Strategies: A logic - automata study
Lecture 4: Making strategies explicit

Sujata Ghosh and R. Ramanujam

University of Groningen, The Netherlands

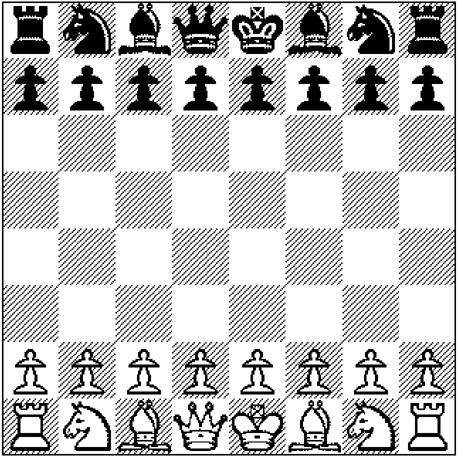
The Institute of Mathematical Sciences, Chennai, India

August 4, 2011

Certain issues

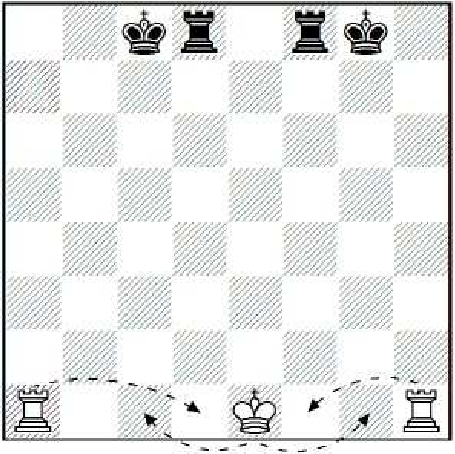
- what can be achieved
- existential strategies hiding in the semantics
- how can that be achieved
- strategies explicitly in the logical language
- logics with atomic strategies (names for strategies)
- logics with more structure in strategies

Playing chess



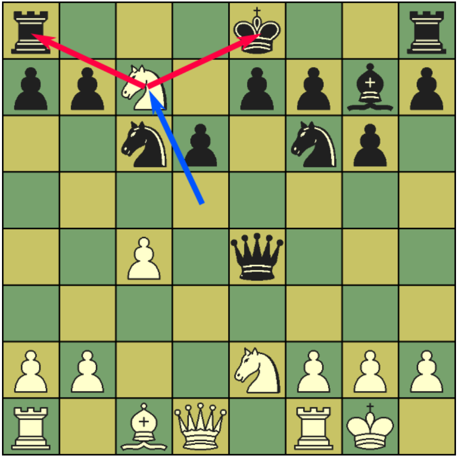
Game of chess

Playing chess



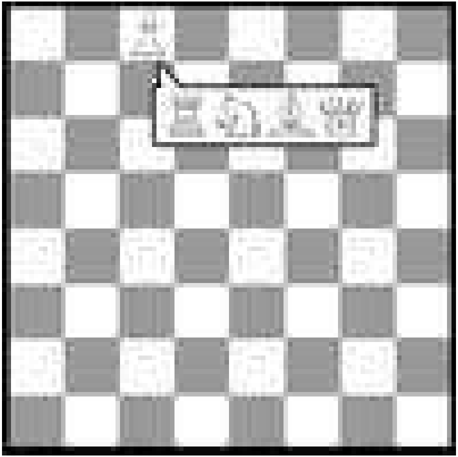
Castling

Playing chess



Forking

Playing chess



Pawn exchange

Outline for today

- Notions of explicit strategizing in different logics
- Structured strategizing in large games
- A dual way of reasoning in large games

Coalition action logic

- Action logic - quantificational dynamic modalities [Borgo, 2005]

Coalition action logic

- Action logic - quantificational dynamic modalities [Borgo, 2005]
- Coalition logic - $[C]\varphi$

Coalition action logic

- Action logic - quantificational dynamic modalities [Borgo, 2005]
- Coalition logic - $[C]\varphi$
- puts actions for the members of the coalitions. [Borgo, 2007]

S. Borgo. Quantificational modal logic with sequential Kripke semantics. *JANCL*, 15(2): pages 137–188, 2005.

S. Borgo. Coalition in action logic. In M.M. Veloso, editor, *Proceedings of IJCAI'07*, pages 1822–1827, 2007.

Alternating-time temporal logic

- Reactive systems modeled as games
- Selective quantification over path

Alternating-time temporal logic

- Reactive systems modeled as games
- Selective quantification over path

$$\varphi := p \mid \neg\varphi \mid \varphi \vee \varphi \mid \langle\!\langle C \rangle\!\rangle \bigcirc \varphi \mid \langle\!\langle C \rangle\!\rangle \Box \varphi \mid \langle\!\langle C \rangle\!\rangle \varphi \,\mathcal{U}\, \varphi,$$

$$p \in \Phi_0, C \subseteq N$$

Alternating transition systems

- $\mathcal{M} = (S, V, \delta)$, where $\delta : S \times N \to \mathcal{P}(\mathcal{P}(S))$.

Alternating transition systems

- $\mathcal{M} = (S, V, \delta)$, where $\delta : S \times N \rightarrow \mathcal{P}(\mathcal{P}(S))$.
- A strategy for $i \in N$: $f_i : S^+ \rightarrow \mathcal{P}(S)$, where, $f_i(s_0, \ldots, s_k) \in \delta(s_k, i)$

Alternating transition systems

- $\mathcal{M} = (S, V, \delta)$, where $\delta : S \times N \rightarrow \mathcal{P}(\mathcal{P}(S))$.
- A strategy for $i \in N$: $f_i : S^+ \rightarrow \mathcal{P}(S)$, where, $f_i(s_0, \ldots, s_k) \in \delta(s_k, i)$
- Collective strategy for $C \subseteq N$: $F_C = \langle f_i \rangle_{i \in C}$

Alternating transition systems

- $\mathcal{M} = (S, V, \delta)$, where $\delta : S \times N \rightarrow \mathcal{P}(\mathcal{P}(S))$.
- A strategy for $i \in N$: $f_i : S^+ \rightarrow \mathcal{P}(S)$, where, $f_i(s_0, \ldots, s_k) \in \delta(s_k, i)$
- Collective strategy for $C \subseteq N : F_C = \langle f_i \rangle_{i \in C}$
- s-computations : infinite sequence of successor states starting from s respecting δ.

Alternating transition systems

- $\mathcal{M} = (S, V, \delta)$, where $\delta : S \times N \rightarrow \mathcal{P}(\mathcal{P}(S))$.
- A strategy for $i \in N$: $f_i : S^+ \rightarrow \mathcal{P}(S)$, where, $f_i(s_0, \ldots, s_k) \in \delta(s_k, i)$
- Collective strategy for $C \subseteq N$: $F_C = \langle f_i \rangle_{i \in C}$
- s-computations : infinite sequence of successor states starting from s respecting δ.
- $out(s, F_C)$ is the set of s-computations when C is using F_C.

Counterfactual ATL

- strategic commitment of agents.

Counterfactual ATL

- strategic commitment of agents.
- $C_i(\sigma, \varphi)$: if it were the case that player i committed to a strategy σ, then φ would hold.

Counterfactual ATL

- strategic commitment of agents.
- $C_i(\sigma, \varphi)$: if it were the case that player i committed to a strategy σ, then φ would hold.
- solution concepts in strategic games.

W. van der Hoek, W. Jamroga, and M. Wooldridge. A logic for strategic reasoning. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 05)*, pages 157–164. ACM Inc, New York, 2005.

ATL with intentions

- strategic ability in accordance with agents' intentions

ATL with intentions

- strategic ability in accordance with agents' intentions
- $(\mathbf{str}_a \sigma)\varphi$: had the player played her strategy according to her intentions, φ would hold

ATL with intentions

- strategic ability in accordance with agents' intentions
- $(\mathbf{str}_a \sigma)\varphi$: had the player played her strategy according to her intentions, φ would hold
- solution concepts parametrized by temporal operators

W. Jamroga, W. van der Hoek, and M. Wooldridge. Intentions and strategies in game-like scenarios. In C. Bento, A. Cardoso, and G. Dias, editors, *Proceedings of EPIA 2005, LNAI 3808*, pages 512–523. Springer-Verlag, 2005.

ATL with explicit strategies

- extending the idea of CATL to extensive form games

ATL with explicit strategies

- extending the idea of CATL to extensive form games
- $\langle\langle A \rangle\rangle_\rho \varphi$, ρ representing partial commitment functions, mapping agents to strategy terms

D. Walther, W. van der Hoek, and M. Wooldridge. Alternating-time temporal logic with explicit strategies. In *Proceedings of XIth Conference (Theoretical Aspects of Rationality and Knowledge)*, pages 269–278, 2007.

First-order strategy logic

- two player nonzero-sum graph games

First-order strategy logic

- two player nonzero-sum graph games
- first order extension in two variables of basic temporal logic

First-order strategy logic

- two player nonzero-sum graph games
- first order extension in two variables of basic temporal logic
- $Qx.\varphi$ or $Qy.\varphi$, where $Q \in \{\exists, \forall\}$, $x \in X$, $y \in Y$

First-order strategy logic

- two player nonzero-sum graph games
- first order extension in two variables of basic temporal logic
- $Qx.\varphi$ or $Qy.\varphi$, where $Q \in \{\exists, \forall\}$, $x \in X$, $y \in Y$
- highly expressive, various solution concepts

K. Chatterjee, T.A. Henzinger, and N. Piterman. Strategy logic. In *Proceedings of the 18th International Conference on Concurrency Theory (CONCUR 07), LNCS 4703*. pages 59–73. Springer-Verlag, 2007.

Second-order strategy logic

- constraints on strategies in ATS

Second-order strategy logic

- constraints on strategies in ATS
- propositional μ-calculus enriched with decision modalities and second-order predicates

Second-order strategy logic

- constraints on strategies in ATS
- propositional μ-calculus enriched with decision modalities and second-order predicates
- $\diamond_n P$: the set of propositions P is ensured by some move of the player n from the current state

Second-order strategy logic

- constraints on strategies in ATS
- propositional μ-calculus enriched with decision modalities and second-order predicates
- $\diamond_n P$: the set of propositions P is ensured by some move of the player n from the current state
- highly expressive, fairness constraints into the syntax

S. Pinchinat. A generic constructive solution for concurrent games with expressive constraints on strategies. In *Proceedings of the 5th International Symposium on Automated Technology for Verification and Analysis (ATVA'07), LNCS 4762*. pages 253–267. Springer-Verlag, 2007.

Common features

- explicit formulation of strategies
- names are given to strategies
- atomic terms, quantifications, constraints
- good descriptions of different solution concepts of game theory

Structured strategies

Choices made by players depend on:

- Observations made during the play.
- Response to observed behaviour of other players.

Structured strategies

Choices made by players depend on:

- Observations made during the play.
- Response to observed behaviour of other players.

Strategies are better viewed as relations constraining moves rather than complete functions.

Question: Can we come up with a framework where strategies are specified as structured objects built in some compositional fashion ?

Present focus

A formal study of structured strategies in extensive form games:

Present focus

A formal study of structured strategies in extensive form games:

- A syntactic framework where local partially specified strategies are composed in a structured fashion.

Present focus

A formal study of structured strategies in extensive form games:

- A syntactic framework where local partially specified strategies are composed in a structured fashion.
- A dual framework describing structured game trees composed of simple subgames, where strategies are complete plans to ensure local outcomes.

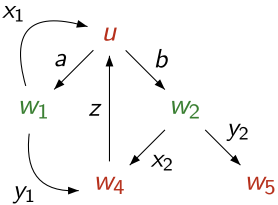
Present focus

A formal study of structured strategies in extensive form games:

- A syntactic framework where local partially specified strategies are composed in a structured fashion.
- A dual framework describing structured game trees composed of simple subgames, where strategies are complete plans to ensure local outcomes.
- Independent of the exact depth of the game tree.

Games on graphs

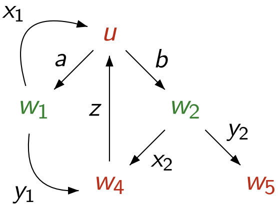
Game model - directed graph where nodes are labelled with players.



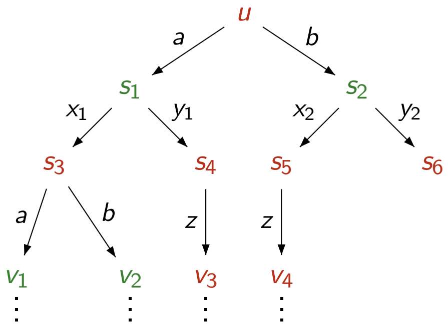
Game arena

Games on graphs

Game model - directed graph where nodes are labelled with players.



Game arena



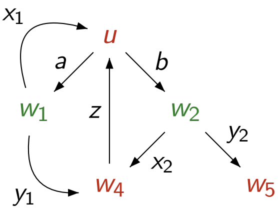
Extensive form game tree

P - countable set of observables

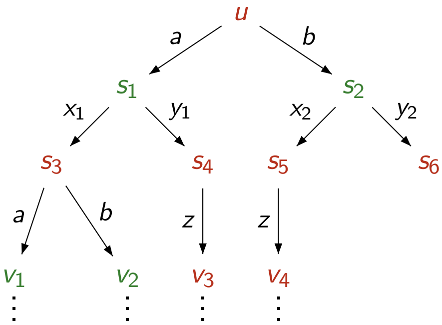
$V : Nodes \rightarrow 2^P$

Games on graphs

Strategies of players - subtrees of the game tree.



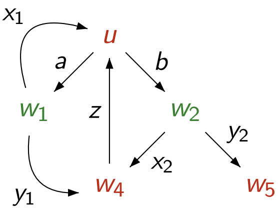
Game arena



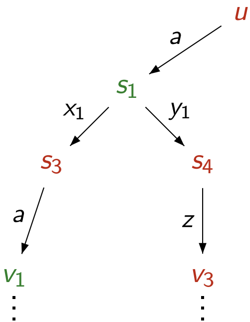
Extensive form game tree

Games on graphs

Strategies of players - subtrees of the game tree.



Game arena



A strategy of player 1

Strategy specification

$$Strat^i(P^i) := [\psi \mapsto a]^i$$

Interpretation

- $[\psi \mapsto a]^i$: If the observable ψ holds then choose action a (positional strategies).

Strategy specification

$$Strat^i(P^i) := [\psi \mapsto a]^i \mid \sigma_1 + \sigma_2 \mid \sigma_1 \cdot \sigma_2$$

Interpretation

- $[\psi \mapsto a]^i$: If the observable ψ holds then choose action a (positional strategies).
- $\sigma_1 + \sigma_2$: Disjunction.
- $\sigma_1 \cdot \sigma_2$: Conjunction.

Strategy specification

$$Strat^i(P^i) := [\psi \mapsto a]^i \mid \sigma_1 + \sigma_2 \mid \sigma_1 \cdot \sigma_2 \mid \pi \Rightarrow \sigma.$$

- π - specification of player $\bar{\imath}$.

Interpretation

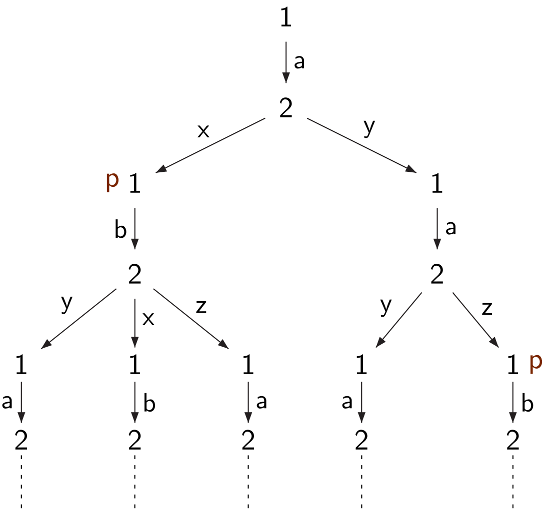
- $[\psi \mapsto a]^i$: If the observable ψ holds then choose action a (positional strategies).
- $\sigma_1 + \sigma_2$: Disjunction.
- $\sigma_1 \cdot \sigma_2$: Conjunction.
- $\pi \Rightarrow \sigma$: If in the history the observed behaviour of player $\bar{\imath}$ conforms to π then play according to σ.

Strategy specification

- Strategy specifications need not define complete strategies.
- Define when a (functional) strategy satisfies a specification.

Strategy conforming to a specification

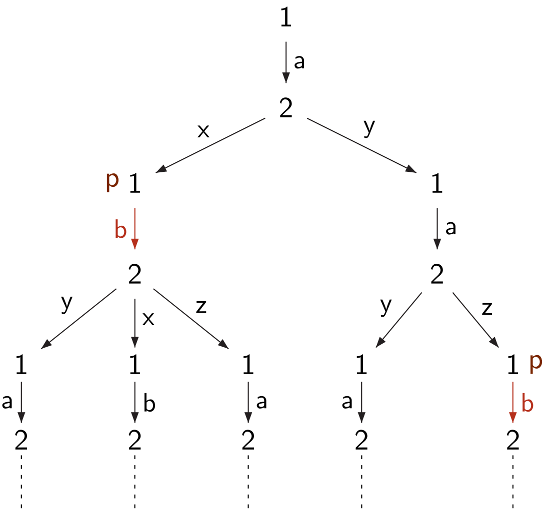
Player 1 strategy.



Strategy conforming to a specification

Player 1 strategy.

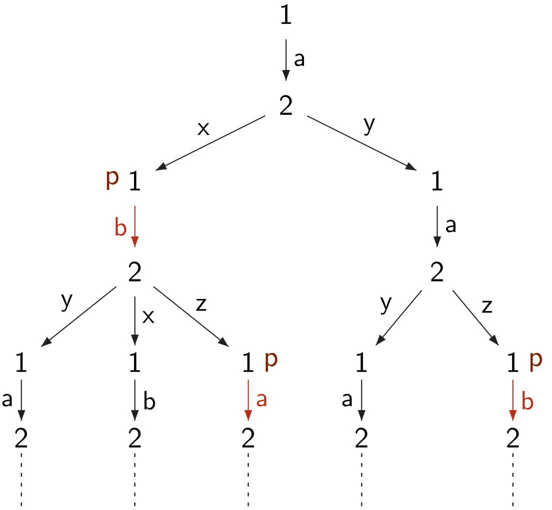
$[p \mapsto b]^1$



Strategy conforming to a specification

Player 1 strategy.

$[p \mapsto b]^1$

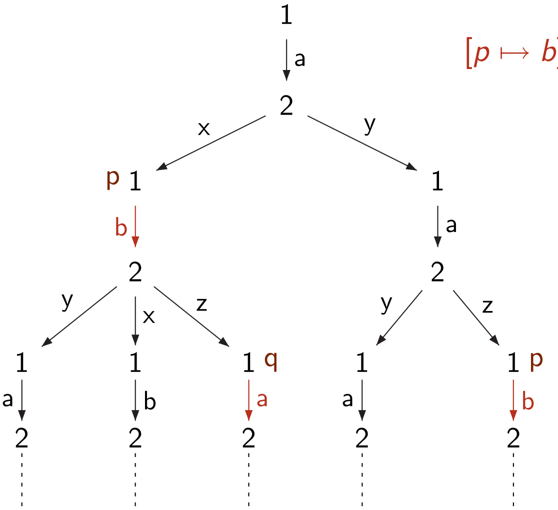


Strategy conforming to a specification

Player 1 strategy.

$\sigma_1 \cdot \sigma_2$

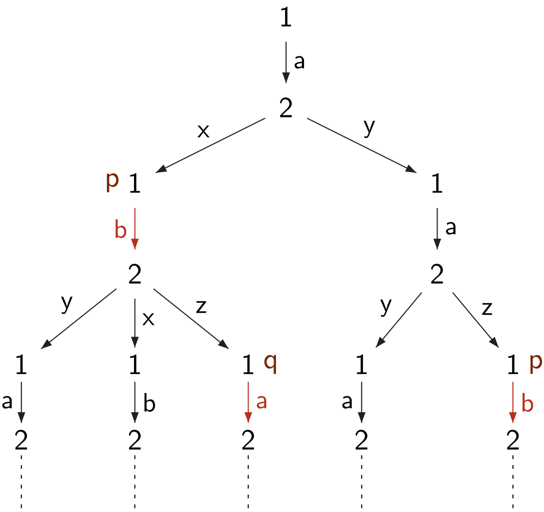
$[p \mapsto b]^1 \cdot [q \mapsto a]^1$



Strategy conforming to a specification

Player 1 strategy.

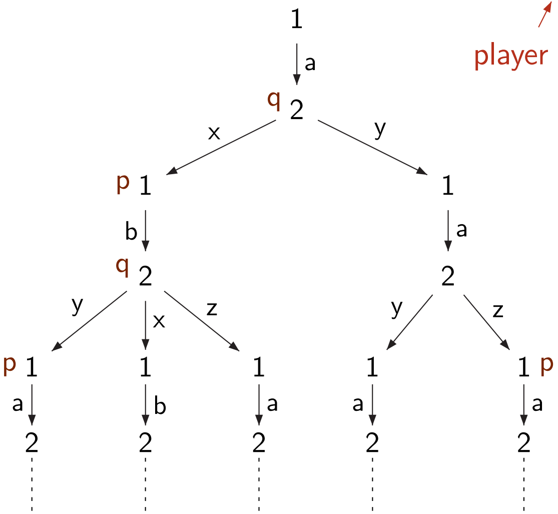
$\sigma_1 + \sigma_2$



Strategy conforming to a specification

$\pi \Rightarrow \sigma$

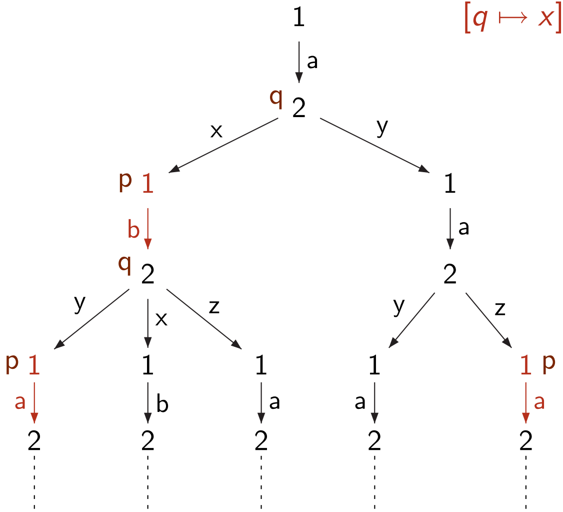
player 2 player 1



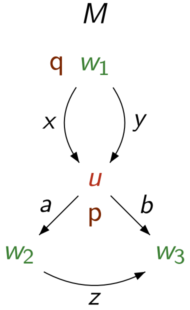
Strategy conforming to a specification

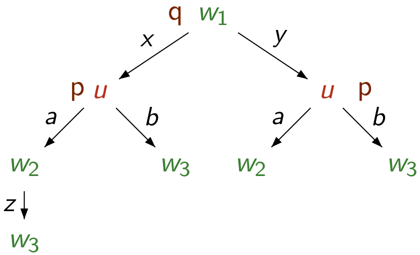
$$\pi \Rightarrow \sigma$$

$$[q \mapsto x]^2 \Rightarrow [p \mapsto b]^1$$

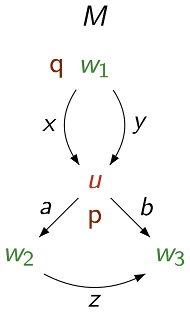


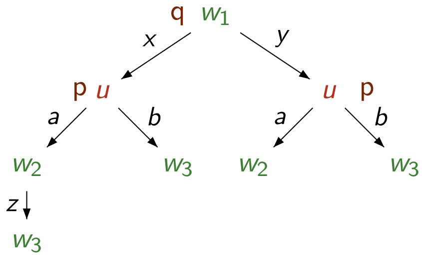
Examples - strategy specification

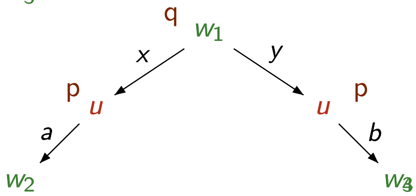




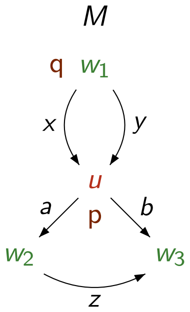
Examples - strategy specification

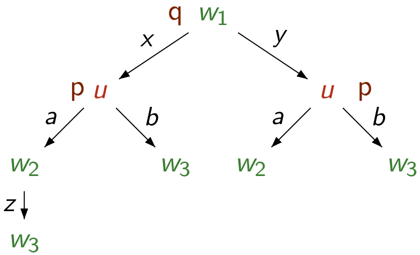


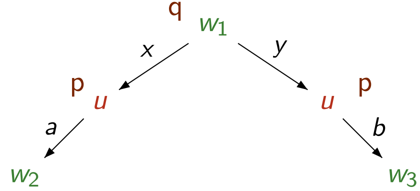




Examples - strategy specification







$$[q \mapsto x]^2 \Rightarrow [p \mapsto a]^1$$
$$[q \mapsto y]^2 \Rightarrow [p \mapsto b]^1$$

A strategy logic

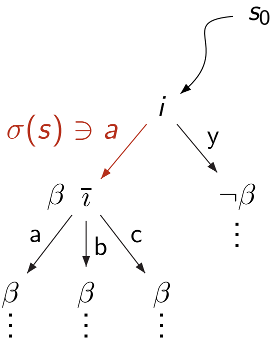
$$p \in P \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle a \rangle \alpha \mid \langle \overline{a} \rangle \alpha \mid \diamondminus\alpha \mid (\sigma)_i : c \mid \sigma \leadsto_i \beta$$

- $(\sigma)_i : c$ - The move c is enabled by the specification σ.
- $\sigma \leadsto_i \beta$ - The strategy specification σ ensures the outcome β.

A strategy logic

$$p \in P \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle a \rangle \alpha \mid \langle \overline{a} \rangle \alpha \mid \diamondminus \alpha \mid (\sigma)_i : c \mid \sigma \leadsto_i \beta$$

- $(\sigma)_i : c$ - The move c is enabled by the specification σ.
- $\sigma \leadsto_i \beta$ - The strategy specification σ ensures the outcome β.



A strategy logic

Empty specification: null^i - existence of strategies.

- $\mathit{null}^i \rightsquigarrow_i \beta$ - There exists a strategy to ensure the outcome β.
- $\sigma \rightsquigarrow_i \beta$ - The mechanism used by the player to ensure β is specified by the property σ.

Finite extensive form games - special case in our setting.

Model

- Model - Kripke structure.
  - A finite set of states W.
  - Labelled edge relation $\longrightarrow \subseteq W \times \Sigma \times W$.
  - Valuation function $V : W \to 2^P$.
  - Player labelling function $\lambda : W \to N$.

Semantics

Let Σ denote the set of all actions of players.

For a game position s, $\sigma(s)$ denotes the actions allowed by σ at s.

Semantics

Let Σ denote the set of all actions of players.

For a game position s, $\sigma(s)$ denotes the actions allowed by σ at s.

- $[p \mapsto a]^1(s) = \begin{cases} \{a\} & \text{if } p \in V(s) \text{ and } s \text{ is a 1 node} \\ \Sigma & \text{otherwise} \end{cases}$

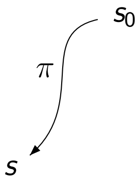
Semantics

Let Σ denote the set of all actions of players.

For a game position s, $\sigma(s)$ denotes the actions allowed by σ at s.

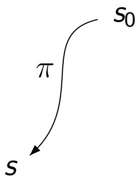
- $[p \mapsto a]^1(s) = \begin{cases} \{a\} & \text{if } \; p \in V(s) \text{ and } s \text{ is a 1 node} \\ \Sigma & \text{otherwise} \end{cases}$
- $(\sigma_1 + \sigma_2)(s) = \sigma_1(s) \cup \sigma_2(s)$
- $(\sigma_1 \cdot \sigma_2)(s) = \sigma_1(s) \cap \sigma_2(s)$

Semantics



$$(\pi \Rightarrow \sigma)(s) = \begin{cases} \sigma(s) & \text{if } \forall j : 0 \leq j < m,\ a_j \in \pi(s_j) \\ \Sigma & \text{otherwise} \end{cases}$$

Semantics

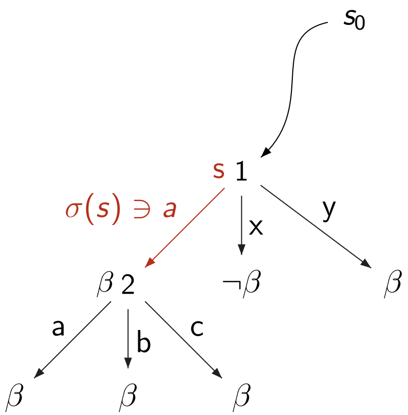


$$(\pi \Rightarrow \sigma)(s) = \begin{cases} \sigma(s) & \text{if } \forall j : 0 \leq j < m,\ a_j \in \pi(s_j) \\ \Sigma & \text{otherwise} \end{cases}$$

$M, s \models (\sigma)_i : c$ iff $c \in \sigma(s)$.

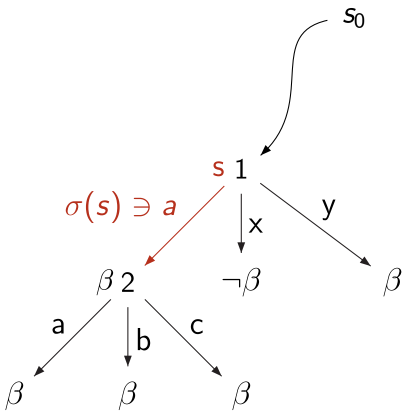
Semantics

$M, s \models \sigma \rightsquigarrow_1 \beta$



Semantics

$M, s \models \sigma \rightsquigarrow_1 \beta$ iff $\forall s' \in \mathcal{T}_s \upharpoonright \sigma$, such that $s \longrightarrow^* s'$, we have $M, s' \models \beta \wedge (\mathbf{turn}_i \Rightarrow enabled_\sigma)$.



Truth checking

Truth checking problem: Given a model $M = (T, V)$ and a formula α, check if $M, s_0 \models \alpha$.

Truth checking

Truth checking problem: Given a model $M = (T, V)$ and a formula α, check if $M, s_0 \models \alpha$.

Theorem

Truth checking problem is decidable.

Truth checking

Truth checking problem: Given a model $M = (T, V)$ and a formula α, check if $M, s_0 \models \alpha$.

Theorem

Truth checking problem is decidable.

Proof idea

- $\{\sigma_1, \ldots, \sigma_m\}$ - strategy specifications in α.

Truth checking

Truth checking problem: Given a model $M = (T, V)$ and a formula α, check if $M, s_0 \models \alpha$.

Theorem

Truth checking problem is decidable.

Proof idea

- $\{\sigma_1, \ldots, \sigma_m\}$ - strategy specifications in α.
- $\{\mathcal{A}_{\sigma_1}, \ldots, \mathcal{A}_{\sigma_m}\}$ - corresponding advice automata.

Truth checking

Truth checking problem: Given a model $M = (T, V)$ and a formula α, check if $M, s_0 \models \alpha$.

Theorem

Truth checking problem is decidable.

Proof idea

- $\{\sigma_1, \ldots, \sigma_m\}$ - strategy specifications in α.
- $\{\mathcal{A}_{\sigma_1}, \ldots, \mathcal{A}_{\sigma_m}\}$ - corresponding advice automata.

Construct a tree automata $\mathcal{T}$ which keeps track of:

- the atoms of α,
- states of each of the advice automaton.

Truth checking

Truth checking problem: Given a model $M = (T, V)$ and a formula α, check if $M, s_0 \models \alpha$.

Theorem

Truth checking problem is decidable.

Proof idea

- $\{\sigma_1, \ldots, \sigma_m\}$ - strategy specifications in α.
- $\{\mathcal{A}_{\sigma_1}, \ldots, \mathcal{A}_{\sigma_m}\}$ - corresponding advice automata.

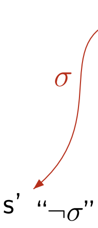
Construct a tree automata $\mathcal{T}$ which keeps track of:

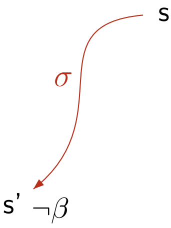
- the atoms of α,
- states of each of the advice automaton.

$$p \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle a \rangle \alpha \mid \langle \overline{a} \rangle \alpha \mid \boxminus \alpha \mid (\sigma)_i : c \mid \sigma \leadsto_i \beta$$

Truth checking

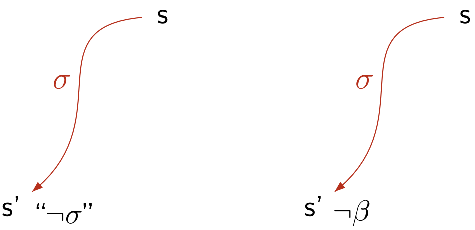
$\neg(\sigma \rightsquigarrow_i \beta)$ - requirement.





Truth checking

$\neg(\sigma \rightsquigarrow_i \beta)$ - requirement.



- $\mathcal{T}$ keeps track of a requirement set U
- For each branch, guesses a partition of requirements.
- Good states - states where U is empty.

Outcome based analysis

Finite extensive form games - special case in our setting.

- Utilities can be coded in terms of propositions.
- Characteristic formulas can be given for:
  - Best response.
  - Dominant strategies.
  - Equilibrium.

Games with compositional structure

Logical analysis of strategies: Explicates the strategic reasoning of players.

- The game representation is taken to be atomic.
- Logical formalism does not dictate the structure of the game.

Games with compositional structure

Logical analysis of strategies: Explicates the strategic reasoning of players.

- The game representation is taken to be atomic.
- Logical formalism does not dictate the structure of the game.

Question: Can we think of the game to be built in a compositional manner?

Game logic [Parikh]

A logic to reason about determined two person zero sum games.

Syntax

- $\Phi := p \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle\gamma\rangle\alpha$.
- $\Gamma := g \in \Gamma_0 \mid \gamma_1 ; \gamma_2 \mid \gamma_1 \cup \gamma_2 \mid \gamma^d \mid \gamma^*$.

Game logic [Parikh]

A logic to reason about determined two person zero sum games.

Syntax

- $\Phi := p \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle\gamma\rangle\alpha$.
- $\Gamma := g \in \Gamma_0 \mid \gamma_1;\gamma_2 \mid \gamma_1 \cup \gamma_2 \mid \gamma^d \mid \gamma^*$.

Interpretation for games

- Final outcomes which players can enforce.
- Set of states S.
- Effectivity relation - $E_g \subseteq S \times 2^S$
  - $(s, X) \in E_g$ iff starting at s, in game g, player 1 can enforce the outcome to be in X.

Game logic [Parikh]

- $\Phi := p \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle\gamma\rangle\alpha$.
- $\Gamma := g \in \Gamma_0 \mid \gamma_1;\gamma_2 \mid \gamma_1 \cup \gamma_2 \mid \gamma^d \mid \gamma^*$.

Model $M = (S, \{E_g \mid g \in \Gamma_0\}, V)$.

Neighbourhood semantics

- $M, s \models \langle\gamma\rangle\alpha$ iff $\exists(s, X) \in E_\gamma$ such that $X \subseteq \{s' \mid M, s' \models \alpha\}$.
- Player 1 has the ability in game γ to ensure α.
- Talks about players' abilities to achieve certain objectives.

Games with compositional structure

Players' strategies need to take into account:

- Ensuring local outcomes in simple sub-games.
- Compositional structure of the game.

Games with compositional structure

Players' strategies need to take into account:

- Ensuring local outcomes in simple sub-games.
- Compositional structure of the game.

At the logical level:

- Strategies can be thought of as complete plans in simple sub-games.
- Game composition and such complete strategies are *not* independent entities.
- Games and strategies need to be composed together.

The logic

Syntax

- $\Phi := p \in P \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle g, i \rangle \alpha$.
- $\Gamma := (h, \beta) \mid g_1; g_2 \mid g_1 \cup g_2 \mid g^*$.

The logic

Syntax

- $\Phi := p \in P \mid \neg\alpha \mid \alpha_1 \vee \alpha_2 \mid \langle g, i\rangle\alpha$.
- $\Gamma := (h, \beta) \mid g_1; g_2 \mid g_1 \cup g_2 \mid g^*$.

Neighbourhood semantics

- $M, s \models \langle g, i\rangle\alpha$ iff $\exists(s, X) \in R_g^i$ such that $X \subseteq \{s' \mid M, s' \models \alpha\}$.

Neighbourhood semantics

To define the neighbourhood relation, we need to fix:

- Representation of game h.

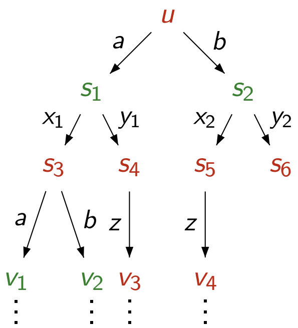
Atomic games: Extensive form games

- Finite tree - nodes represent game positions labelled with players.
- Edge relation - specifies the moves which are enabled at a particular position.

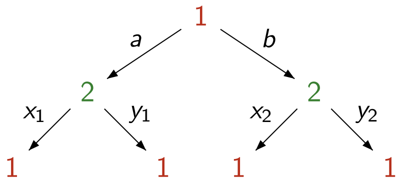
Model

- Model - pairs $M = (T, V)$.
  - $T = (S, \Longrightarrow, s_0, \widehat{\lambda})$ is an extensive form game tree.
    - S is a set of nodes.
    - $\Longrightarrow: S \times \Sigma \rightarrow S$
    - $\widehat{\lambda}$ is a player labelling function.
  - Valuation function $V : S \rightarrow 2^P$.
- When $g = (h, \beta)$, a pair $(s, X) \in R_g^i$ if h is enabled at state s and there is a strategy for player i to ensure outcome β such that X is the set of leaf nodes of the strategy.

Game - outcome pairs

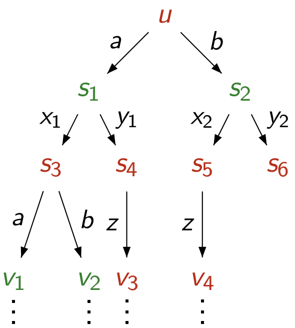


Model - M

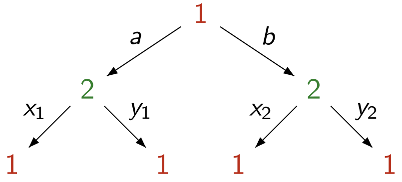


Game - h

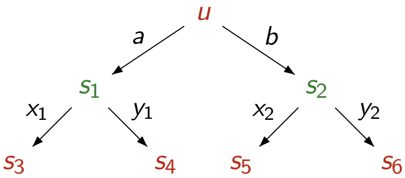
Game - outcome pairs



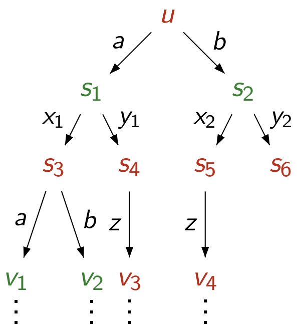
Model - M



Game - h

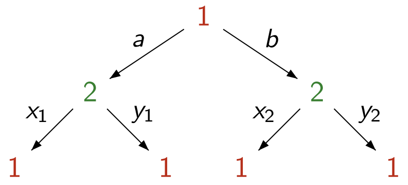


Game - outcome pairs

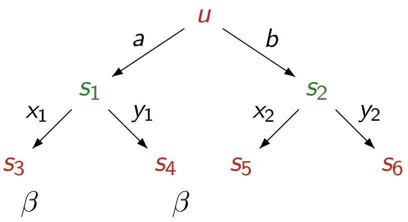


Model - M

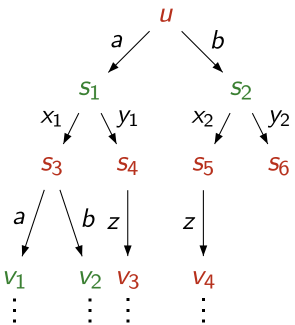
$$g = (h, \beta)$$



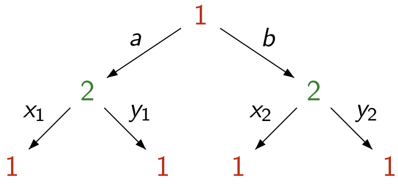
Game - h



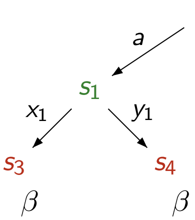
Game - outcome pairs



Model - M



Game - h



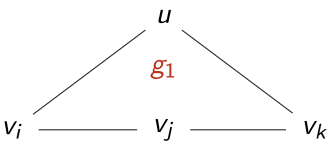
$$g = (h, \beta)$$

$$(u, \{s_3, s_4\}) \in R^i_{(h,\beta)}$$

Semantics of game - outcome pairs

$$\Gamma := (h, \beta) \mid g_1 ; g_2 \mid g_1 \cup g_2 \mid g^*$$

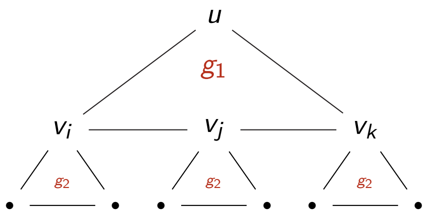
- $(u, X) \in R^i_{g_1;g_2}$



Semantics of game - outcome pairs

$$\Gamma := (h, \beta) \mid g_1 ; g_2 \mid g_1 \cup g_2 \mid g^*$$

- $(u, X) \in R^i_{g_1;g_2}$



Semantics of game - outcome pairs

$\Gamma := (h, \beta) \mid g_1; g_2 \mid g_1 \cup g_2 \mid g^*$

- $(u, X) \in R^i_{g_1;g_2}$ iff
  - $\exists Y = \{v_1, \ldots, v_k\}$ such that $(u, Y) \in R^i_{g_1}$.
  - $\forall v_j \in Y$, $\exists X_j \in X$ such that $(v_j, X_j) \in R^i_{g_2}$.
  - $X = \bigcup_{j=1,\ldots,k} X_j$.

Semantics of game - outcome pairs

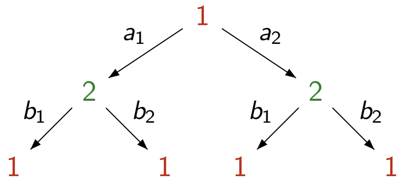
$\Gamma := (h, \beta) \mid g_1; g_2 \mid g_1 \cup g_2 \mid g^*$

- $(u, X) \in R^i_{g_1;g_2}$ iff
  - $\exists Y = \{v_1, \ldots, v_k\}$ such that $(u, Y) \in R^i_{g_1}$.
  - $\forall v_j \in Y$, $\exists X_j \in X$ such that $(v_j, X_j) \in R^i_{g_2}$.
  - $X = \bigcup_{j=1,\ldots,k} X_j$.
- $R^i_{g_1 \cup g_2} = R^i_{g_1} \cup R_{g_2}$.
- $R^i_{g^*} = \bigcup_{n \geq 0} (R^i_g)^n$.

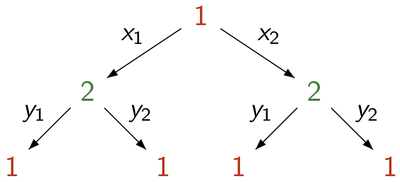
Examples - game composition

- Consider a two stage game h_1 followed by h_2.
- Player 1's planning at the end of h_1 may not only depend on how h_2 is structured but also on how player 2 played in h_1.
- (h_1, β) and $(h_2, \top)$.
  - (h_1, β): a complete strategy of player 2.
  - $(h_2, \top)$: a complete strategy of player 1.
- $(h_2, \top)$ is the response of player 1 to (h_1, β).

Examples - game composition



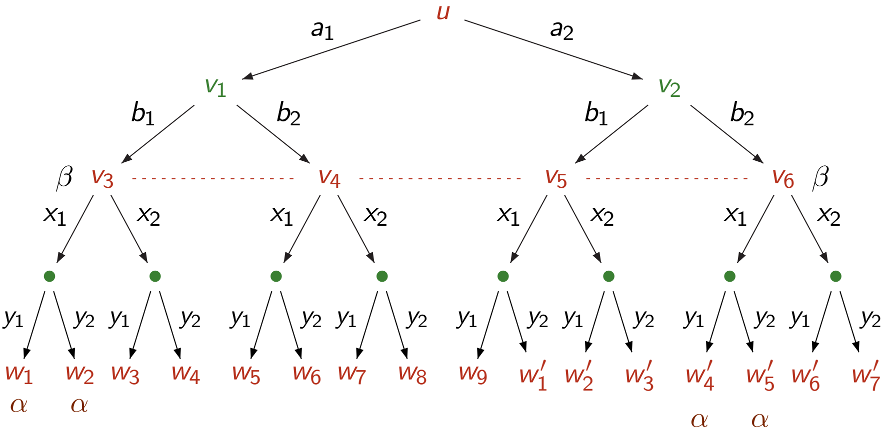
h_1



h_2

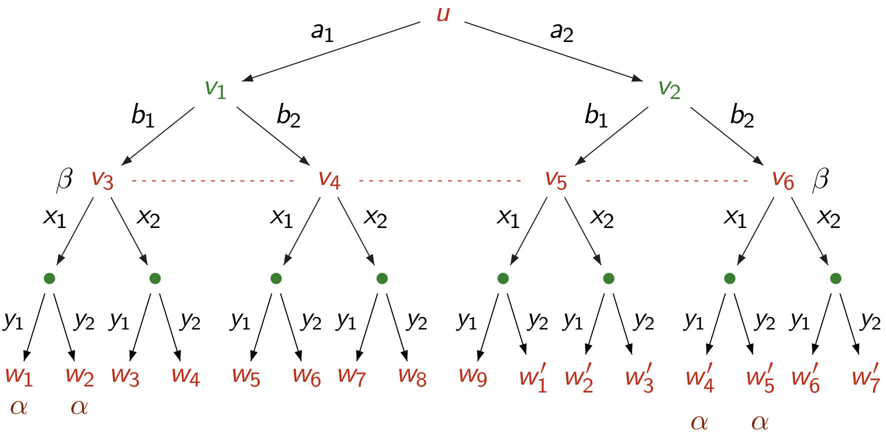
$h_1; h_2$

Examples - game composition



- Player 1 does not have a strategy in g to ensure α.

Examples - game composition



- Let $g_1 \equiv (h_1, \beta)$ and $g_2 \equiv (h_2, \top)$.
- $M, u \models \langle g_1, 2\rangle\langle g_2, 1\rangle\alpha$

Axiom system

- $\langle (h, \beta), i \rangle \alpha \equiv$?

(Informally): Game h is enabled and there exists a strategy μ ensuring β such that *frontier*(μ) satisfies α.

Axiom system

- $\langle (h, \beta), i \rangle \alpha \equiv h^{\checkmark} \wedge \downarrow_{(h,i,\beta,\alpha)}$.

(Informally): Game h is enabled and there exists a strategy μ ensuring β such that *frontier*(μ) satisfies α.

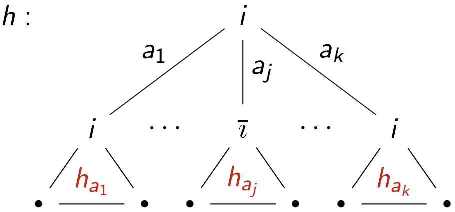
- $\langle a \rangle \alpha$ - can be encoded in the logic.
- $h^{\checkmark}$ can be defined.

Definition of *push*

h is a single node:

- $\downarrow_{(h,i,\beta,\alpha)} = \beta \wedge \alpha$.

Axiom system



$\downarrow_{(h,i,\beta,\alpha)}$ holds at state u:

- $\exists w$ such that $u \xrightarrow{a} w$ and $\langle (h_a, \beta), i \rangle \alpha$ holds at w.

Axiom system

General idea behind *push*

- If the root is a player i-node then
  - an edge is chosen and the requirement is "pushed" to the relevant subtree.
- If the root is an $\bar{\imath}$-node then
  - all outgoing edges need to be taken into account and the requirement is "pushed" to all the resulting subtrees.

Axiom system

- Propositional axioms:
  - All the substitutional instances of tautologies of PC.
  - $turn_i \equiv \neg turn_{\bar{\imath}}$.
- Axiom for single edge games:
  - $\langle a \rangle(\alpha_1 \vee \alpha_2) \equiv \langle a \rangle \alpha_1 \vee \langle a \rangle \alpha_2$.
  - $\langle a \rangle turn_i \supset [a] turn_i$.
- Dynamic logic axioms:
  - $\langle g_1 \cup g_2, i \rangle \alpha \equiv \langle g_1, i \rangle \alpha \vee \langle g_2, i \rangle \alpha$.
  - $\langle g_1 ; g_2, i \rangle \alpha \equiv \langle g_1, i \rangle \langle g_2, i \rangle \alpha$.
  - $\langle g^*, i \rangle \alpha \equiv \alpha \vee \langle g, i \rangle \langle g^*, i \rangle \alpha$.

Inference rules

$$(MP)\ \frac{\alpha, \quad \alpha \supset \beta}{\beta} \qquad (NG)\ \frac{\alpha}{[a]\alpha}$$

$$(IND)\ \frac{\langle g, i\rangle\alpha \supset \alpha}{\langle g^*, i\rangle\alpha \supset \alpha}$$

Summing up

- Various logics of games and strategies

Summing up

- Various logics of games and strategies
- Different kinds of compositions of game trees.

Summing up

- Various logics of games and strategies
- Different kinds of compositions of game trees.
- How to reason in large games, introducing structures in strategies

Summing up

- Various logics of games and strategies
- Different kinds of compositions of game trees.
- How to reason in large games, introducing structures in strategies
- Perfect information game trees.

Summing up

- Various logics of games and strategies
- Different kinds of compositions of game trees.
- How to reason in large games, introducing structures in strategies
- Perfect information game trees.
- Imperfect information, large number of players, stabilising strategies.