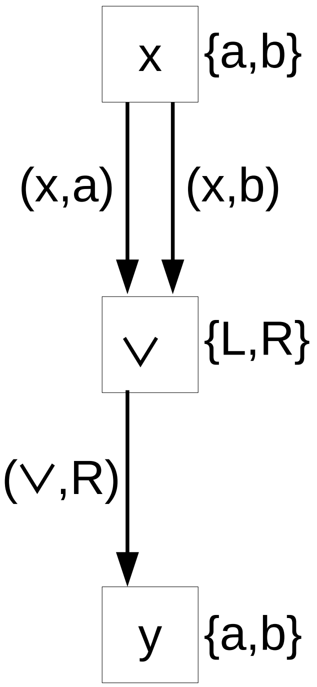
$$\forall x\ (P(x) \vee \exists y\ Q(y))$$

$C = \{x, y, \vee\};$

$V = \{a, b, L, R\};$

$D = \{(x, a), (x, b), (y, a), (y, b), (\vee, L), (\vee, R)\};$

$\vdash = \{ (\{(x, a)\}, \vee), (\{(x,b)\}, \vee), (\{(\vee, R)\}, y)\}.$



Concrete Data Structures

Primitives:

Atomic formulas: empty game ($\emptyset$, $\emptyset$, $\emptyset$, $\emptyset$)

Concrete Data Structures

Primitives:

Atomic formulas: empty game $(\emptyset, \emptyset, \emptyset, \emptyset)$;

Quantifiers: Q_α, $\alpha \in A$:

$(\{c_0\}, E, \{c_0\} \times E, \vdash_Q)$, where $\vdash_Q = \{(\emptyset, c_0)\}$

$\lambda(c_0) = \alpha$

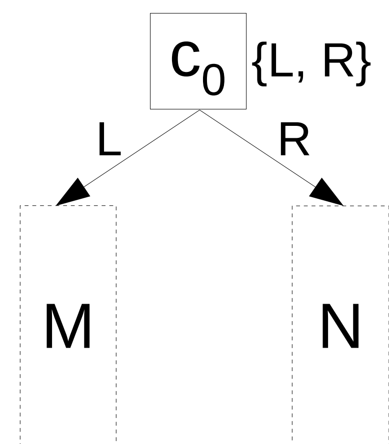
c_0 $E = \{a,b,c,\ldots\}$

Concrete Data Structures

Operations:

Choice connectives: $M \oplus_\alpha N$, $\alpha \in A$:

$(C_M \cup^d C_N \cup^d \{c_0\}, V_M \cup V_N \cup \{L, R\},$
$\quad D_M \cup^d D_N \cup^d \{(c_0, L), (c_0, R)\}, \vdash_{M\oplus_\alpha N})$



$(c_0, L), \Gamma \vdash_{M\oplus_\alpha N} c$ iff $\Gamma \vdash_M c$;

$(c_0, R), \Gamma \vdash_{M\oplus_\alpha N} c$ iff $\Gamma \vdash_N c$.

$\lambda_{M\oplus_\alpha N}(c_0) = \alpha$;

$c \in M \Rightarrow \lambda_{M\oplus_\alpha N}(c) = \lambda_M(c)$;

$c \in N \Rightarrow \lambda_{M\oplus_\alpha N}(c) = \lambda_N(c)$.

Concrete Data Structures

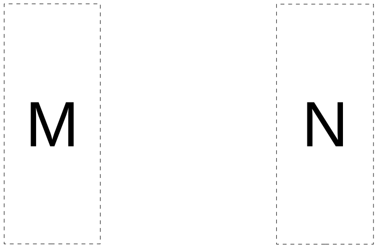
Operations:

Parallel Composition: $M \parallel N$:

$(C_M \cup^d C_N, V_M \cup V_N, D_M \cup^d D_N, \vdash_M \cup^d \vdash_N)$

$c \in M \Rightarrow \lambda_{M \oplus_\alpha N}(c) = \lambda_M(c);$

$c \in N \Rightarrow \lambda_{M \oplus_\alpha N}(c) = \lambda_N(c).$



Concrete Data Structures

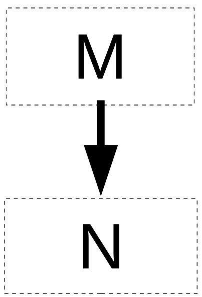
Operations:

Sequential Composition: $M \cdot N$:

$(C_M \cup^d C_N, V_M \cup V_N, D_M \cup^d D_N, \vdash_{M \cdot N})$

$\Gamma \vdash_{M \cdot N} c$ iff $\Gamma \vdash_M c$ or $\Gamma = s \cup \Delta$,

$s \in Max(M), \Delta \vdash_N c$;



$c \in M \Rightarrow \lambda_{M \oplus_\alpha N}(c) = \lambda_M(c)$;

$c \in N \Rightarrow \lambda_{M \oplus_\alpha N}(c) = \lambda_N(c)$.

Concrete Data Structures

Operations:

Role switching: $\pi(M)$, $\pi : A \to A$ permutation

$(C_M, V_M, D_M , |\text{-}_M)$

$c \in M \Rightarrow \lambda_{\pi(M)} = \pi(\lambda_M(c));$

Concrete Data Structures

(Tulenheimo, 2007) and (Tulenheimo and Venema, 2007):

From role permutations to *role distribution permutations*

Concrete Data Structures

(Tulenheimo, 2007) and (Tulenheimo and Venema, 2007):

From role permutations to *role distribution permutations*

Role

I

II

III

Concrete Data Structures

(Tulenheimo, 2007) and (Tulenheimo and Venema, 2007):

From role permutations to *role distribution permutations*

Role	Player
I	Abelard
II	Barbara
III	Charlie

Concrete Data Structures

(Tulenheimo, 2007) and (Tulenheimo and Venema, 2007):

From role permutations to *role distribution permutations*

(A,B,C) → (B,C,A); (A,C,B) → (C,B,A); (B,A,C) → (A,C,B);
(B,C,A) → (C,A,B); (C,A,B) → (A,B,C); (C,B,A) → (B,A,C).

Role	Player
I	Abelard
II	Barbara
III	Charlie

Concrete Data Structures

(Tulenheimo, 2007) and (Tulenheimo and Venema, 2007):

From role permutations to *role distribution permutations*

(A,B,C) → (B,C,A); (A,C,B) → (C,B,A); (B,A,C) → (A,C,B);
(B,C,A) → (C,A,B); (C,A,B) → (A,B,C); (C,B,A) → (B,A,C).

Role	Player	π(Player)
I	Abelard	Barbara
II	Barbara	Charlie
III	Charlie	Abelard

Concrete Data Structures

(Tulenheimo, 2007) and (Tulenheimo and Venema, 2007):

From role permutations to *role distribution permutations*

(A,B,C) → (B,C,A); (A,C,B) → (C,B,A); (B,A,C) → (A,C,B);
(B,C,A) → (C,A,B); (C,A,B) → (A,B,C); (C,B,A) → (B,A,C).

Role	Player	π(Player)	Player	π(Player)
I	Abelard	Barbara	Abelard	Charlie
II	Barbara	Charlie	Charlie	Barbara
III	Charlie	Abelard	Barbara	Abelard

Concrete Data Structures

(Tulenheimo, 2007) and (Tulenheimo and Venema, 2007):

From role permutations to *role distribution permutations*

(A,B,C) → (B,C,A); (A,C,B) → (C,B,A); (B,A,C) → (A,C,B);
(B,C,A) → (C,A,B); (C,A,B) → (A,B,C); (C,B,A) → (B,A,C).

Role	Player	π(Player)	Player	π(Player)
I	Abelard	Barbara	Abelard	Charlie
II	Barbara	Charlie	Charlie	Barbara
III	Charlie	Abelard	Barbara	Abelard

With n players, (n!)! “negations”!

Concrete Data Structures

$\phi ::= Rx_1.,.x_n \mid \pi(\phi) \mid \phi \oplus_\alpha \phi \mid Q_\alpha\, x \mid (\phi \,||\, \phi) \mid \phi \cdot \phi$

Then associate to every formula ϕ a labeled CDS $||\phi||_G$:

Concrete Data Structures

$\phi ::= Rx_1.,.x_n \mid \pi(\phi) \mid \phi \oplus_\alpha \phi \mid Q_\alpha\, x \mid (\phi \,||\, \phi) \mid \phi \cdot \phi$

Then associate to every formula ϕ a labeled CDS $||\phi||_G$:

$$(Q_I\, x \cdot (Px \oplus_{III} P'x)) \oplus_{II}\; Q_I\, x \cdot Q_{II}\, y \cdot R(x,y,y)$$

Concrete Data Structures

$\phi ::= Rx_1.,.x_n \mid \pi(\phi) \mid \phi \oplus_\alpha \phi \mid Q_\alpha\, x \mid (\phi \,||\, \phi) \mid \phi \cdot \phi$

Then associate to every formula ϕ a labeled CDS $||\phi||_G$:

$$(Q_I\, x \cdot (Px \oplus_{III} P'x)) \oplus_{II}\; Q_I\, x \cdot Q_{II}\, y \cdot R(x,y,y)$$

Concrete Data Structures

$\phi ::= Rx_1.,.x_n \mid \pi(\phi) \mid \phi \oplus_\alpha \phi \mid Q_\alpha\, x \mid (\phi \mid\mid \phi) \mid \phi \cdot \phi$

Then associate to every formula ϕ a labeled CDS $||\phi||_G$:

$$(Q_I\, x \cdot (Px \oplus_{III} P'x)) \oplus_{II} Q_I\, x \cdot Q_{II}\, y \cdot R(x,y,y)$$

L,R

E

Concrete Data Structures

$\phi ::= Rx_1.,.x_n \mid \pi(\phi) \mid \phi \oplus_\alpha \phi \mid Q_\alpha\, x \mid (\phi \,||\, \phi) \mid \phi \cdot \phi$

Then associate to every formula ϕ a labeled CDS $||\phi||_G$:

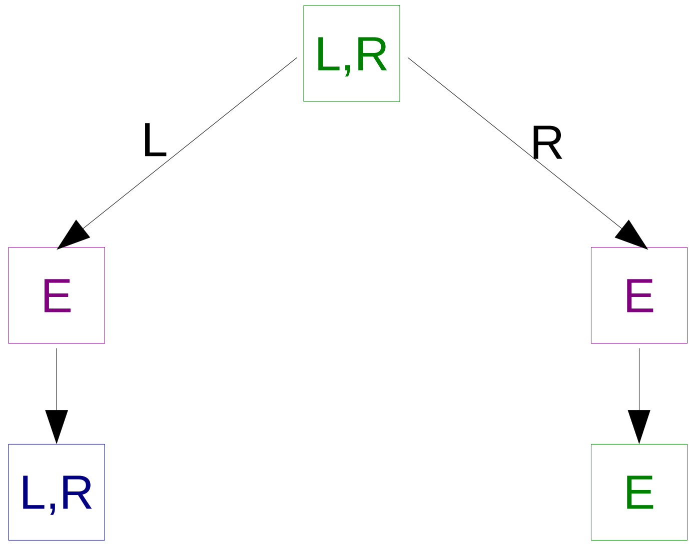
$$(Q_I\, x \cdot (Px \oplus_{III} P'x)) \oplus_{II} Q_I\, x \cdot Q_{II}\, y \cdot R(x,y,y)$$

Concrete Data Structures

$\phi ::= Rx_1.,.x_n \mid \pi(\phi) \mid \phi \oplus_\alpha \phi \mid Q_\alpha\, x \mid (\phi \parallel \phi) \mid \phi \cdot \phi$

Then associate to every formula ϕ a labeled CDS $||\phi||_G$:

$(Q_I\, x \cdot (Px \oplus_{III} P'x)) \oplus_{II}\ Q_I\, x \cdot Q_{II}\, y \cdot R(x,y,y)$



Strategies

M Labeled Concrete Data Structure, D(M) = set of *configurations*

$D(M)^T = D(M) \cup \{T\}$ (T = "no strategy found")

A *strategy* for Player α is a function $\sigma : D(M)^T \rightarrow D(M)^T$

Strategies

M Labeled Concrete Data Structure, D(M) = set of *configurations*

$D(M)^T = D(M) \cup \{T\}$ (T = “no strategy found”)

A *strategy* for Player α is a function $\sigma : D(M)^T \rightarrow D(M)^T$

(S1): $s \subseteq \sigma(s)$;

Strategies

M Labeled Concrete Data Structure, D(M) = set of *configurations*

$D(M)^T = D(M) \cup \{T\}$ (T = "no strategy found")

A *strategy* for Player α is a function $\sigma : D(M)^T \rightarrow D(M)^T$

(S1): $s \subseteq \sigma(s)$;

(S2): $(c, v) \in \sigma(s) - s \Rightarrow \lambda(c) = \alpha$;

Strategies

M Labeled Concrete Data Structure, D(M) = set of *configurations*

$D(M)^T = D(M) \cup \{T\}$ (T = "no strategy found")

A *strategy* for Player α is a function $\sigma : D(M)^T \rightarrow D(M)^T$

(S1): $s \subseteq \sigma(s)$;

(S2): $(c, v) \in \sigma(s) - s \Rightarrow \lambda(c) = \alpha$;

(S3): $\sigma(\sigma(s)) = \sigma(s)$;

Strategies

M Labeled Concrete Data Structure, D(M) = set of *configurations*

$D(M)^T = D(M) \cup \{T\}$ (T = “no strategy found”)

A *strategy* for Player α is a function $\sigma : D(M)^T \rightarrow D(M)^T$

(S1): $s \subseteq \sigma(s)$;

(S2): $(c, v) \in \sigma(s) - s \Rightarrow \lambda(c) = \alpha$;

(S3): $\sigma(\sigma(s)) = \sigma(s)$;

(S4): $s \subseteq t \Rightarrow \sigma(s) \subseteq \sigma(t)$.

Strategies

M Labeled Concrete Data Structure, D(M) = set of *configurations*

$D(M)^T = D(M) \cup \{T\}$ (T = “no strategy found”)

A *strategy* for Player α is a function $\sigma : D(M)^T \rightarrow D(M)^T$

(S1): $s \subseteq \sigma(s)$;

(S2): $(c, v) \in \sigma(s) - s \Rightarrow \lambda(c) = \alpha$;

(S3): $\sigma(\sigma(s)) = \sigma(s)$;

(S4): $s \subseteq t \Rightarrow \sigma(s) \subseteq \sigma(t)$.

$Cl_\alpha(M) = \{s : s$ satisfies S1 … S4$\}$

Strategies

We must now find a way to restrict the information available to the strategies:

Strategies

We must now find a way to restrict the information available to the strategies:

$$S_\alpha(M) \subseteq Cl_\alpha(M)$$

Strategies

We must now find a way to restrict the information available to the strategies:

$$S_\alpha(M) \subseteq Cl_\alpha(M)$$

Global approach: directly define $S_\alpha(M)$

Strategies

We must now find a way to restrict the information available to the strategies:

$$S_\alpha(M) \subseteq Cl_\alpha(M)$$

Global approach: directly define $S_\alpha(M)$

Local approach: add structure to the game, make this explicit

Strategies

We must now find a way to restrict the information available to the strategies:

$$S_\alpha(M) \subseteq Cl_\alpha(M)$$

Global approach: directly define $S_\alpha(M)$

Local approach: add structure to the game, make this explicit

Strategies

$$\gamma_M : C_M \rightarrow [D(M) \rightarrow D(M)]$$

Strategies

$$\gamma_M: C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

Strategies

$$\gamma_M: C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

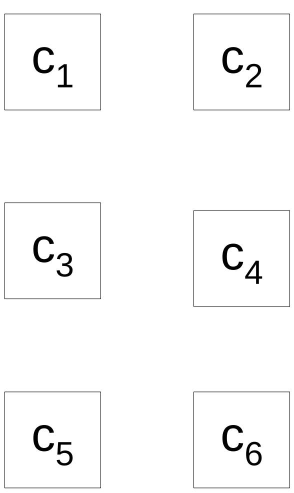
$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M): \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$

Strategies

$$\gamma_M: C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M): \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$

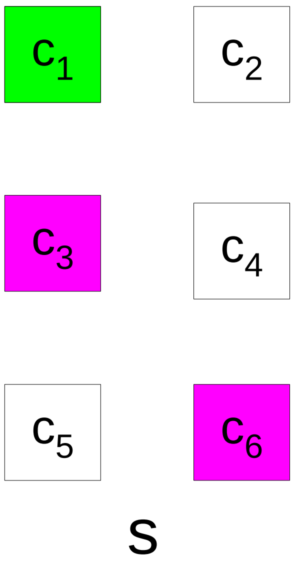


Strategies

$$\gamma_M : C_M \to [D(M) \to D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M) : \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$



Strategies

$$\gamma_M : C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

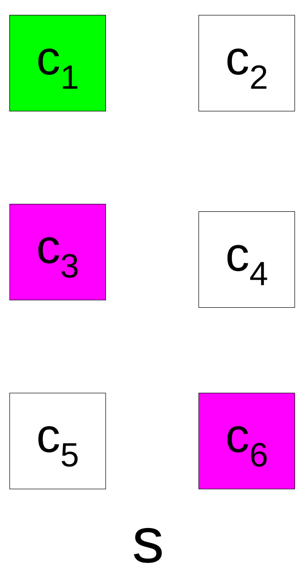
$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M) : \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$

Strategies

$$\gamma_M: C_M \to [D(M) \to D(M)]$$

$$s \searrow c \text{ iff } \exists v \ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M): \forall s \ \forall c, \ \sigma(s)\searrow c \Rightarrow \sigma(\gamma_M(c)(s))\searrow c\}$$

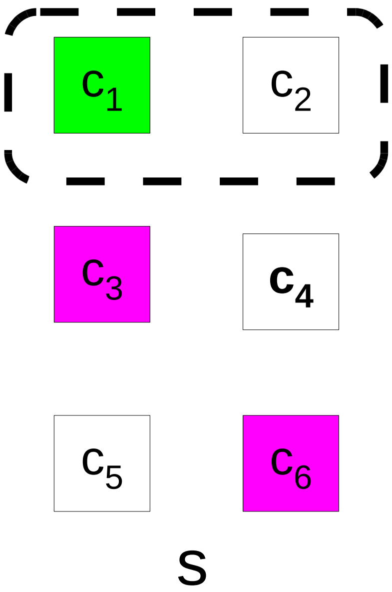


Strategies

$$\gamma_M: C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M): \forall s\ \forall c,\ \sigma(s)\searrow c \Rightarrow \sigma(\gamma_M(c)(s))\searrow c\}$$

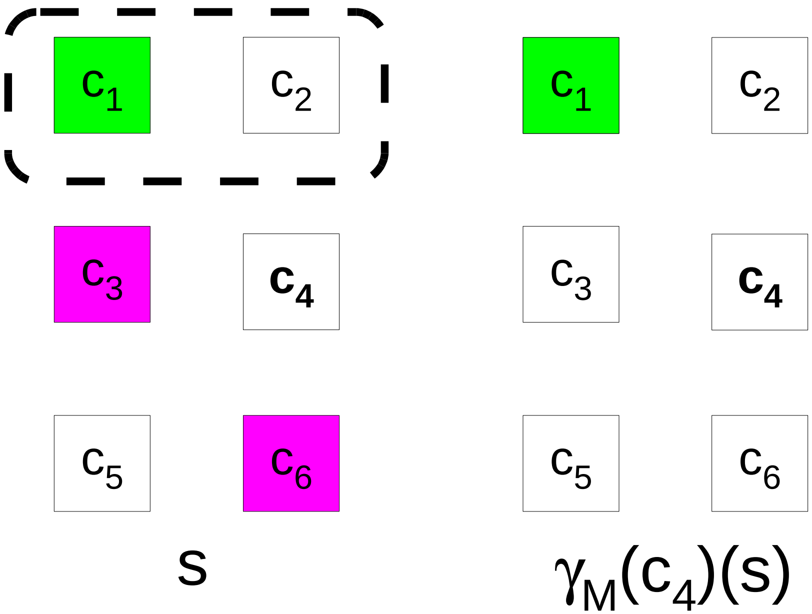


Strategies

$$\gamma_M : C_M \to [D(M) \to D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M) : \forall s\ \forall c,\ \sigma(s)\searrow c \Rightarrow \sigma(\gamma_M(c)(s))\searrow c\}$$

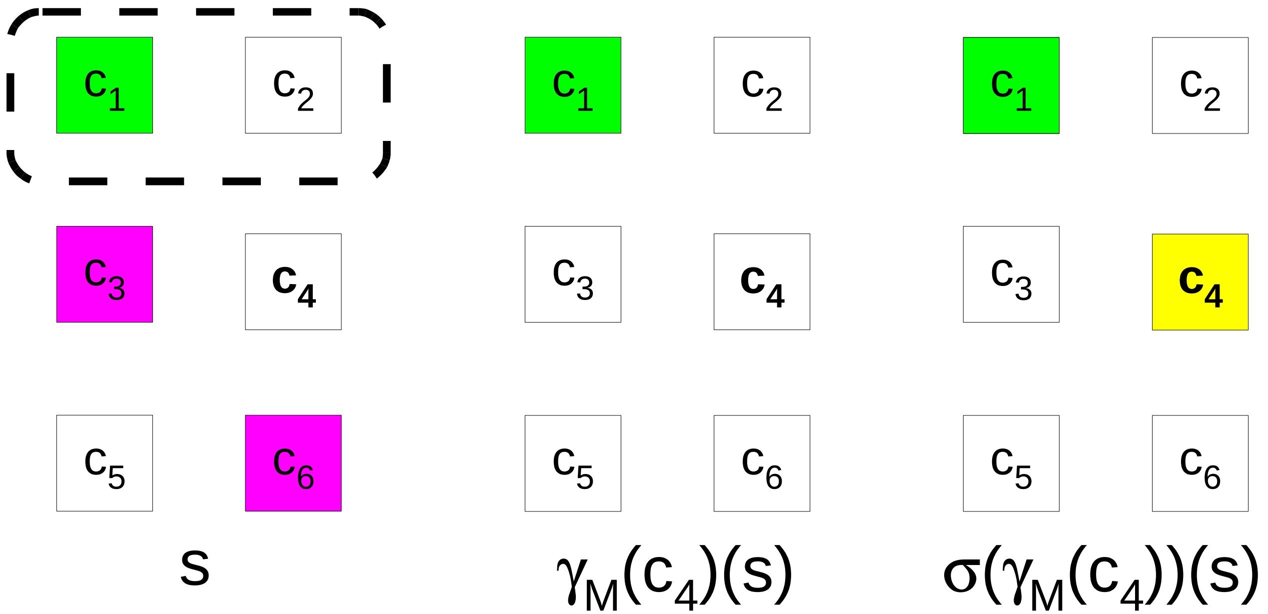


Strategies

$$\gamma_M: C_M \to [D(M) \to D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M): \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$

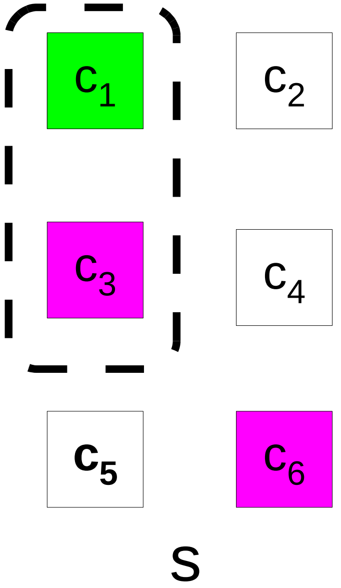


Strategies

$$\gamma_M: C_M \to [D(M) \to D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M): \forall s\ \forall c,\ \sigma(s)\searrow c \Rightarrow \sigma(\gamma_M(c)(s))\searrow c\}$$

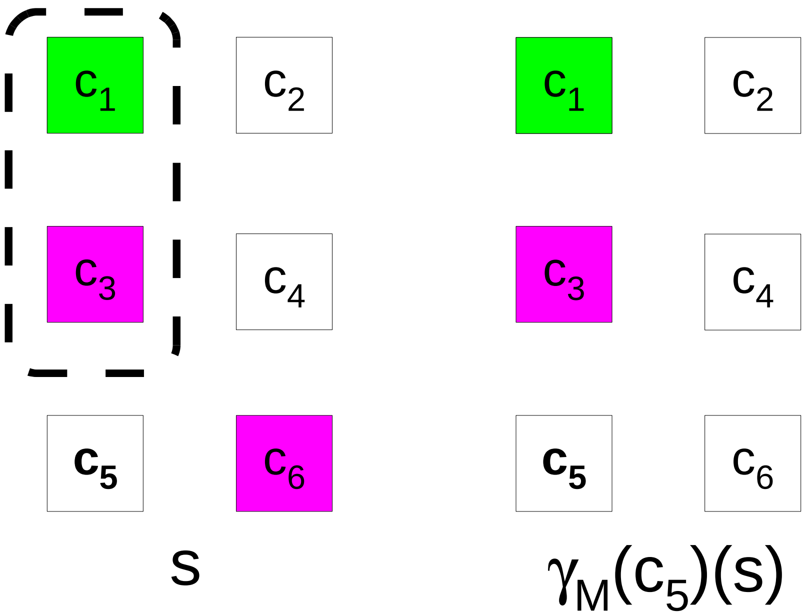


Strategies

$$\gamma_M : C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M) : \forall s\ \forall c,\ \sigma(s)\searrow c \Rightarrow \sigma(\gamma_M(c)(s))\searrow c\}$$

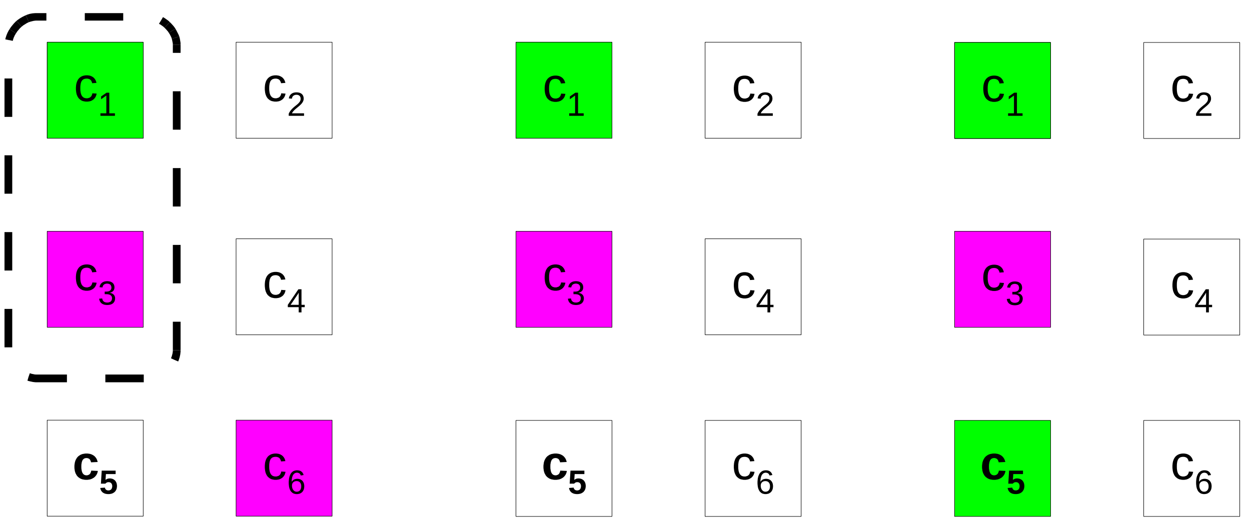


Strategies

$$\gamma_M: C_M \to [D(M) \to D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M): \forall s\ \forall c,\ \sigma(s)\searrow c \Rightarrow \sigma(\gamma_M(c)(s))\searrow c\}$$



s $\quad$ $\gamma_M(c_5)(s)$ $\quad$ $\sigma(\gamma_M(c_5))(s)$

Strategies

$$\gamma_M: C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M): \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$

If ϕ is atomic, $||\phi|| = (\emptyset,\emptyset,\emptyset,\emptyset)$ and $\gamma_{||\phi||} = \emptyset$;

If ϕ is a quantifier, $C_{||\phi||} = \{c_0\}$ and $\gamma_{||\phi||} = \emptyset$;

$\gamma_{||\pi(\phi)||} = \gamma_{||\phi||}$.

Strategies

$$\gamma_M : C_M \rightarrow [D(M) \rightarrow D(M)]$$

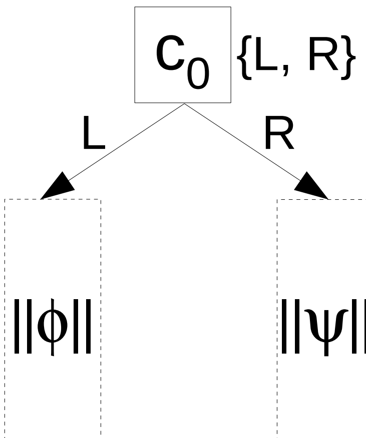
$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M) : \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$

$$\gamma_{\|\phi \oplus_\alpha \psi\|}(c)(s) = \varnothing \quad \text{if } c = c_0;$$

$$= \{(c_0,L)\} \cup \gamma_{\|\phi\|}(c)(s - \{(c_0,L)\}) \quad \text{if } c \text{ in } C_{\|\phi\|};$$

$$= \{(c_0,R)\} \cup \gamma_{\|\psi\|}(c)(s - \{(c_0,T)\}) \quad \text{if } c \text{ in } C_{\|\psi\|};$$



Strategies

$$\gamma_M: C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M): \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$

$$\begin{aligned} \gamma_{\|(\phi \,\|\, \psi)\|}(c)(s) &= \gamma_{\|\phi\|}(c)(\pi_M(s)) \quad \text{if } c \in C_{\|\phi\|}; \\ &= \gamma_{\|\psi\|}(c)(\pi_N(s)) \quad \text{if } c \in C_{\|\psi\|}. \end{aligned}$$

$\|\phi\|$ $\|\psi\|$

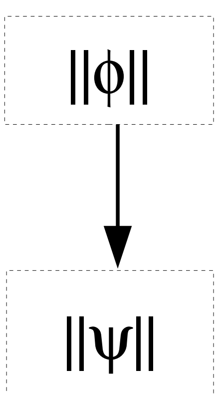
Strategies

$$\gamma_M : C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M) : \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$

$$\begin{aligned} \gamma_{||(\phi \cdot \psi)||}(c)(s) &= \gamma_{||\phi||}(c)(\pi_M(s)) && \text{if } c \in C_{||\phi||}; \\ &= \pi_{||\phi|}(s) \cup \gamma_{||\psi||}(c)(\pi_N(s)) && \text{if } c \in C_{||\psi||}. \end{aligned}$$



Strategies

$$\gamma_M : C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M) : \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$

Weak Progression Assumption (WP): $s \mid\!-_M c,\ s \not\searrow c \Rightarrow \sigma(s) \neq s$

Strategies

$$\gamma_M : C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M) : \forall s\ \forall c,\ \sigma(s) \searrow c \Rightarrow \sigma(\gamma_M(c)(s)) \searrow c\}$$

Weak Progression Assumption (WP): $s \vdash_M c,\ s \not\searrow c \Rightarrow \sigma(s) \neq s$

Strong Progression Assumption (SP): $s \vdash_M c \Rightarrow \sigma(s) \searrow c$

Strategies

$$\gamma_M : C_M \rightarrow [D(M) \rightarrow D(M)]$$

$$s \searrow c \text{ iff } \exists v\ (c,v) \in s$$

$$S_\alpha(M) = \{\sigma \in Cl_\alpha(M) : \forall s\ \forall c,\ \sigma(s)\searrow c \Rightarrow \sigma(\gamma_M(c)(s))\searrow c\}$$

Weak Progression Assumption (WP): $s \mid\!-_M c,\ s \not\searrow c \Rightarrow \sigma(s) \neq s$

Strong Progression Assumption (SP): $s \mid\!-_M c \Rightarrow \sigma(s)\searrow c$

Theorem: WP $\Leftrightarrow$ SP

$\Rightarrow$: obvious.

$\Leftarrow$: if $s \mid\!-_M c,\ \sigma(s)\not\searrow c$, then $\sigma(\sigma(s)) = \sigma(s)\not\searrow c$.

Strategies

Dom(σ) is the smallest set s.t.

$\varnothing \in$ Dom(σ);

Strategies

Dom(σ) is the smallest set s.t.

$\varnothing \in$ Dom(σ);

$s \in$ Dom(σ) $\Rightarrow$ σ(s) $\in$ Dom(σ);

Strategies

Dom(σ) is the smallest set s.t.

$\emptyset \in$ Dom(σ);

s $\in$ Dom(σ) $\Rightarrow$ σ(s) $\in$ Dom(σ);

S $\subseteq$ Dom(σ), S directed $\Rightarrow$ $\cup$S $\in$ Dom(σ);

Strategies

Dom(σ) is the smallest set s.t.

$\varnothing \in$ Dom(σ);

s $\in$ Dom(σ) $\Rightarrow$ σ(s) $\in$ Dom(σ);

S $\subseteq$ Dom(σ), S directed $\Rightarrow$ $\cup$S $\in$ Dom(σ);

s $\in$ Dom(σ), s $\subseteq$ t, [(c,b) $\in$ t-s $\Rightarrow$ λ_M(c) $\neq$ a] $\Rightarrow$ t $\in$ Dom(s).

Strategies

Dom(σ) is the smallest set s.t.

$\emptyset \in$ Dom(σ);

s $\in$ Dom(σ) $\Rightarrow$ σ(s) $\in$ Dom(σ);

S $\subseteq$ Dom(σ), S directed $\Rightarrow$ $\cup$S $\in$ Dom(σ);

s $\in$ Dom(σ), s $\subseteq$ t, [(c,b) $\in$ t-s $\Rightarrow$ λ_M(c) $\neq$ a] $\Rightarrow$ t $\in$ Dom(σ).

S is *safe* iff T $\notin$ Dom(s)

Strategies

$S_\alpha^{lsp} = \{s \in S_\alpha : \text{s satisfies WP, s is safe}\}$

Strategies

$$S_{\alpha}^{lsp} = \{s \in S_{\alpha} : s \text{ satisfies WP}, s \text{ is safe}\}$$

Given a *strategy profile*

Strategies

$S_\alpha^{lsp} = \{s \in S_\alpha : \text{s satisfies WP, s is safe}\}$

Given a *strategy profile*

$$(\sigma_\alpha)_{\alpha \in A} \in \Pi_{\alpha \in A} S_\alpha$$

Strategies

$S_\alpha^{lsp} = \{s \in S_\alpha : \text{s satisfies WP, s is safe}\}$

Given a *strategy profile*

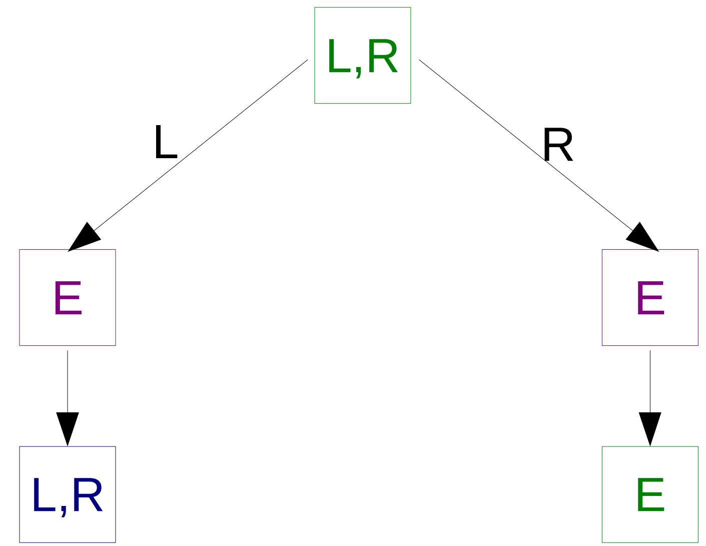
$$(\sigma_\alpha)_{\alpha \in A} \in \Pi_{\alpha \in A} S_\alpha$$

Define

$$< \sigma_\alpha >_{\alpha \in A} = \min \{ s \in D(M)^T : \sigma_\alpha(s) = s \text{ for all } \alpha\}$$

Strategies

$(Q_{I}\, x \cdot (Px \oplus_{III} Sx)) \oplus_{II}\; Q_{I}\, x \cdot Q_{II}\, y \cdot T(x,y,y)$



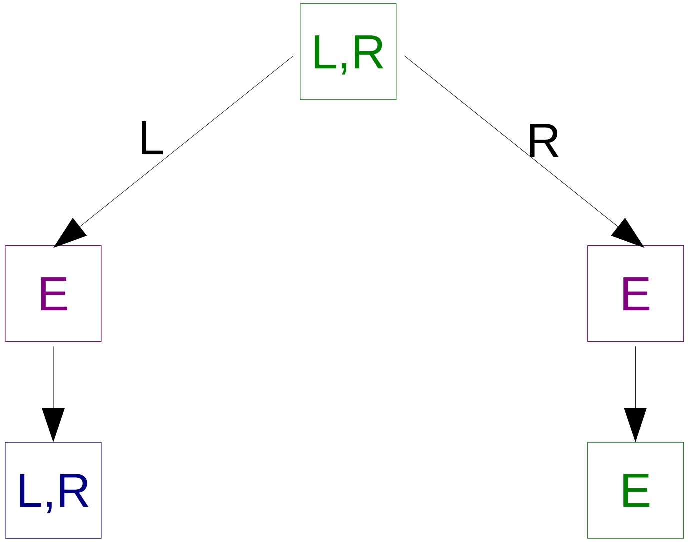
Strategies

$$(Q_{I}\, x \cdot (Px \oplus_{III} Sx)) \oplus_{II} Q_{I}\, x \cdot Q_{II}\, y \cdot T(x,y,y)$$

s_I:
{(II, L)} -> {(II, L), (I, a)};
{(II, R)} -> {(II, L), (I, b)};
s -> s otherwise.

s_{II}:
∅ -> {(II, R)};
{(II, R), (I,a)} -> {(II, R), (I, a), (II, b)};
{(II, R), (I,b)} -> {(II, R), (I, b), (II, a)};
s -> s otherwise.

s_{III}:
{(II, L), (I,a)} -> {(II, L), (I, a), (III, L)};
{(II, L), (I,b)} -> {(II, L), (I, b), (III, R)};
s -> s otherwise.



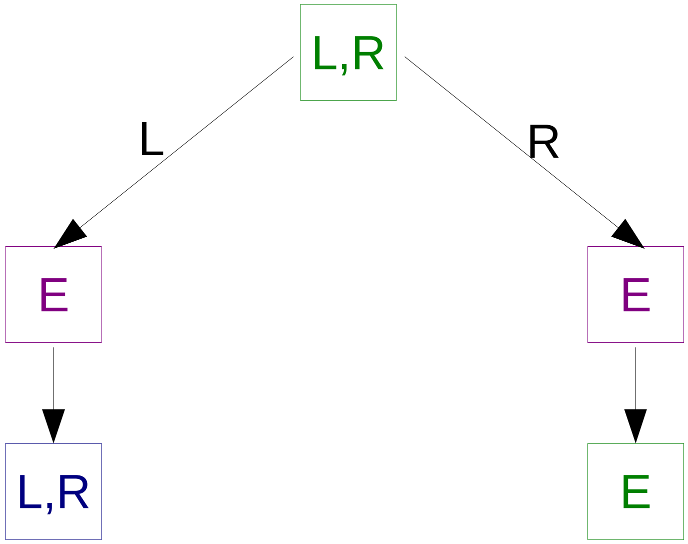
Strategies

$$(Q_I\, x \cdot (Px \oplus_{III} Sx)) \oplus_{II} Q_I\, x \cdot Q_{II}\, y \cdot T(x,y,y)$$

s_I:
{(II, L)} -> {(II, L), (I, a)};
{(II, R)} -> {(II, L), (I, b)};
s -> s otherwise.

s_{II}:
∅ -> {(II, R)};
{(II, R), (I,a)} -> {(II, R), (I, a), (II, b)};
{(II, R), (I,b)} -> {(II, R), (I, b), (II, a)};
s -> s otherwise.

s_{III}:
{(II, L), (I,a)} -> {(II, L), (I, a), (III, L)};
{(II, L), (I,b)} -> {(II, L), (I, b), (III, R)};
s -> s otherwise.

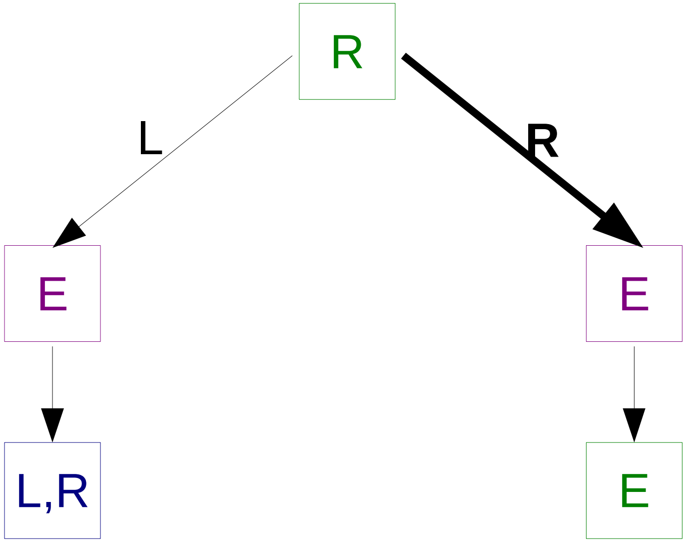


∅

Strategies

$$(Q_{I}\, x \cdot (Px \oplus_{III} Sx)) \oplus_{II}\; Q_{I}\, x \cdot Q_{II}\, y \cdot T(x,y,y)$$

s_{I}: {(II, L)} -> {(II, L), (I, a)};
{(II, R)} -> {(II, L), (I, b)};
s -> s otherwise.



$\mathbf{s_{II}}$: **∅ -> {(II, R)};**
{(II, R), (I,a)} -> {(II, R), (I, a), (II, b)};
{(II, R), (I,b)} -> {(II, R), (I, b), (II, a)};
s -> s otherwise.

{(II, R)}

s_{III}: {(II, L), (I,a)} -> {(II, L), (I, a), (III, L)};
{(II, L), (I,b)} -> {(II, L), (I, b), (III, R)};
s -> s otherwise.

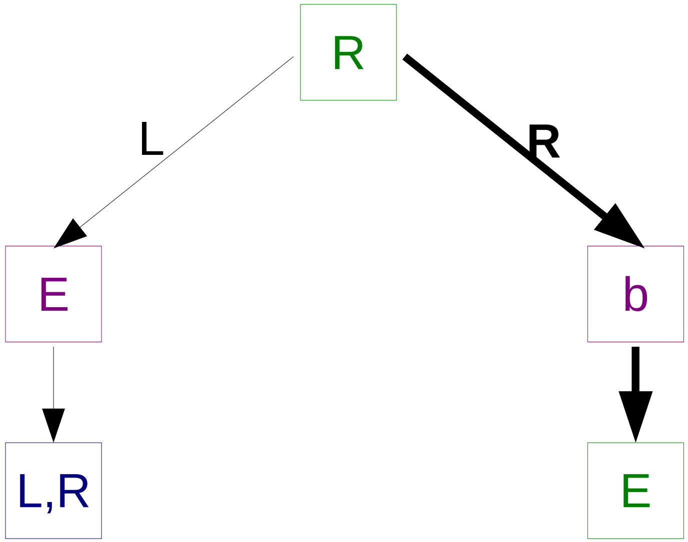
Strategies

$(Q_{I}\, x \cdot (Px \oplus_{III} Sx)) \oplus_{II}\; Q_{I}\, x \cdot Q_{II}\, y \cdot T(x,y,y)$

s_I: {(II, L)} -> {(II, L), (I, a)};
{(II, R)} -> {(II, L), (I, b)};
s -> s otherwise.

s_{II}: ∅ -> {(II, R)};
{(II, R), (I,a)} -> {(II, R), (I, a), (II, b)};
{(II, R), (I,b)} -> {(II, R), (I, b), (II, a)};
s -> s otherwise.

s_{III}: {(II, L), (I,a)} -> {(II, L), (I, a), (III, L)};
{(II, L), (I,b)} -> {(II, L), (I, b), (III, R)};
s -> s otherwise.

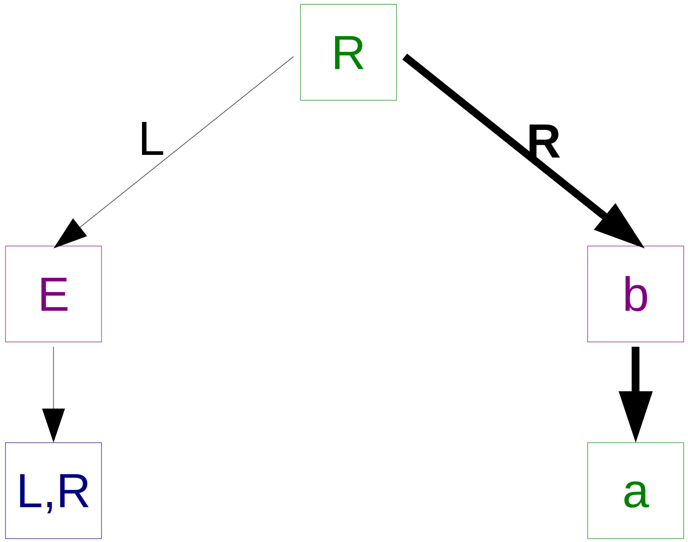


{(II, R), (I, b)}

Strategies

$$(Q_I\, x \cdot (Px \oplus_{III} Sx)) \oplus_{II}\ Q_I\, x \cdot Q_{II}\, y \cdot T(x,y,y)$$

s_I:
{(II, L)} -> {(II, L), (I, a)};
{(II, R)} -> {(II, L), (I, b)};
s -> s otherwise.



s_{II}:
∅ -> {(II, R)};
{(II, R), (I,a)} -> {(II, R), (I, a), (II, b)};
{(II, R), (I,b)} -> {(II, R), (I, b), (II, a)};
s -> s otherwise.

s_{III}:
{(II, L), (I,a)} -> {(II, L), (I, a), (III, L)};
{(II, L), (I,b)} -> {(II, L), (I, b), (III, R)};
s -> s otherwise.

{(II, R), (I, b), (II, a)}

Outcomes and Valutations

So far, no payoffs, and variable assignments.

Outcomes and Valutations

So far, no payoffs, and variable assignments.

ψ: Ar(ψ) -> Co(ψ)

Ar(ψ) = variables that must be bound for ψ to be interpreted
Co(ψ) = variables bound after ψ has been interpreted
(Variables not in Ar(ψ) $\cup$ Co(ψ) "pass through")

Outcomes and Valutations

So far, no payoffs, and variable assignments.

ψ: Ar(ψ) -> Co(ψ)

Ar(ψ) = variables that must be bound for ψ to be interpreted
Co(ψ) = variables bound after ψ has been interpreted
(Variables not in Ar(ψ) $\cup$ Co(ψ) "pass through")

If $\phi(x_1 ... x_n)$ atomic, ϕ : $\{x_1 ... x_n\}$ -> $\emptyset$;

Outcomes and Valutations

So far, no payoffs, and variable assignments.

ψ: Ar(ψ) -> Co(ψ)

Ar(ψ) = variables that must be bound for ψ to be interpreted
Co(ψ) = variables bound after ψ has been interpreted
(Variables not in Ar(ψ) $\cup$ Co(ψ) "pass through")

If $\phi(x_1...x_n)$ atomic, ϕ : $\{x_1...x_n\}$ -> $\emptyset$;

$Q_\alpha x$: $\emptyset$ -> {x};

Outcomes and Valutations

So far, no payoffs, and variable assignments.

ψ: Ar(ψ) -> Co(ψ)

Ar(ψ) = variables that must be bound for ψ to be interpreted
Co(ψ) = variables bound after ψ has been interpreted
(Variables not in Ar(ψ) $\cup$ Co(ψ) "pass through")

If $\phi(x_1 \ldots x_n)$ atomic, $\phi : \{x_1 \ldots x_n\} \to \emptyset$;

$Q_\alpha x: \emptyset \to \{x\}$;

$$\frac{\phi: X \to Y}{\pi(\phi): X \to Y};$$

Outcomes and Valutations

So far, no payoffs, and variable assignments.

ψ: Ar(ψ) -> Co(ψ)

Ar(ψ) = variables that must be bound for ψ to be interpreted
Co(ψ) = variables bound after ψ has been interpreted
(Variables not in Ar(ψ) $\cup$ Co(ψ) "pass through")

If $\phi(x_1...x_n)$ atomic, $\phi : \{x_1...x_n\} \rightarrow \emptyset$;

$Q_\alpha x: \emptyset \rightarrow \{x\}$;

$$\frac{\phi: X \rightarrow Y}{\pi(\phi): X \rightarrow Y} ;$$

$$\frac{\phi: X \rightarrow Y}{\phi: X \cup Z \rightarrow Y \cup W} \; (W \subseteq Z);$$

Outcomes and Valutations

$$\frac{\phi: X \rightarrow Y \qquad \psi: Y \rightarrow Z}{\phi \cdot \psi: X \rightarrow Z};$$

Outcomes and Valutations

$$\frac{\phi: X \to Y \qquad \psi: Y \to Z}{\phi \cdot \psi: X \to Z}\;;$$

$$\frac{\phi: X \to Y \qquad \psi: X \to Y}{\phi \oplus_\alpha \psi: X \to Y}\;;$$

Outcomes and Valutations

$$\frac{\phi: X \text{ -> } Y \qquad \psi: Y \text{->} Z}{\phi \cdot \psi: X \text{ -> } Z};$$

$$\frac{\phi: X \text{ -> } Y \qquad \psi: X \text{ -> } Y}{\phi \oplus_{\alpha} \psi: X \text{ -> } Y};$$

$$\frac{\phi: X_1 \text{ -> } Y_1 \qquad \psi: X_2 \text{ -> } Y_2}{\phi \,\|\, \psi: X_1 \cup^d X_2 \text{ -> } Y_1 \cup^d Y_2}.$$

Outcomes and Valutations

$$\frac{\phi: X \to Y \qquad \psi: Y \to Z}{\phi \cdot \psi: X \to Z};$$

$$\frac{\phi: X \to Y \qquad \psi: X \to Y}{\phi \oplus_\alpha \psi: X \to Y};$$

$$\frac{\phi: X_1 \to Y_1 \qquad \psi: X_2 \to Y_2}{\phi \parallel \psi: X_1 \cup^d X_2 \to Y_1 \cup^d Y_2}.$$

ϕ is a *formula* iff $\phi: X \to \emptyset$

ϕ is a *sentence* iff $\phi: \emptyset \to \emptyset$

Outcomes and Valutations

$$Q_I x \cdot (Q_{II} y \cdot (Px \oplus_{II} Rxy) \parallel Rxx)$$

∅ -> {x}   ∅ -> {y}   {x} ->∅   {x,y} ->∅   {x} ->∅

Outcomes and Valutations

$$Q_I x \cdot (Q_{II} y \cdot (Px \oplus_{II} Rxy) \,||\, Rxx)$$

$\emptyset \to \{x\}$ $\emptyset \to \{y\}$ $\frac{\{x\} \to \emptyset}{\{x,y\} \to \emptyset}$ $\{x,y\} \to \emptyset$ $\{x\} \to \emptyset$

Outcomes and Valutations

$$Q_{I}x \cdot (Q_{II}y \cdot (Px \oplus_{II} Rxy) \,||\, Rxx)$$

∅ -> {x} ∅ -> {y} {x} ->∅ {x,y} ->∅ {x} ->∅

{x,y} ->∅

{x,y} ->∅

Outcomes and Valutations

$$Q_I x \cdot (Q_{II} y \cdot (Px \oplus_{II} Rxy) \,||\, Rxx)$$

∅ -> {x} ∅ -> {y} {x} ->∅ {x,y} ->∅ {x} ->∅

{x} -> {x,y} {x,y} ->∅

{x,y} ->∅

Outcomes and Valutations

$$Q_I x \cdot (Q_{II} y \cdot (Px \oplus_{II} Rxy) \,||\, Rxx)$$

∅ -> {x}   ∅ -> {y}   {x} ->∅   {x,y} ->∅   {x} ->∅

{x} -> {x,y}   {x,y} ->∅

{x,y} ->∅

{x} -> ∅

Outcomes and Valutations

$$Q_I x \cdot (Q_{II} y \cdot (Px \oplus_{II} Rxy) \,||\, Rxx)$$

∅ -> {x} ∅ -> {y} {x} ->∅ {x,y} ->∅ {x} ->∅

{x} -> {x,y} {x,y} ->∅

{x,y} ->∅

{x} -> ∅

{x} -> ∅

Outcomes and Valutations

$$Q_{I}x \cdot (Q_{II}y \cdot (Px \oplus_{II} Rxy) \parallel Rxx)$$

∅ -> {x} ∅ -> {y} {x} ->∅ {x,y} ->∅ {x} ->∅

{x} -> {x,y} {x,y} ->∅

{x,y} ->∅

{x} -> ∅

{x} -> ∅

∅ -> ∅

Outcomes and Valutations

$$(Q_{I}x \ \oplus_{III} Q_{II}x) \cdot Q_{III}y \ ) \cdot Rxy$$

∅ -> {x}	∅ -> {x}	∅ -> {y}	{x,y} ->∅

Outcomes and Valutations

$$(Q_I x \ \oplus_{III} Q_{II} x) \cdot Q_{III} y \ ) \cdot Rxy$$

$\emptyset$ -> {x} $\emptyset$ -> {x} $\emptyset$ -> {y} {x,y} ->$\emptyset$

$\emptyset$ -> {x}

Outcomes and Valutations

$$(Q_{I}x \ \oplus_{III} Q_{II}x) \cdot Q_{III}y \ ) \cdot Rxy$$

$$\frac{\emptyset \text{ -> } \{x\} \qquad\qquad \emptyset \text{ -> } \{x\}}{\emptyset \text{ -> } \{x\}} \qquad \frac{\emptyset \text{ -> } \{y\}}{\{x\} \text{ -> } \{x,y\}} \qquad \{x,y\} \text{ ->} \emptyset$$

Outcomes and Valutations

$$(Q_{I}x \;\; \oplus_{III} \; Q_{II}x) \cdot Q_{III}y \;) \cdot Rxy$$

$\varnothing$ -> {x} $\quad$ $\varnothing$ -> {x} $\quad$ $\varnothing$ -> {y} $\quad$ {x,y} ->$\varnothing$

$\varnothing$ -> {x} $\quad$ {x} -> {x,y}

$\varnothing$ -> {x,y}

Outcomes and Valutations

$$(Q_{I}x \;\; \oplus_{III} Q_{II}x) \cdot Q_{III}y\ ) \cdot Rxy$$

∅ -> {x} ∅ -> {x} ∅ -> {y} {x,y} ->∅

∅ -> {x} {x} -> {x,y}

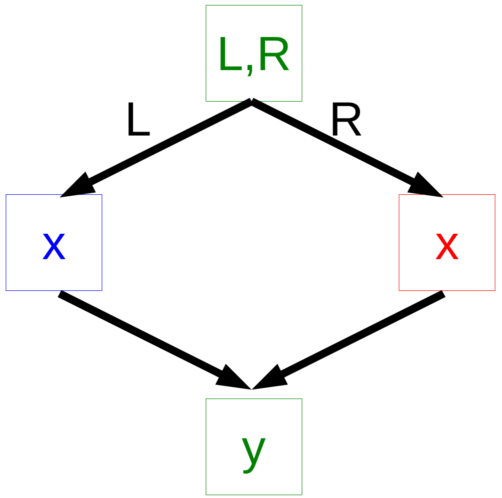
∅ -> {x,y}

∅ -> ∅

Outcomes and Valutations

$$(Q_I x \oplus_{III} Q_{II} x) \cdot Q_{III} y) \cdot Rxy$$

$$\frac{\dfrac{\dfrac{\varnothing \to \{x\} \qquad \varnothing \to \{x\}}{\varnothing \to \{x\}} \qquad \dfrac{\varnothing \to \{y\}}{\{x\} \to \{x,y\}}}{\varnothing \to \{x,y\}} \qquad \{x,y\} \to \varnothing}{\varnothing \to \varnothing}$$



Strategies for open formulas

$$\phi: X \to Y$$

strategy for $\phi = (\sigma_\eta) : \eta \in X^E$

Strategies for open formulas

$$\phi: X \to Y$$

strategy for $\phi = (\sigma_\eta) : \eta \in X^E$

$$<(\sigma_\eta)_\alpha> : X^E \to \mathrm{Max}(\phi)$$

$$<(\sigma_\eta)_\alpha>(\eta_0) = <\sigma_{\eta_0}>$$

Outcomes and Valutations

$$\phi: X \rightarrow Y, s \in Max(||\phi||)$$

$$bind_{||\phi||}(s) : (Y \setminus X) \rightarrow C_M$$

Outcomes and Valutations

$$\phi: X \rightarrow Y, s \in Max(||\phi||)$$

$$bind_{||\phi||}(s) : (Y \setminus X) \rightarrow C_M$$

If ϕ atomic, $Y \setminus X = \varnothing$;

Outcomes and Valutations

$$\phi: X \to Y,\ s \in \mathrm{Max}(||\phi||)$$

$$\mathrm{bind}_{||\phi||}(s) : (Y \setminus X) \to C_M$$

If ϕ atomic, $Y \setminus X = \emptyset$;

$\mathrm{bind}_{||Q_\alpha \phi||}(s)(x) = c_0$;

Outcomes and Valutations

$$\phi: X \rightarrow Y, s \in \mathrm{Max}(||\phi||)$$

$$\mathrm{bind}_{||\phi||}(s) : (Y \setminus X) \rightarrow C_M$$

If ϕ atomic, $Y \setminus X = \varnothing$;

$\mathrm{bind}_{||Q_\alpha\phi||}(s)(x) = c_0$;

$\mathrm{bind}_{||\pi(\phi)||}(s)(x) = \mathrm{bind}_{||\phi||}(s)(x)$;

Outcomes and Valutations

$$\phi, \psi: X \to Y$$

$$\mathrm{bind}_{||\phi \oplus_a \psi||}(s)(y) = \begin{cases} \mathrm{bind}_{||\phi||}(s)(y) \text{ if } (c_0, L) \in s \\ \mathrm{bind}_{||\psi||}(s)(y) \text{ if } (c_0, R) \in s \end{cases}$$

Outcomes and Valutations

$$\phi: X_1 \rightarrow Y_1, \psi: X_2 \rightarrow Y_2$$

$$\text{bind}_{||(\phi \;\; ||\psi\;)||}(s)(y) = \begin{cases} \text{bind}_{||\phi||}(s)(y) \text{ if } y \in Y_1 \setminus (X_1 \cup X_2) \\ \text{bind}_{||\psi||}(s)(y) \text{ if } y \in Y_2 \setminus (X_1 \cup X_2) \end{cases}$$

Outcomes and Valutations

$\phi: X \to Y, \psi: Y \to Z$

$$\text{bind}_{||\phi \cdot \psi||}(s)(y) = \begin{cases} \text{bind}_{||\phi||}(s)(y) \text{ if } y \in Z \cap (Y \setminus X) \\ \text{bind}_{||\psi||}(s)(y) \text{ if } y \in Z \setminus (Y \cup X) \end{cases}$$

Outcomes and Valutations

$$(V, \otimes, \odot, 1)$$

Outcomes and Valutations

$(V, \otimes, \odot, 1)$

$\otimes$, $\odot$ associative;
$\otimes$ commutative;
$x \otimes 1 = 1 \otimes x = x$;
$x \odot 1 = 1 \odot x = x$.

Outcomes and Valutations

$$\phi: X \to Y$$

$$\mathrm{val}_{||\phi||,\alpha} : E^{X} \times \mathrm{Max}(||\phi||) \to V$$

- If ϕ atomic, $\mathrm{val}_{||\phi||,\alpha}(\eta, \varnothing) \in V$ is given;
- $\mathrm{val}_{||Q_{\alpha}\phi||,\alpha}(\eta, s) = 1$;
- $\mathrm{val}_{||\pi(\phi)||,\alpha}(\eta, s) = \mathrm{val}_{||\phi||,\pi^{-1}(\alpha)}(\eta, s)$;

Outcomes and Valutations

$$\phi: X \to Y$$

$$\mathrm{val}_{||\phi||,\alpha} : E^{X} \times \mathrm{Max}(||\phi||) \to V$$

$$\mathrm{val}_{||\phi \oplus_a \psi||,\alpha}(\eta, \{(c_0, L)\} \cup s) = \mathrm{val}_{||\phi||,\alpha}(\eta, s);$$

$$\mathrm{val}_{||\phi \oplus_a \psi||,\alpha}(\eta, \{(c_0, R)\} \cup s) = \mathrm{val}_{||\psi||,\alpha}(\eta, s).$$

Outcomes and Valutations

$$\phi: X_1 \rightarrow Y_1, \psi: X_2 \rightarrow Y_2$$

$$\text{val}_{||(\phi\ ||\psi\ )\ ||,\alpha}(\eta, s) = \text{val}_{||\phi||,\alpha}\,(\pi_{||\phi||}\eta, \pi_{||\phi||}s) \otimes \text{val}_{||\psi||,\alpha}\,(\pi_{||\psi||}\eta, \pi_{||\psi||}s)$$

Outcomes and Valutations

$$\phi: X \to Y, \psi: Y \to Z$$

$$val_{||(\phi \cdot \psi)||,\alpha}(\eta, s) = val_{||\phi||,\alpha}(\pi_{||\phi||}\eta, \pi_{||\phi||}s) \odot val_{||\psi||,\alpha}(\eta', \pi_{||\psi||}s)$$

$$\eta'(y) = \begin{cases} \eta(y), \text{ if } y \in X; \\ s(bind_{||\phi||}(s)(y)), \text{ if } y \in (Y \setminus X). \end{cases}$$

A very trivial example, part II

$$Q_{I}x \cdot Q_{II}y \cdot (x{=}y) : \varnothing \to \varnothing$$

A very trivial example, part II

$$Q_I x \cdot Q_{II} y \cdot (x=y) : \varnothing \to \varnothing$$

$$Q_I x : \varnothing \to \{x\};$$

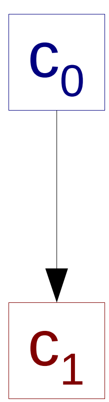
$\mathrm{Bind}_{Q_I x}(s)(x)$ $\boxed{c_0}$ $\mathrm{Val}_{Q_I x}(\eta, s) = 1$

A very trivial example, part II

$$Q_I x \cdot Q_{II} y \cdot (x=y) : \varnothing \to \varnothing$$

$$Q_I x : \varnothing \to \{x\}; \quad Q_{II} y : \{x\} \to \{x,y\};$$

$\mathrm{Bind}_{Q_I x}(s)(x)$ $\quad$ $\mathrm{Val}_{Q_I x}(\eta,s) = 1$



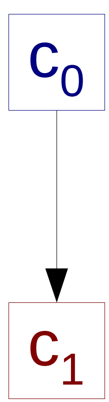
$\mathrm{Bind}_{Q_{II} y}(s)(y)$ $\quad$ $\mathrm{Val}_{Q_{II} y}(\eta,s) = 1$

A very trivial example, part II

$$Q_{I}x \cdot Q_{II}y \cdot (x=y) : \varnothing \rightarrow \varnothing$$

$$Q_{I}x : \varnothing \rightarrow \{x\}; \quad Q_{II}y : \{x\} \rightarrow \{x,y\}; \quad (x=y): \{x,y\} \rightarrow \varnothing.$$

$Bind_{Q_{I}x}(s)(x)$ $Val_{Q_{I}x}(\eta,s) = 1$



$Bind_{Q_{II}y}(s)(y)$ $Val_{Q_{II}y}(\eta,s) = 1$

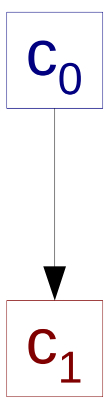
$$Val_{x=y}(\eta, \varnothing) = \begin{cases} 1 \text{ if } \eta(x) = \eta(y); \\ 0 \text{ if } \eta(x) \neq \eta(y). \end{cases}$$

A very trivial example, part II

$$Q_I x \cdot Q_{II} y \cdot (x{=}y) : \varnothing \to \varnothing$$

$$Q_I x : \varnothing \to \{x\};\ \ Q_{II} y : \{x\} \to \{x,y\};\ (x{=}y): \{x,y\} \to \varnothing.$$

$\mathrm{Bind}_{Q_I x}(s)(x)$ $\qquad \mathrm{Val}_{Q_I x}(\eta,s) = 1$



$\mathrm{Bind}_{Q_{II} y}(s)(y)$ $\qquad \mathrm{Val}_{Q_{II} y}(\eta,s) = 1$

$$\mathrm{Val}_{x=y}(\eta, \varnothing) = \begin{cases} 1 \text{ if } \eta(x) = \eta(y); \\ 0 \text{ if } \eta(x) \neq \eta(y). \end{cases}$$

$$\mathrm{Val}_{Q_I x \,\cdot\, Q_{II} y \,\cdot\, x=y}(\varnothing, s) = 1 \odot 1 \odot \mathrm{Val}_{x=y}(\{x = s(c_0), y = s(c_1)\}, \varnothing) =$$

$$= 1 \text{ if } s(c_0) = s(c_1),\ 0 \text{ if } s(c_0) \neq s(c_1)$$

Branching Quantifiers and IF-Logic: interlude

(Henkin, 1961)

$$\begin{pmatrix} \forall x & \exists y \\ \forall z & \exists w \end{pmatrix} \phi(x,y,z,w)$$

Branching Quantifiers and IF-Logic: interlude

(Henkin, 1961)

$$\begin{pmatrix} \forall x & \exists y \\ \forall z & \exists w \end{pmatrix} \phi(x,y,z,w)$$

y may depend on x, but not on z;
w may depend on z, but not on x.

Branching Quantifiers and IF-Logic: interlude

(Henkin, 1961)

$$\begin{pmatrix} \forall x & \exists y \\ \forall z & \exists w \end{pmatrix} \phi(x,y,z,w)$$

y may depend on x, but not on z;
w may depend on z, but not on x.

$$\exists f\ \exists g\ \forall x\ \forall y\ \phi(x,f(x),\ y,\ g(y))$$

Branching Quantifiers and IF-Logic: interlude

(Henkin, 1961)

$$\begin{pmatrix} \forall x & \exists y \\ \forall z & \exists w \end{pmatrix} \phi(x,y,z,w)$$

Not expressible in First Order Logic:

$$\exists a \begin{pmatrix} \forall x & \exists y \\ \forall z & \exists w \end{pmatrix} (x=y \leftrightarrow y=w) \wedge y \neq a$$

Branching Quantifiers and IF-Logic: interlude

In general,

$$\begin{pmatrix} Qx_{1\,1} \ldots & Qx_{1\,m} \\ \ldots\ldots\ldots\ldots\ldots\ldots & \\ Qx_{n\,1} \ldots & Qx_{n\,m} \end{pmatrix} \phi(x,y,z,w)$$

Branching Quantifiers and IF-Logic: interlude

In general,

$$\begin{pmatrix} Qx_{1\,1} \ldots & Qx_{1\,m} \\ \ldots\ldots & \ldots\ldots \\ Qx_{n\,1} \ldots & Qx_{n\,m} \end{pmatrix} \phi(x,y,z,w)$$

FOL + Branching Quantifiers = Existential Second Order

Branching Quantifiers and IF-Logic: interlude

(Hintikka and Sandu, 1989): IF-Logic

$$\begin{pmatrix} \forall x & \exists y \\ \forall z & \exists w \end{pmatrix} \phi(x,y,z,w) \rightarrow \forall x\, \exists y\, \forall z\ \ (\exists w/x)\ \phi(x,y,z,w)$$

Branching Quantifiers and IF-Logic: interlude

(Hintikka and Sandu, 1989): IF-Logic

$$\begin{pmatrix} \forall x & \exists y \\ \forall z & \exists w \end{pmatrix} \phi(x,y,z,w) \rightarrow \forall x\, \exists y\, \forall z\, (\exists w/x)\, \phi(x,y,z,w)$$

$$\forall x\, \exists y\, \forall z\, (\exists w/x,y)\, \phi(x,y,z,w)$$

Branching Quantifiers and IF-Logic: interlude

(Hintikka and Sandu, 1989): IF-Logic

$$\begin{pmatrix}\forall x & \exists y \\ \forall z & \exists w\end{pmatrix}\phi(x,y,z,w) \rightarrow \forall x\, \exists y\, \forall z\ (\exists w/x)\, \phi(x,y,z,w)$$

$$\forall x\, \exists y\, \forall z\ (\exists w/x,y)\, \phi(x,y,z,w)$$

(Hodges, 1997): Compositional Semantics

$X \models \phi$, X set of assignments *(team)*

Branching Quantifiers and IF-Logic: interlude

(Hintikka and Sandu, 1989): IF-Logic

$$\begin{pmatrix} \forall x & \exists y \\ \forall z & \exists w \end{pmatrix} \phi(x,y,z,w) \rightarrow \forall x\, \exists y\, \forall z\, (\exists w/x)\, \phi(x,y,z,w)$$

$$\forall x\, \exists y\, \forall z\, (\exists w/x,y)\, \phi(x,y,z,w)$$

(Hodges, 1997): Compositional Semantics

$X \models \phi$, X set of assignments *(team)*

(Hodges and Cameron, 2001): cannot do better than this.

Branching Quantifiers and IF-Logic: interlude

(Hintikka and Sandu, 1989): IF-Logic

$$\begin{pmatrix} \forall x & \exists y \\ \forall z & \exists w \end{pmatrix} \phi(x,y,z,w) \rightarrow \forall x\, \exists y\, \forall z\, (\exists w/x)\, \phi(x,y,z,w)$$

$$\forall x\, \exists y\, \forall z\, (\exists w/x,y)\, \phi(x,y,z,w)$$

(Hodges, 1997): Compositional Semantics

$X \models \phi$, X set of assignments *(team)*

(Hodges and Cameron, 2001): cannot do better than this.

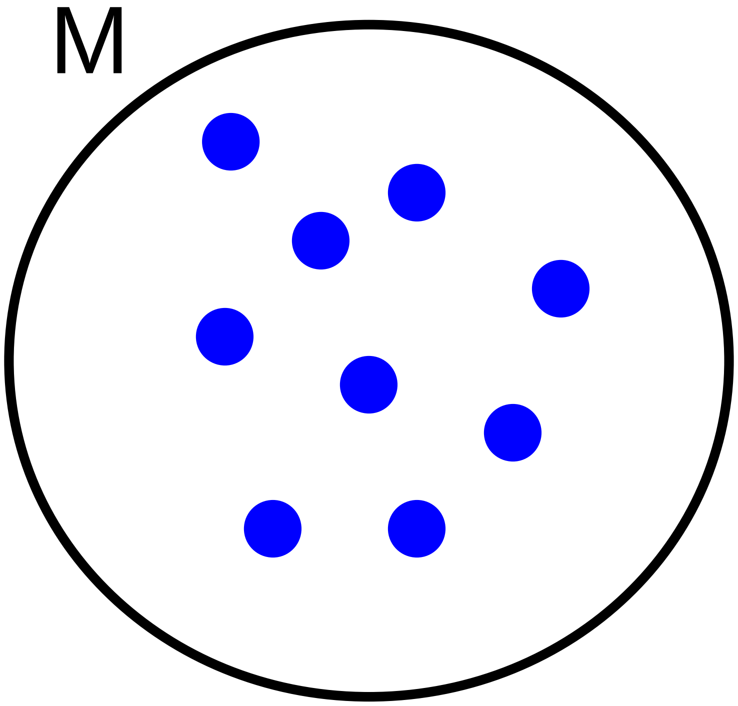
(Väänanen, 2007): $\forall x \exists y \forall z \exists w\, (=(w,z) \wedge \phi(x,y,z,w))$

A very trivial example, part III

$$\phi = \forall x\ (\exists y/x)\ (x=y)$$

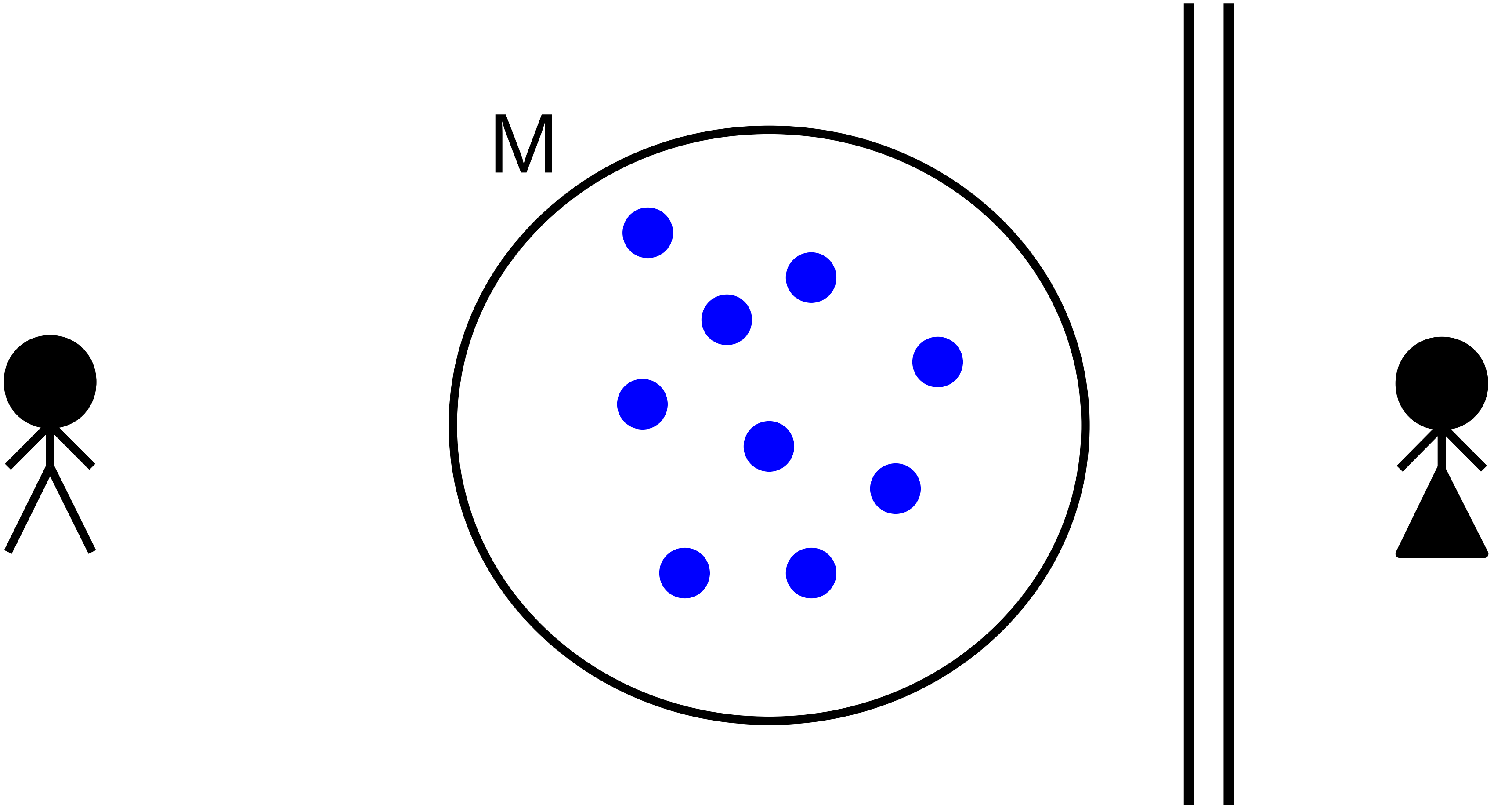
A very trivial example, part III

$$\phi = \forall x\ (\exists y/x)\ (x=y)$$



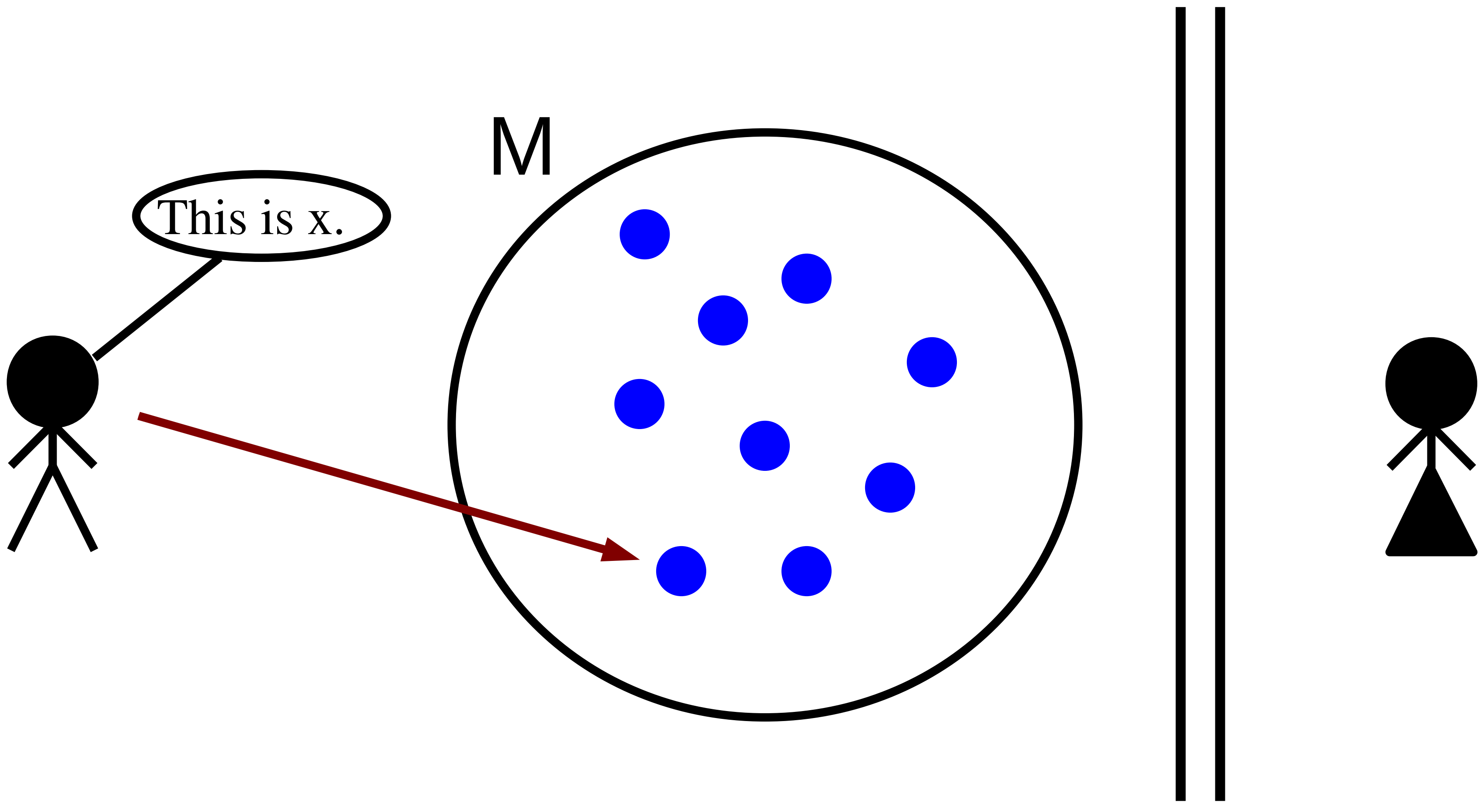
A very trivial example, part III

$$\phi = \forall x\ (\exists y/x)\ (x=y)$$



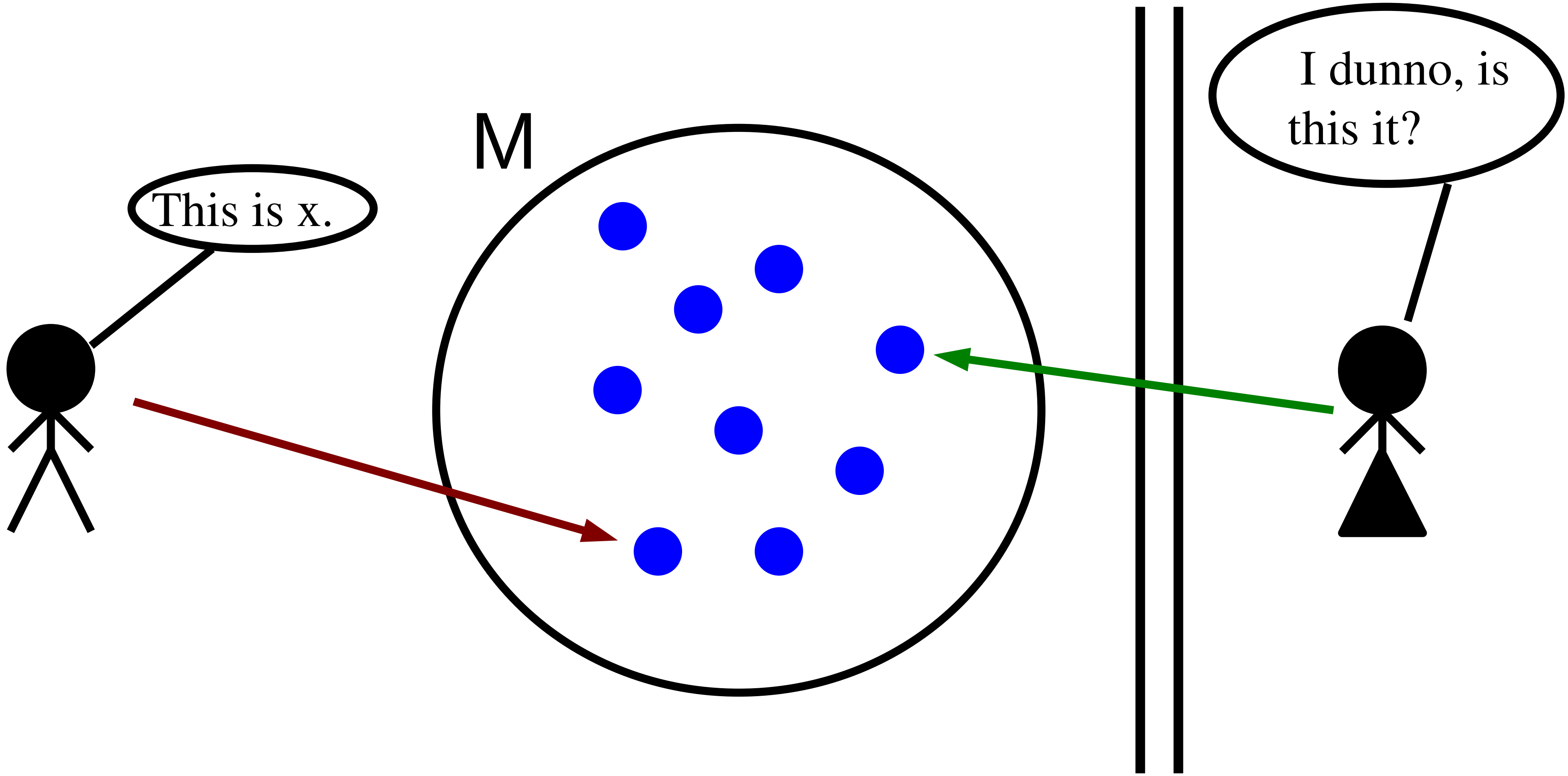
A very trivial example, part III

$$\phi = \forall x\ (\exists y/x)\ (x=y)$$



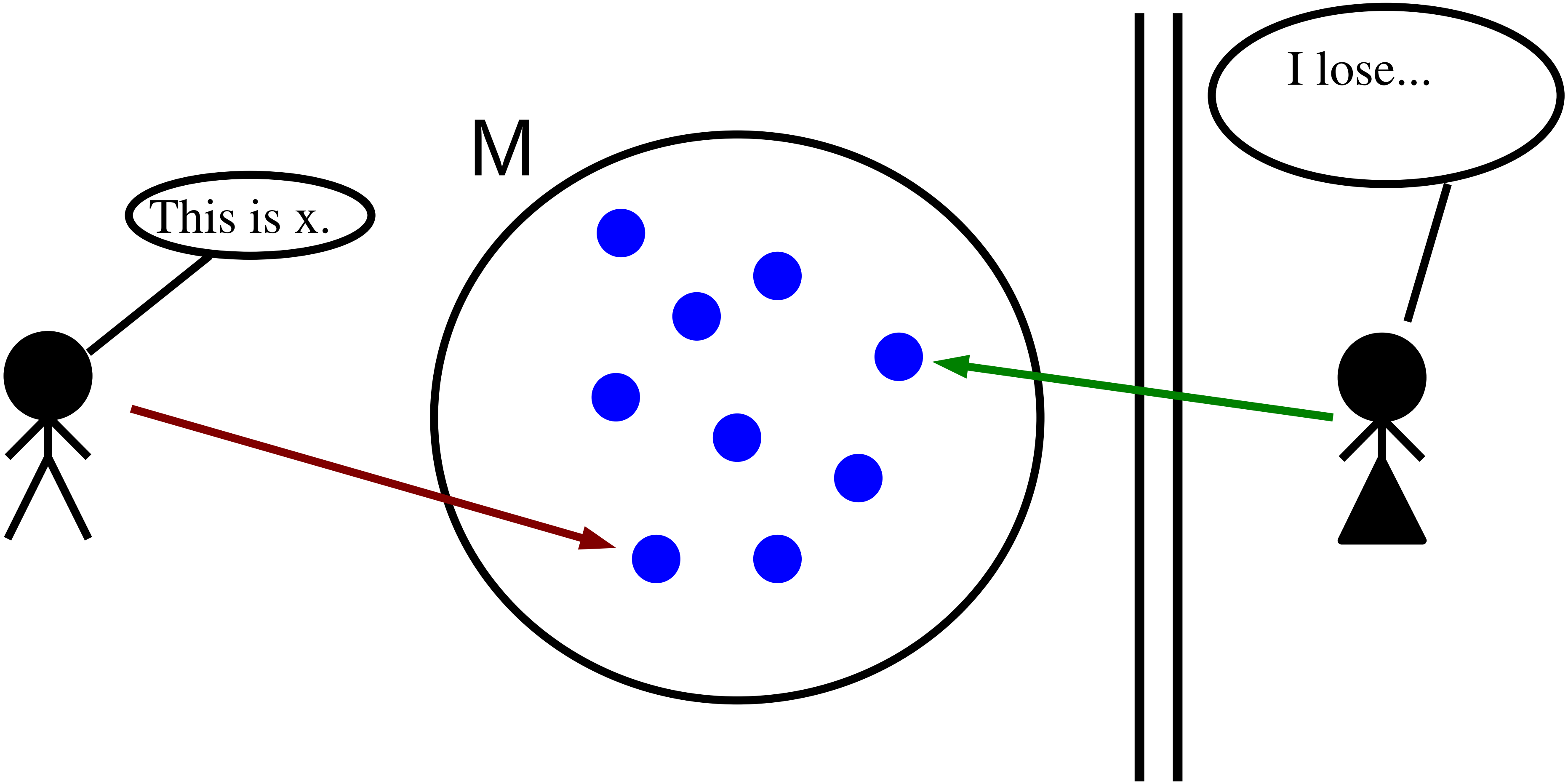
A very trivial example, part III

$$\phi = \forall x\, (\exists y/x)\, (x=y)$$



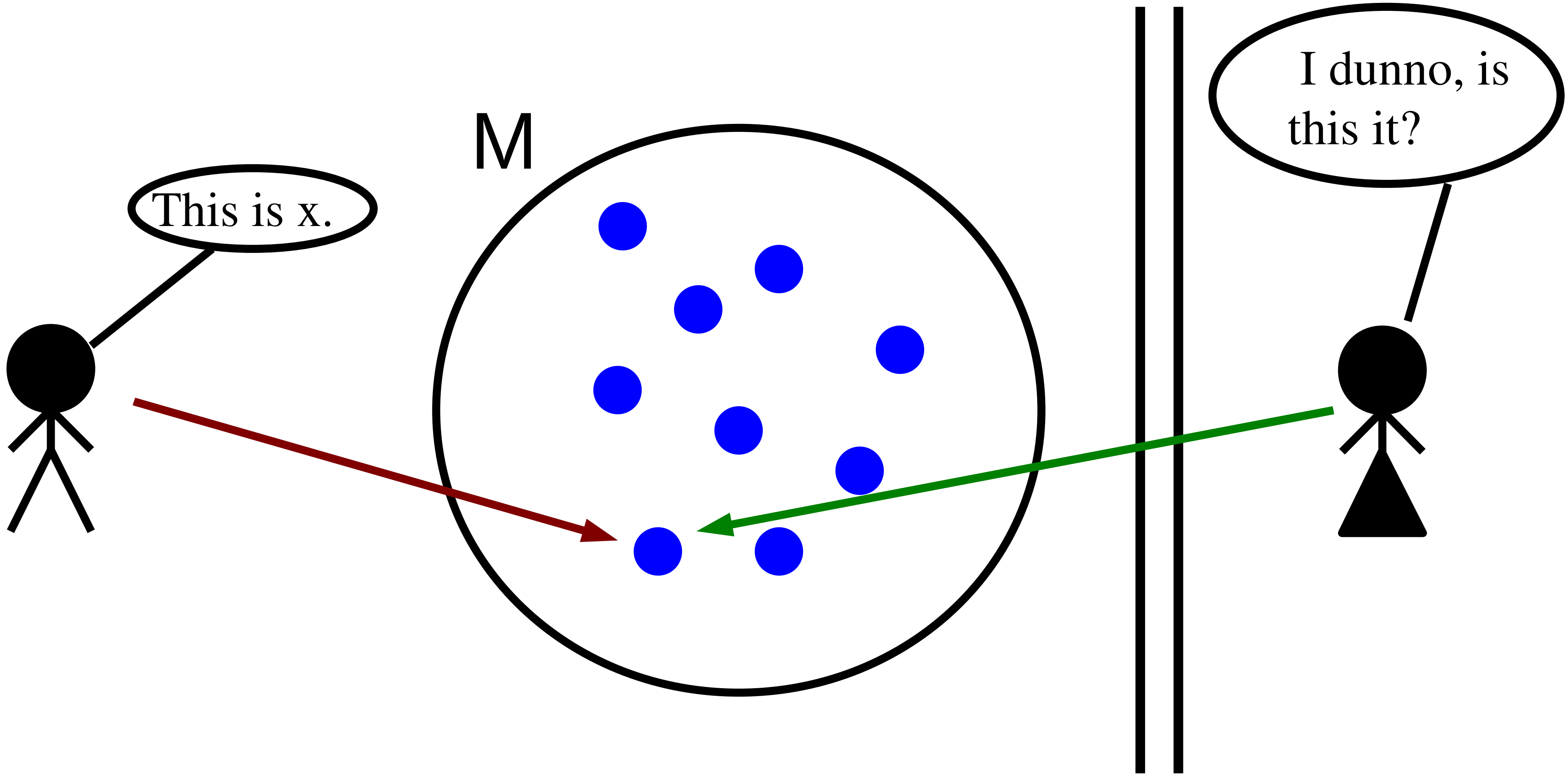
A very trivial example, part III

$\phi = \forall x\ (\exists y/x)\ (x{=}y)$



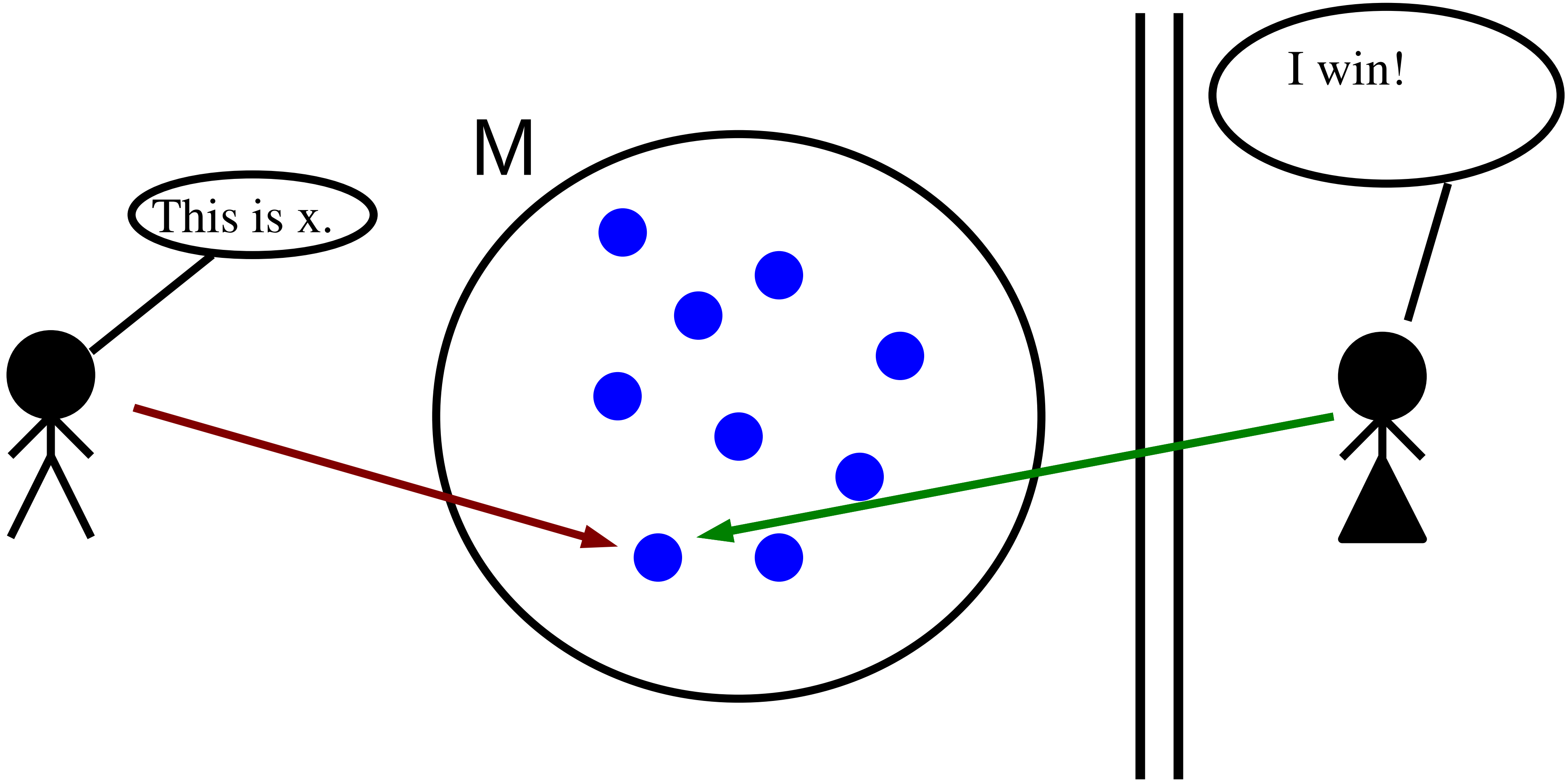
A very trivial example, part III

$$\phi = \forall x\ (\exists y/x)\ (x{=}y)$$



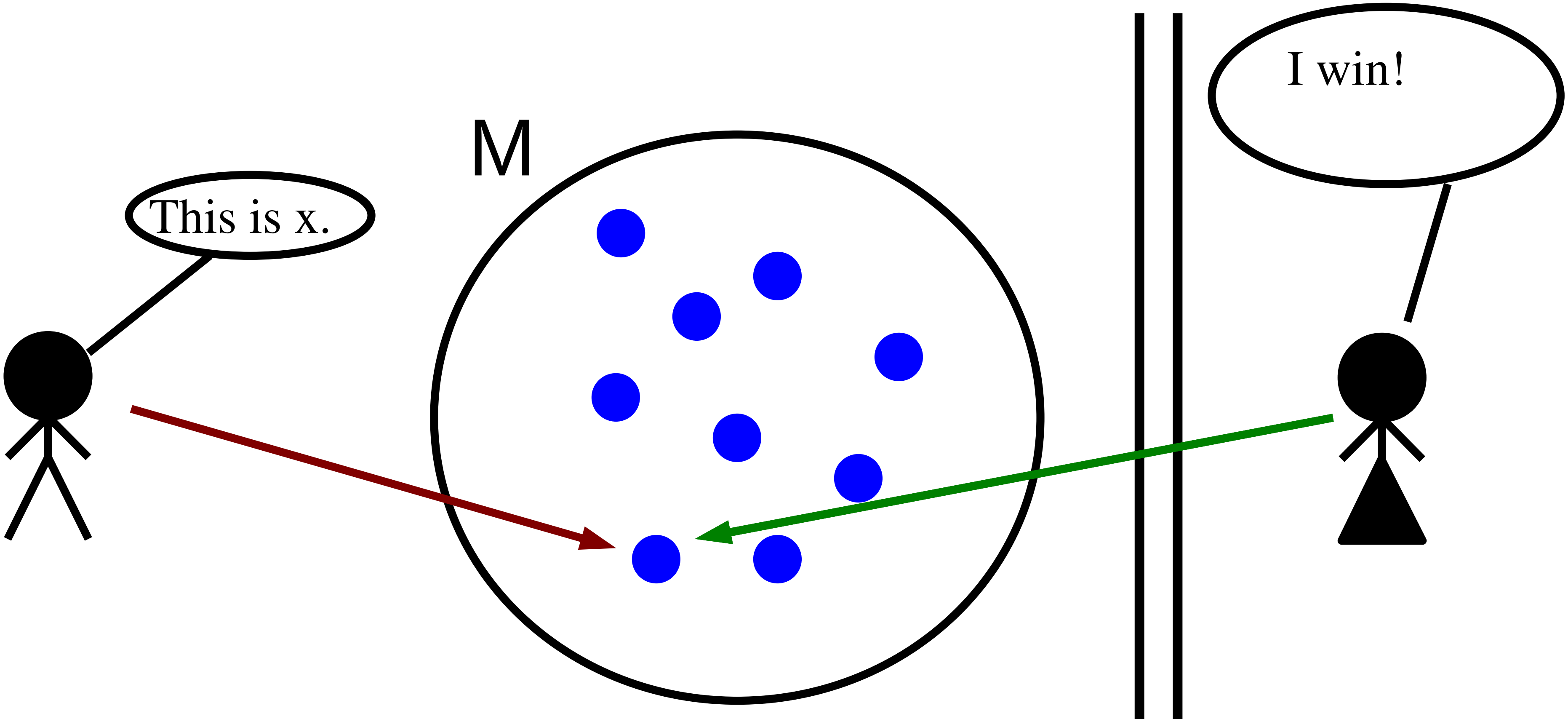
A very trivial example, part III

$$\phi = \forall x\ (\exists y/x)\ (x{=}y)$$



A very trivial example, part III

$$\phi = \forall x\, (\exists y/x)\, (x=y)$$



Neither player has a winning strategy.

Multi-player imperfect information

Branching quantifiers can be expressed using parallel products:

$$\begin{pmatrix} Q_\alpha x \; Q_\beta y \\ Q_\gamma z \; Q_\delta w \end{pmatrix} \phi(x,y,z,w) \equiv (Q_\alpha x \cdot Q_\beta y) || (Q_\gamma x \cdot Q_\delta y) \; \phi(x,y,z,w)$$

Multi-player imperfect information

Branching quantifiers can be expressed using parallel products:

$$\begin{pmatrix} Q_\alpha x & Q_\beta y \\ Q_\gamma z & Q_\delta w \end{pmatrix} \phi(x,y,z,w) \equiv (Q_\alpha x \cdot Q_\beta y) || (Q_\gamma x \cdot Q_\delta y)\ \phi(x,y,z,w)$$

We get imperfect information through the parallel product!

Multi-player imperfect information

Branching quantifiers can be expressed using parallel products:

$$\begin{pmatrix} Q_\alpha x \; Q_\beta y \\ Q_\gamma z \; Q_\delta w \end{pmatrix} \phi(x,y,z,w) \equiv (Q_\alpha x \cdot Q_\beta y) || (Q_\gamma x \cdot Q_\delta y) \; \phi(x,y,z,w)$$

We get imperfect information through the parallel product!

What if we want IF-Quantifiers too?

Multi-player imperfect information

Add to Concrete Data Structure an *Occlusion Function*:

Multi-player imperfect information

Add to Concrete Data Structure an
Occlusion Function:

$$\mathrm{Occ}_{\|\phi\|}: C_{\|\phi\|} \rightarrow \mathcal{P}(X)$$

Multi-player imperfect information

Add to Concrete Data Structure an *Occlusion Function*:

$$\mathrm{Occ}_{||\phi||} : C_{||\phi||} \rightarrow \mathcal{P}(X)$$

- If ϕ atomic, $= C_{||\phi||} = \varnothing$;

Multi-player imperfect information

Add to Concrete Data Structure an *Occlusion Function*:

$$Occ_{||\phi||}: C_{||\phi||} \rightarrow \mathcal{P}(X)$$

- If ϕ atomic, = $C_{||\phi||} = \emptyset$;

- $Occ_{||Q_\alpha x||}(c_0) = \emptyset$;

Multi-player imperfect information

Add to Concrete Data Structure an *Occlusion Function*:

$$\mathrm{Occ}_{\|\phi\|} : C_{\|\phi\|} \rightarrow \mathcal{P}(X)$$

- If ϕ atomic, $= C_{\|\phi\|} = \emptyset$;

- $\mathrm{Occ}_{\|Q_\alpha x\|}(c_0) = \emptyset$; **$\mathrm{Occ}_{\|Q_\alpha x\, /X\|}(c_0) = X$;**

Multi-player imperfect information

Add to Concrete Data Structure an *Occlusion Function*:

$$\mathrm{Occ}_{||\phi||} : C_{||\phi||} \rightarrow \mathcal{P}(X)$$

- If ϕ atomic, $= C_{||\phi||} = \varnothing$;

- $\mathrm{Occ}_{||Q_\alpha x||}(c_0) = \varnothing$; **$\mathrm{Occ}_{||Q_\alpha x\ /X||}(c_0) = X$;**

- $\mathrm{Occ}_{||\pi(\phi)||} = \mathrm{Occ}_{||\phi||}$;

Multi-player imperfect information

Add to Concrete Data Structure an *Occlusion Function*:

$$Occ_{||\phi||}: C_{||\phi||} \rightarrow P(X)$$

- $Occ_{||\phi \otimes_\alpha \psi||}(c) = \begin{cases} Occ_{||\phi||}(c) \text{ if } c \in C_{||\phi||}; \\ Occ_{||\psi||}(c) \text{ if } c \in C_{||\psi||}; \end{cases}$

Multi-player imperfect information

Add to Concrete Data Structure an *Occlusion Function*:

$$Occ_{||\phi||} : C_{||\phi||} \rightarrow \mathcal{P}(X)$$

$$\phi : X_1 \rightarrow Y_1,\ \psi : X_2 \rightarrow Y_2$$

- $Occ_{||(\phi \parallel \psi)||}(c) = \begin{cases} Occ_{||\phi||}(c) \cup X_2 \text{ if } c \in C_{||\phi||}; \\ Occ_{||\psi||}(c) \cup X_1 \text{ if } c \in C_{||\psi||}; \end{cases}$

Multi-player imperfect information

Add to Concrete Data Structure an *Occlusion Function*:

$$\mathrm{Occ}_{||\phi||}: C_{||\phi||} \to \mathcal{P}(X)$$

$$\phi: X \to Y,\ \psi: Y \to Z$$

- $\mathrm{Occ}_{||(\phi \cdot \psi)||}(c) = \begin{cases} \mathrm{Occ}_{||\phi||}(c) & \text{if } c \in C_{||\phi||}; \\ \mathrm{Occ}_{||\psi||}(c) \cup (X \setminus Y) & \text{if } c \in C_{||\psi||}; \end{cases}$

Multi-player imperfect information

Adapt the definition of the visibility condition for sequential composition:

$$\gamma_{\|(\phi \cdot \psi)\|} = \gamma_{\|\phi\| \cdot \|\psi\|}(c)(s) = \begin{cases} \gamma_{\|\phi\|}(c)(\pi_M(s)) & \text{if } c \in C_{\|\phi\|}; \\ \pi_{\|\phi\|}(s) \cup \gamma_{\|\psi\|}(c)(\pi_N(s)) & \text{if } c \in C_{\|\psi\|}. \end{cases}$$

Multi-player imperfect information

Adapt the definition of the visibility condition for sequential composition:

$$\gamma_{\|(\phi \cdot \psi)\|} = \gamma_{\|\phi\| \cdot \|\psi\|}(c)(s) = \begin{cases} \gamma_{\|\phi\|}(c)(\pi_M(s)) & \text{if } c \in C_{\|\phi\|}; \\ (\pi_{\|\phi\|}(s) \setminus \mathbf{S(s)}) \cup \gamma_{\|\psi\|}(c)(\pi_N(s)) & \text{if } c \in C_{\|\psi\|}. \end{cases}$$

$$S(s) = \{(c', v) \in D_{\|\psi\|} \mid \exists\, y \in Occ_{\|\psi\|}(c),\ bind_{\|\phi\|}(s)(y) = c'\}$$

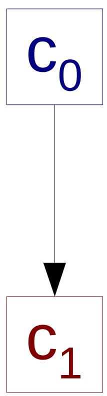
A very trivial example, part IV

$$\phi = Q_{I}x \cdot (Q_{II}y/x) \cdot (x{=}y) : \emptyset \to \emptyset$$

A very trivial example, part IV

$$\phi = Q_{I}x \cdot (Q_{II}y/x) \cdot (x=y) : \varnothing \rightarrow \varnothing$$

$\mathrm{Bind}_{Q_{I}x}(s)(x)$ $\mathrm{Val}_{Q_{I}x}(\eta,s) = 1$



$\mathrm{Bind}_{Q_{II}y}(s)(y)$ $\mathrm{Val}_{Q_{II}y\,/\,x}(\eta,s) = 1$

$\mathbf{Occ}_{Q_{II}y\,/\,x}(\mathbf{c_1}) = \mathbf{c_0}$

A very trivial example, part IV

$$\phi = Q_{I}x \cdot (Q_{II}y/x) \cdot (x{=}y) : \varnothing \rightarrow \varnothing$$

$\mathrm{Bind}_{Q_{I}x}(s)(x)$ c_0 $\mathrm{Val}_{Q_{I}x}(\eta, s) = 1$

$\downarrow$

$\mathrm{Bind}_{Q_{II}y}(s)(y)$ c_1 $\mathrm{Val}_{Q_{II}y\,/\,x}(\eta, s) = 1$

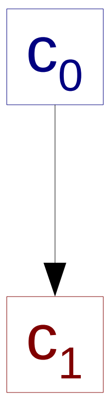
$\mathbf{Occ}_{Q_{II}y\,/\,x}(\mathbf{c_1}) = \mathbf{c_0}$

$$\mathrm{Val}_{\phi}(\varnothing, s) = \begin{cases} 1 \text{ if } s(c_0) = s(c_1); \\ 0 \text{ if } s(c_0) \neq s(c_1). \end{cases}$$

A very trivial example, part IV

$$\phi = Q_{\mathrm{I}}x \cdot (Q_{\mathrm{II}}y/x) \cdot (x=y) : \varnothing \to \varnothing$$

$\mathrm{Bind}_{Q_{\mathrm{I}}x}(s)(x)$



$\mathrm{Val}_{Q_{\mathrm{I}}x}(\eta,s) = 1$

$\gamma_{||\phi||}(c_1)(s) = \pi_{Q_{\mathrm{I}}x}(s) \setminus c_0 \cup \varnothing = \varnothing$

$\mathrm{Bind}_{Q_{\mathrm{II}}y}(s)(y)$

$\mathrm{Val}_{Q_{\mathrm{II}}y\,/\,x}(\eta,s) = 1$

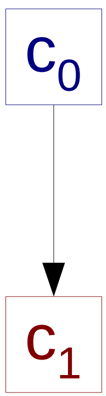
$\mathbf{Occ}_{Q_{\mathrm{II}}y\,/\,x}(\mathbf{c_1}) = \mathbf{c_0}$

$$\mathrm{Val}_{\phi}(\varnothing, s) = \begin{cases} 1 \text{ if } s(c_0) = s(c_1); \\ 0 \text{ if } s(c_0) \neq s(c_1). \end{cases}$$

A very trivial example, part IV

$$\phi = Q_{I}x \cdot (Q_{II}y/x) \cdot (x=y) : \varnothing \to \varnothing$$

$\mathrm{Bind}_{Q_{I}x}(s)(x)$ $\quad \mathrm{Val}_{Q_{I}x}(\eta,s) = 1$



$\mathrm{Bind}_{Q_{II}y}(s)(y)$ $\quad \mathrm{Val}_{Q_{II}y\,/\,x}(\eta,s) = 1$

$\mathbf{Occ}_{Q_{II}y\,/\,x}(c_1) = c_0$

$$\mathrm{Val}_{\phi}(\varnothing, s) = \begin{cases} 1 \text{ if } s(c_0) = s(c_1); \\ 0 \text{ if } s(c_0) \neq s(c_1). \end{cases}$$

$$\gamma_{\|\phi\|}(c_1)(s) = \pi_{Q_{I}x}(s) \setminus c_0 \cup \varnothing = \varnothing$$

$\sigma(\{(c_0,a)\} = (c_1,a)$

$\sigma(\{(c_0,b)\} = (c_1,b)$

is not admissible!

A very trivial example, part IV

$$\phi = Q_{I}x \cdot (Q_{II}y/x) \cdot (x{=}y) : \varnothing \to \varnothing$$

$\mathrm{Bind}_{Q_I x}(s)(x)$ c_0 $\mathrm{Val}_{Q_I x}(\eta, s) = 1$

$\gamma_{||\phi||}(c_1)(s) = \pi_{Q_I x}(s) \setminus c_0 \cup \varnothing = \varnothing$

$\mathrm{Bind}_{Q_{II} y}(s)(y)$ c_1 $\mathrm{Val}_{Q_{II} y\,/\,x}(\eta, s) = 1$

$\mathbf{Occ}_{Q_{II} y\,/\,x}(c_1) = c_0$

$\sigma(\{(c_0, a)\} = (c_1, a)$

$\sigma(\{(c_0, b)\} = (c_1, b)$

is not admissible!

$$\mathrm{Val}_{\phi}(\varnothing, s) = \begin{cases} 1 \text{ if } s(c_0) = s(c_1); \\ 0 \text{ if } s(c_0) \neq s(c_1). \end{cases}$$

The value of y may not depend on the value of x

Branching Quantifiers and IF-Logic: interlude

What we have:

- Very general framework for multiplayer Game Semantics;

- Highly abstract treatment of strategies, bound variables, valutations;

- IF quantifiers.

Branching Quantifiers and IF-Logic: interlude

What would be nice:

- Compositional semantics for winning strategies (generalize Hodges 1997);
- Compositional semantics for *(strong?)* Nash Equilibria (cfr. Galliani 2008) and other equilibrium notions;
- Expressive power and complexity;
- Generalize (van Benthem, 2003);
- Metalogical results

(cfr. Loohuis, Tulenheimo, Venema)