How to be both rich and happy: Combining quantitative and qualitative strategic reasoning in multi-player games

Valentin Goranko
Technical University of Denmark
and
Nils Bulling
Clausthal University of Technology, Germany

Highlights’2013
Paris, September 21, 2013
Introduction:
strategic abilities of agents in multi-player games

Two traditions:
Introduction:
strategic abilities of agents in multi-player games

Two traditions:

Quantitative: study of the abilities of rational players achieve quantitative objectives: optimizing payoffs or, more generally, preferences on outcomes.
Introduction:
strategic abilities of agents in multi-player games

Two traditions:

Quantitative: study of the abilities of rational players achieve quantitative objectives: optimizing payoffs or, more generally, preferences on outcomes.

Typical models:
normal form games, repeated games, extensive games.
Introduction: strategic abilities of agents in multi-player games

Two traditions:

Quantitative: study of the abilities of rational players achieve quantitative objectives: optimizing payoffs or, more generally, preferences on outcomes.

Typical models: normal form games, repeated games, extensive games.

Qualitative: study of strategic abilities of players for achieving qualitative objectives: reaching or maintaining outcome states with desired properties, e.g., winning states, or safe states, etc.
Introduction:
strategic abilities of agents in multi-player games

Two traditions:

Quantitative: study of the abilities of rational players achieve quantitative objectives: optimizing payoffs or, more generally, preferences on outcomes.

Typical models:
normal form games, repeated games, extensive games.

Qualitative: study of strategic abilities of players for achieving qualitative objectives: reaching or maintaining outcome states with desired properties, e.g., winning states, or safe states, etc.

Typical models:
multi-agent transition systems, a.k.a. concurrent game models.
Introduction:
strategic abilities of agents in multi-player games

Two traditions:

Quantitative: study of the abilities of rational players achieve quantitative objectives: optimizing payoffs or, more generally, preferences on outcomes.

Typical models:
normal form games, repeated games, extensive games.

Qualitative: study of strategic abilities of players for achieving qualitative objectives: reaching or maintaining outcome states with desired properties, e.g., winning states, or safe states, etc.

Typical models:
multi-agent transition systems, a.k.a. concurrent game models.

We develop a logical framework combining both traditions.
Introduction:
strategic abilities of agents in multi-player games

Two traditions:

**Quantitative**: study of the abilities of rational players achieve quantitative objectives: optimizing payoffs or, more generally, preferences on outcomes.

Typical models:
- normal form games, repeated games, extensive games.

**Qualitative**: study of strategic abilities of players for achieving qualitative objectives: reaching or maintaining outcome states with desired properties, e.g., winning states, or safe states, etc.

Typical models:
- multi-agent transition systems, a.k.a. concurrent game models.

We develop a logical framework combining both traditions.

Builds on several existing types of models and logics.
Towards quantitative reasoning:
Concurrent game models with payoffs and guards
Towards quantitative reasoning:
Concurrent game models with payoffs and guards

Concurrent game model with payoffs and guards (GGMPG):
extend concurrent game models by associating with every state a strategic game with payoffs. Thus:

– at every state each player chooses an action; all actions are applied simultaneously and determine transition to successor state;
– the collective action also determines each player's payoff;
– same happens at the successor state, etc., thus eventually generating an infinite play;

So, players accumulate utilities in the course of the play;
The players' current utility values determine their available actions at the current state, by means of guards – arithmetical constraints over the current utilities.

CGMPGs: games with qualitative and quantitative objectives.
Towards quantitative reasoning:
Concurrent game models with payoffs and guards

Concurrent game model with payoffs and guards (GGMPG):
extend concurrent game models by associating with every state a strategic game with payoffs. Thus:

– at every state each player chooses an action; all actions are applied simultaneously and determine transition to successor state;
Towards quantitative reasoning: Concurrent game models with payoffs and guards

Concurrent game model with payoffs and guards (GGMPG): extend concurrent game models by associating with every state a strategic game with payoffs. Thus:

– at every state each player chooses an action; all actions are applied simultaneously and determine transition to successor state;
– the collective action also determines each player’s payoff;
Towards quantitative reasoning:
Concurrent game models with payoffs and guards

Concurrent game model with payoffs and guards (GGMPG): extend concurrent game models by associating with every state a strategic game with payoffs. Thus:

– at every state each player chooses an action; all actions are applied simultaneously and determine transition to successor state;

– the collective action also determines each player’s payoff;

– same happens at the successor state, etc., thus eventually generating an infinite play;
Towards quantitative reasoning:
Concurrent game models with payoffs and guards

Concurrent game model with payoffs and guards (GGMPG): extend concurrent game models by associating with every state a strategic game with payoffs. Thus:

– at every state each player chooses an action; all actions are applied simultaneously and determine transition to successor state;

– the collective action also determines each player’s payoff;

– same happens at the successor state, etc., thus eventually generating an infinite play;

So, players accumulate utilities in the course of the play;
Towards quantitative reasoning: Concurrent game models with payoffs and guards

Concurrent game model with payoffs and guards (GGMPG): extend concurrent game models by associating with every state a strategic game with payoffs. Thus:

– at every state each player chooses an action; all actions are applied simultaneously and determine transition to successor state;
– the collective action also determines each player’s payoff;
– same happens at the successor state, etc., thus eventually generating an infinite play;

So, players accumulate utilities in the course of the play;

The players’ current utility values determine their available actions at the current state, by means of guards – arithmetical constraints over the current utilities.
Towards quantitative reasoning: Concurrent game models with payoffs and guards

Concurrent game model with payoffs and guards (GGMPG): extend concurrent game models by associating with every state a strategic game with payoffs. Thus:

– at every state each player chooses an action; all actions are applied simultaneously and determine transition to successor state;
– the collective action also determines each player’s payoff;
– same happens at the successor state, etc., thus eventually generating an infinite play;

So, players accumulate utilities in the course of the play;

The players’ current utility values determine their available actions at the current state, by means of guards – arithmetical constraints over the current utilities.

CGMPGs: games with qualitative and quantitative objectives.
The guards for both players are defined at each state so that the player may:
The guards for both players are defined at each state so that the player may:

- apply any action if she has a positive current accumulated utility,
Guarded concurrent game model with payoffs: example

The guards for both players are defined at each state so that the player may:

- apply any action if she has a positive current accumulated utility,
- only apply action C if she has accumulated utility 0,
Guarded concurrent game model with payoffs: example

The guards for both players are defined at each state so that the player may:

- apply any action if she has a positive current accumulated utility,
- only apply action $C$ if she has accumulated utility 0,
- must play an action maximizing her minimum payoff in the current game if she has a negative accumulated utility.
Configurations, plays and histories in a GCGMP

Configuration in \( M = ( S, \text{payoff}, \{ g_a \}_{a \in A}, \{ d_a \}_{a \in A}) \):

- A pair \((s, \vec{u})\) of a state \(s\) and a vector \(\vec{u} = (u_1, ..., u_k)\) of currently accumulated utilities of the agents at that state.

The set of possible configurations: \( \text{Con}(M) = S \times D \mid |A| \).

Partial configuration transition function: \( \hat{\text{out}} : \text{Con}(M) \times \text{Act} \rightarrow \text{Con}(M) \) where \( \hat{\text{out}}((s, \vec{u}), \vec{\alpha}) = (s', \vec{u'}\) iff:

1. \( \text{out}(s, \vec{u}, \vec{\alpha}) = s' \)
2. The value \( u_a \) assigned to \( v_a \) satisfies \( g_a(s, \alpha_a) \) for each \( a \in A \)
3. \( u'_a = u_a + \text{payoff}_a(s, \vec{u}, \vec{\alpha}) \) for each \( a \in A \)

The configuration graph on \( M \) with an initial configuration \((s_0, \vec{u}_0)\) consists of all configurations in \( M \) reachable from \((s_0, \vec{u}_0)\) by \( \hat{\text{out}} \).

A play in \( M \): an infinite sequence \( \pi = c_0 \vec{\alpha}_0, c_1 \vec{\alpha}_1, ... \) from \((\text{Con}(M) \times \text{Act})^\omega\) such that \( c_n \in \hat{\text{out}}(c_{n-1}, \vec{\alpha}_{n-1}) \) for all \( n > 0 \).

A history: any finite initial sequence of a play in \( \text{Plays}(M) \).
Configurations, plays and histories in a GCGMP

Configuration in $\mathcal{M} = (S, \text{payoff}, \{g_a\}_{a \in A}, \{d_a\}_{a \in A})$: a pair $(s, \overrightarrow{u})$ of a state $s$ and a vector $\overrightarrow{u} = (u_1, \ldots, u_k)$ of currently accumulated utilities of the agents at that state.
Configurations, plays and histories in a GCGMP

**Configuration in** $\mathcal{M} = (S, \text{payoff}, \{g_a\}_{a \in A}, \{d_a\}_{a \in A})$: a pair $(s, \overrightarrow{u})$ of a state $s$ and a vector $\overrightarrow{u} = (u_1, \ldots, u_k)$ of currently accumulated utilities of the agents at that state.

The set of possible configurations: $\text{Con}(\mathcal{M}) = S \times D^{|A|}$. 
Configurations, plays and histories in a GCGMP

Configuration in $\mathcal{M} = (S, \text{payoff}, \{g_a\}_{a \in A}, \{d_a\}_{a \in A})$: a pair $(s, \underline{u})$ of a state $s$ and a vector $\underline{u} = (u_1, \ldots, u_k)$ of currently accumulated utilities of the agents at that state.

The set of possible configurations: $\text{Con}(\mathcal{M}) = S \times D^{|A|}$.

Partial configuration transition function:

$$\hat{\text{out}} : \text{Con}(\mathcal{M}) \times \text{Act}_A \rightarrow \text{Con}(\mathcal{M})$$
Configurations, plays and histories in a GCGMP

Configuration in $\mathcal{M} = (S, \text{payoff}, \{g_a\}_{a \in A}, \{d_a\}_{a \in A})$: a pair $(s, \overrightarrow{u})$ of a state $s$ and a vector $\overrightarrow{u} = (u_1, \ldots, u_k)$ of currently accumulated utilities of the agents at that state.

The set of possible configurations: $\text{Con}(\mathcal{M}) = S \times D^{|A|}$.

Partial configuration transition function:

$$\widehat{\text{out}} : \text{Con}(\mathcal{M}) \times \text{Act}_A \longrightarrow \text{Con}(\mathcal{M})$$

where $\widehat{\text{out}}((s, \overrightarrow{u}), \alpha) = (s', \overrightarrow{u}')$ iff:
Configurations, plays and histories in a GCGMP

Configuration in $\mathcal{M} = (S, \text{payoff}, \{g_a\}_{a \in A}, \{d_a\}_{a \in A})$: a pair $(s, \overrightarrow{u})$ of a state $s$ and a vector $\overrightarrow{u} = (u_1, \ldots, u_k)$ of currently accumulated utilities of the agents at that state.

The set of possible configurations: $\text{Con}(\mathcal{M}) = S \times D^{|A|}$.

Partial configuration transition function:

$$\widehat{\text{out}} : \text{Con}(\mathcal{M}) \times \text{Act}_A \longrightarrow \text{Con}(\mathcal{M})$$

where $\widehat{\text{out}}((s, \overrightarrow{u}), \overrightarrow{\alpha}) = (s', \overrightarrow{u'})$ iff:

(i) $\text{out}(s, \overrightarrow{\alpha}) = s'$
Configurations, plays and histories in a GCGMP

**Configuration** in $\mathcal{M} = (S, \text{payoff}, \{g_a\}_{a \in A}, \{d_a\}_{a \in A})$: a pair $(s, \vec{u})$ of a state $s$ and a vector $\vec{u} = (u_1, \ldots, u_k)$ of currently accumulated utilities of the agents at that state.

The set of possible configurations: $\text{Con}(\mathcal{M}) = S \times D^{|A|}$.

Partial configuration transition function:

$$\widehat{\text{out}} : \text{Con}(\mathcal{M}) \times \text{Act}_A \longrightarrow \text{Con}(\mathcal{M})$$

where $\widehat{\text{out}}((s, \vec{u}), \vec{\alpha}) = (s', \vec{u}')$ iff:

(i) $\text{out}(s, \vec{\alpha}) = s'$

(ii) the value $u_a$ assigned to $v_a$ satisfies $g_a(s, \alpha_a)$ for each $a \in A$
Configurations, plays and histories in a GCGMP

**Configuration in** $\mathcal{M} = (S, \text{payoff}, \{g_a\}_{a \in A}, \{d_a\}_{a \in A})$: a pair $(s, \vec{u})$ of a state $s$ and a vector $\vec{u} = (u_1, \ldots, u_k)$ of currently accumulated utilities of the agents at that state.

The set of possible configurations: $\text{Con}(\mathcal{M}) = S \times \mathbb{D}^{|A|}$.

**Partial configuration transition function:**

$$\hat{\text{out}} : \text{Con}(\mathcal{M}) \times \text{Act}_\mathbb{D} \rightarrow \text{Con}(\mathcal{M})$$

where $\hat{\text{out}}((s, \vec{u}), \vec{\alpha}) = (s', \vec{u}')$ iff:

(i) $\text{out}(s, \vec{\alpha}) = s'$

(ii) the value $u_a$ assigned to $\nu_a$ satisfies $g_a(s, \alpha_a)$ for each $a \in A$

(iii) $u'_a = u_a + \text{payoff}_a(s, \vec{\alpha})$ for each $a \in A$
Configurations, plays and histories in a GCGMP

Configuration in $\mathcal{M} = (S, \text{payoff}, \{g_a\}_{a \in A}, \{d_a\}_{a \in A})$: a pair $(s, \overrightarrow{u})$ of a state $s$ and a vector $\overrightarrow{u} = (u_1, \ldots, u_k)$ of currently accumulated utilities of the agents at that state.

The set of possible configurations: $\text{Con}(\mathcal{M}) = S \times D^{\lvert A \rvert}$.

Partial configuration transition function:

$$\widehat{\text{out}} : \text{Con}(\mathcal{M}) \times \text{Act}_A \rightarrow \text{Con}(\mathcal{M})$$

where $\widehat{\text{out}}((s, \overrightarrow{u}), \overrightarrow{\alpha}) = (s', \overrightarrow{u'})$ iff:

(i) $\text{out}(s, \overrightarrow{\alpha}) = s'$

(ii) the value $u_a$ assigned to $\nu_a$ satisfies $g_a(s, \alpha_a)$ for each $a \in A$

(iii) $u'_a = u_a + \text{payoff}_a(s, \overrightarrow{\alpha})$ for each $a \in A$

The configuration graph on $\mathcal{M}$ with an initial configuration $(s_0, \overrightarrow{u_0})$ consists of all configurations in $\mathcal{M}$ reachable from $(s_0, \overrightarrow{u_0})$ by $\widehat{\text{out}}.$
Configurations, plays and histories in a GCGMP

Configuration in $\mathcal{M} = (S, \text{payoff}, \{g_a\}_{a \in A}, \{d_a\}_{a \in A})$: a pair $(s, \overrightarrow{u})$ of a state $s$ and a vector $\overrightarrow{u} = (u_1, \ldots, u_k)$ of currently accumulated utilities of the agents at that state.

The set of possible configurations: $\text{Con}(\mathcal{M}) = S \times D^{|A|}$.

Partial configuration transition function:

$$\widehat{\text{out}} : \text{Con}(\mathcal{M}) \times \text{Act}_A \rightarrow \text{Con}(\mathcal{M})$$

where $\widehat{\text{out}}((s, \overrightarrow{u}), \overrightarrow{\alpha}) = (s', \overrightarrow{u'})$ iff:

(i) $\text{out}(s, \overrightarrow{\alpha}) = s'$

(ii) the value $u_a$ assigned to $v_a$ satisfies $g_a(s, \alpha_a)$ for each $a \in A$

(iii) $u'_a = u_a + \text{payoff}_a(s, \overrightarrow{\alpha})$ for each $a \in A$

The configuration graph on $\mathcal{M}$ with an initial configuration $(s_0, \overrightarrow{u_0})$ consists of all configurations in $\mathcal{M}$ reachable from $(s_0, \overrightarrow{u_0})$ by $\widehat{\text{out}}$.

A play in $\mathcal{M}$: an infinite sequence $\pi = c_0\overrightarrow{\alpha_0}, c_1\overrightarrow{\alpha_1}, \ldots$ from $(\text{Con}(\mathcal{M}) \times \text{Act})^{\omega}$ such that $c_n \in \widehat{\text{out}}(c_{n-1}, \overrightarrow{\alpha}_{n-1})$ for all $n > 0$. 
Configurations, plays and histories in a GCGMP

Configuration in $\mathcal{M} = (\mathcal{S}, \text{payoff}, \{g_a\}_{a \in A}, \{d_a\}_{a \in A})$: a pair $(s, \overrightarrow{u})$ of a state $s$ and a vector $\overrightarrow{u} = (u_1, \ldots, u_k)$ of currently accumulated utilities of the agents at that state.

The set of possible configurations: $\text{Con}(\mathcal{M}) = \mathcal{S} \times \mathcal{D}^{|A|}$.

Partial configuration transition function:

$$\hat{\text{out}} : \text{Con}(\mathcal{M}) \times \text{Act}_A \longrightarrow \text{Con}(\mathcal{M})$$

where $\hat{\text{out}}((s, \overrightarrow{u}), \overrightarrow{\alpha}) = (s', \overrightarrow{u'})$ iff:

(i) $\text{out}(s, \overrightarrow{\alpha}) = s'$

(ii) the value $u_a$ assigned to $v_a$ satisfies $g_a(s, \alpha_a)$ for each $a \in A$

(iii) $u'_a = u_a + \text{payoff}_a(s, \overrightarrow{\alpha})$ for each $a \in A$

The configuration graph on $\mathcal{M}$ with an initial configuration $(s_0, \overrightarrow{u_0})$ consists of all configurations in $\mathcal{M}$ reachable from $(s_0, \overrightarrow{u_0})$ by $\hat{\text{out}}$.

A play in $\mathcal{M}$: an infinite sequence $\pi = c_0\overrightarrow{\alpha_0}, c_1\overrightarrow{\alpha_1}, \ldots$ from $(\text{Con}(\mathcal{M}) \times \text{Act})^\omega$ such that $c_n \in \hat{\text{out}}(c_{n-1}, \overrightarrow{\alpha}_{n-1})$ for all $n > 0$.

A history: any finite initial sequence of a play in $\text{Plays}_\mathcal{M}$. 
Strategies

A strategy of a player $a$ is a function $s^a : \text{Hist} \rightarrow \text{Act}$ that respects the guards, i.e., if $s^a(h) = \alpha$ then $h[u]_{\text{last}} a| = g^a(h[s^a[\text{last}], \alpha])$.

NB: strategies are based on histories of configurations and actions. Some natural restrictions: state-, action-, or configuration-based; memoryless, bounded memory, of perfect recall strategies.

We assume that two classes of strategies $S^p$ and $S^o$ are fixed as parameters, resp. for the proponents and opponents to select from.

A unique outcome play $M(c, (s^A, s^A \setminus A))$ emerges from the execution of any strategy profile $(s^A, s^A \setminus A)$ from configuration $c$.

Effective strategies: bounded memory strategies determined by transducers with transitions and outputs defined by arithmetical constraints on the current configurations.
A strategy of a player $a$ is a function $s_a : \text{Hist} \rightarrow \text{Act}$ that respects the guards, i.e., if $s_a(h) = \alpha$ then $h^u[\text{last}]_a \models g_a(h^s[\text{last}], \alpha)$. 
A strategy of a player \( a \) is a function \( s_a : \text{Hist} \to \text{Act} \) that respects the guards, i.e., if \( s_a(h) = \alpha \) then \( h^u[\text{last}]_a \models g_a(h^s[\text{last}], \alpha) \).

NB: strategies are based on histories of configurations and actions.
A strategy of a player $a$ is a function $s_a : \text{Hist} \rightarrow \text{Act}$ that respects the guards, i.e., if $s_a(h) = \alpha$ then $h^u[\text{last}]_a \models g_a(h^s[\text{last}], \alpha)$.

NB: strategies are based on histories of configurations and actions.

Some natural restrictions: state-, action-, or configuration-based; memoryless, bounded memory, of perfect recall strategies.
A strategy of a player \(a\) is a function \(s_a : \text{Hist} \rightarrow \text{Act}\) that respects the guards, i.e., if \(s_a(h) = \alpha\) then \(h^u[\text{last}]_a \models g_a(h^s[\text{last}], \alpha)\).

NB: strategies are based on histories of configurations and actions.

Some natural restrictions: state-, action-, or configuration-based; memoryless, bounded memory, of perfect recall strategies.

We assume that two classes of strategies \(S^p\) and \(S^o\) are fixed as parameters, resp. for the proponents and opponents to select from.
Strategies

A strategy of a player $a$ is a function $s_a : \text{Hist} \to \text{Act}$ that respects the guards, i.e., if $s_a(h) = \alpha$ then $h^u[\text{last}]_a \models g_a(h^s[\text{last}], \alpha)$.

NB: strategies are based on histories of configurations and actions.

Some natural restrictions: state-, action-, or configuration-based; memoryless, bounded memory, of perfect recall strategies.

We assume that two classes of strategies $S^p$ and $S^o$ are fixed as parameters, resp. for the proponents and opponents to select from.

A unique outcome $\text{play}_M(c, (s_A, s_{A\setminus A}))$ emerges from the execution of any strategy profile $(s_A, s_{A\setminus A})$ from configuration $c$. 
A strategy of a player $a$ is a function $s_a : \text{Hist} \rightarrow \text{Act}$ that respects the guards, i.e., if $s_a(h) = \alpha$ then $h^u[\text{last}]_a \models g_a(h^s[\text{last}], \alpha)$.

NB: strategies are based on histories of configurations and actions.

Some natural restrictions: state-, action-, or configuration-based; memoryless, bounded memory, or perfect recall strategies.

We assume that two classes of strategies $S^p$ and $S^o$ are fixed as parameters, resp. for the proponents and opponents to select from.

A unique outcome $\text{play}_M(c, (s_A, s_{A\setminus A}))$ emerges from the execution of any strategy profile $(s_A, s_{A\setminus A})$ from configuration $c$.

Effective strategies: bounded memory strategies determined by transducers with transitions and outputs defined by arithmetical constraints on the current configurations.
QATL*: Quantitative extension of ATL*

Language AC of arithmetic formulae over accumulated utilities:
Boolean combinations of equalities and inequalities between terms built by applying addition over a set of variables \( V = \{ v | a \in A \} \) for the accumulated utilities and a fixed set \( X \) of constants.

Language of QATL*. Extends ATL* with formulae from AC:

State formulae \( \varphi ::= p | ac | \neg \varphi | \varphi \land \varphi | \langle\langle A \rangle\rangle \gamma \)

Path formulae:
\( \gamma ::= \varphi | \neg \gamma | \gamma \land \gamma | X \gamma | G \gamma | \gamma U \gamma \)

where \( A \subseteq A, ac \in AC \) and \( p \in \text{Prop.} \)

An extension: with arithmetic formulae over entire plays. Requires adding discounting factors on payoffs. Will not be discussed here.
Language AC of arithmetic formulae over accumulated utilities: Boolean combinations of equalities and inequalities between terms built by applying addition over a set of variables $V_\mathcal{A} = \{v_a \mid a \in \mathcal{A}\}$ for the accumulated utilities and a fixed set $X$ of constants.
QATL*: Quantitative extension of ATL*

Language **AC** of arithmetic formulae over accumulated utilities: Boolean combinations of equalities and inequalities between terms built by applying addition over a set of variables $V_A = \{ v_a | a \in A \}$ for the accumulated utilities and a fixed set $X$ of constants.

**Language of QATL***. Extends ATL* with formulae from AC:
QATL*: Quantitative extension of ATL*

Language **AC** of arithmetic formulae over accumulated utilities: Boolean combinations of equalities and inequalities between terms built by applying addition over a set of variables $V_A = \{v_a \mid a \in A\}$ for the accumulated utilities and a fixed set $X$ of constants.

**Language of QATL***. Extends ATL* with formulae from AC:

State formulae $\varphi ::= p \mid ac \mid \neg \varphi \mid \varphi \land \varphi \mid \langle A \rangle \gamma$

Path formulae: $\gamma ::= \varphi \mid \neg \gamma \mid \gamma \land \gamma \mid X \gamma \mid G \gamma \mid \gamma U \gamma$ where $A \subseteq A$, $ac \in AC$ and $p \in Prop$. An extension: with arithmetic formulae over entire plays. Requires adding discounting factors on payoffs. Will not be discussed here.
QATL*: Quantitative extension of ATL*

Language $\mathbf{AC}$ of arithmetic formulae over accumulated utilities: Boolean combinations of equalities and inequalities between terms built by applying addition over a set of variables $V_A = \{v_a \mid a \in A\}$ for the accumulated utilities and a fixed set $X$ of constants.

Language of QATL*. Extends ATL* with formulae from AC:

State formulae $\varphi ::= p \mid \text{ac} \mid \neg \varphi \mid \varphi \land \varphi \mid \langle A \rangle \gamma$

Path formulae: $\gamma ::= \varphi \mid \neg \gamma \mid \gamma \land \gamma \mid X\gamma \mid G\gamma \mid U\gamma$

Valentin Goranko
QATL*: Quantitative extension of ATL*

Language AC of arithmetic formulae over accumulated utilities: Boolean combinations of equalities and inequalities between terms built by applying addition over a set of variables $V_{\mathcal{A}} = \{ v_a | a \in \mathcal{A} \}$ for the accumulated utilities and a fixed set $\mathcal{X}$ of constants.

Language of QATL*. Extends ATL* with formulae from AC:

State formulae $\varphi ::= p \mid ac \mid \neg \varphi \mid \varphi \land \varphi \mid \langle\langle \mathcal{A}\rangle\rangle \gamma$

Path formulae: $\gamma ::= \varphi \mid \neg \gamma \mid \gamma \land \gamma \mid X\gamma \mid G\gamma \mid \gamma U\gamma$

where $\mathcal{A} \subseteq \mathcal{A}$, $ac \in AC$ and $p \in Prop$. 
QATL*: Quantitative extension of ATL*

Language $AC$ of arithmetic formulae over accumulated utilities:
Boolean combinations of equalities and inequalities between terms
built by applying addition over a set of variables $V_A = \{v_a \mid a \in A\}$
for the accumulated utilities and a fixed set $X$ of constants.

Language of QATL*. Extends ATL* with formulae from AC:

State formulae $\varphi ::= p \mid ac \mid \neg \varphi \mid \varphi \land \varphi \mid \langle\langle A\rangle\rangle\gamma$

Path formulae: $\gamma ::= \varphi \mid \neg \gamma \mid \gamma \land \gamma \mid X\gamma \mid G\gamma \mid U\gamma$
where $A \subseteq A$, $ac \in AC$ and $p \in Prop$.

An extension: with arithmetic formulae over entire plays. Requires
adding discounting factors on payoffs. Will not be discussed here.
Semantics of QATL*
Semantics of QATL*

Given: $\mathcal{M}$ be a GCGMP, $c$ a configuration, $\varphi$ state formula, $\gamma, \gamma_1, \gamma_2$ path formulae, $S^p$ and $S^o$ two classes of strategies.
Semantics of QATL*

Given: $\mathcal{M}$ be a GCGMP, $c$ a configuration, $\varphi$ state formula, $\gamma, \gamma_1, \gamma_2$ path formulae, $S^p$ and $S^o$ two classes of strategies.

$\mathcal{M}, c \models p$ iff $p \in L(c^s)$;
Semantics of QATL*

Given: $M$ be a GCGMP, $c$ a configuration, $\varphi$ state formula, $\gamma, \gamma_1, \gamma_2$ path formulae, $S^p$ and $S^o$ two classes of strategies.

$M, c \models p$ iff $p \in L(c^s)$;

$M, c \models ac$ iff $c^u \models ac,$
Semantics of QATL*

Given: $M$ be a GCGMP, $c$ a configuration, $\varphi$ state formula, $\gamma, \gamma_1, \gamma_2$ path formulae, $S^p$ and $S^o$ two classes of strategies.

$M, c \models p$ iff $p \in L(c^s)$;

$M, c \models ac$ iff $c^u \models ac$,

$M, c \models \langle \langle A \rangle \rangle \gamma$ iff there is a $S^p$-strategy $s_A$ such that for all $S^o$-strategies $s_{A\setminus A}$: $M$, outcome-play$^M(c, (s_A, s_{A\setminus A})) \models \gamma$. 

Valentin Goranko
Semantics of QATL*

Given: $\mathcal{M}$ be a GCGMP, $c$ a configuration, $\varphi$ state formula, $\gamma, \gamma_1, \gamma_2$ path formulae, $S^p$ and $S^o$ two classes of strategies.

$\mathcal{M}, c \models p$ iff $p \in L(c^s)$;
$\mathcal{M}, c \models ac$ iff $c^u \models ac$,
$\mathcal{M}, c \models \langle\langle A\rangle\rangle\gamma$ iff there is a $S^p$-strategy $s_A$ such that for all $S^o$-strategies $s_{A\setminus A}$: $\mathcal{M}, \text{outcome}_{\text{play}}^\mathcal{M}(c, (s_A, s_{A\setminus A})) \models \gamma$.
$\mathcal{M}, \pi \models \varphi$ iff $\mathcal{M}, \pi[0] \models \varphi,$
Semantics of QATL*

Given: $\mathcal{M}$ be a GCGMP, $c$ a configuration, $\varphi$ state formula, $\gamma, \gamma_1, \gamma_2$ path formulae, $S^p$ and $S^o$ two classes of strategies.

$\mathcal{M}, c \models p$ iff $p \in L(c^s)$;

$\mathcal{M}, c \models ac$ iff $c^u \models ac$,

$\mathcal{M}, c \models \langle A \rangle \gamma$ iff there is a $S^p$-strategy $s_A$ such that for all $S^o$-strategies $s_A \backslash A$: $\mathcal{M}, \text{outcome}_\text{play}^\mathcal{M}(c, (s_A, s_A \backslash A)) \models \gamma$.

$\mathcal{M}, \pi \models \varphi$ iff $\mathcal{M}, \pi[0] \models \varphi$,

$\mathcal{M}, \pi \models \mathcal{X}\gamma$ iff $\mathcal{M}, \pi[1] \models \gamma$. 

Valentin Goranko
Semantics of QATL*

Given: $\mathcal{M}$ be a GCGMP, $c$ a configuration, $\varphi$ state formula, $\gamma, \gamma_1, \gamma_2$ path formulae, $S^p$ and $S^o$ two classes of strategies.

$\mathcal{M}, c \models p$ iff $p \in L(c^s)$;
$\mathcal{M}, c \models ac$ iff $c^u \models ac$,
$\mathcal{M}, c \models \langle A \rangle \gamma$ iff there is a $S^p$-strategy $s_A$ such that for all $S^o$-strategies $s_A \backslash A$: $\mathcal{M}$, outcome\_play$^m(c, (s_A, s_A \backslash A)) \models \gamma$.

$\mathcal{M}, \pi \models \varphi$ iff $\mathcal{M}, \pi[0] \models \varphi$,
$\mathcal{M}, \pi \models X\gamma$ iff $\mathcal{M}, \pi[1] \models \gamma$,
$\mathcal{M}, \pi \models G\gamma$ iff $\mathcal{M}, \pi[i] \models \gamma$ for all $i \in \mathbb{N}$,
Semantics of QATL*

Given: $M$ be a GCGMP, $c$ a configuration, $\varphi$ state formula, $\gamma, \gamma_1, \gamma_2$ path formulae, $S^p$ and $S^o$ two classes of strategies.

$M, c \models p$ iff $p \in L(c^s)$;

$M, c \models ac$ iff $c^u \models ac$,

$M, c \models \langle A \rangle \gamma$ iff there is a $S^p$-strategy $s_A$ such that for all $S^o$-strategies $s_{A\setminus A}$: $M, \text{outcome}_p^M((c, (s_A, s_{A\setminus A}))) \models \gamma$.

$M, \pi \models \varphi$ iff $M, \pi[0] \models \varphi$,

$M, \pi \models X \gamma$ iff $M, \pi[1] \models \gamma$,

$M, \pi \models G \gamma$ iff $M, \pi[i] \models \gamma$ for all $i \in \mathbb{N}$,

$M, \pi \models \gamma_1 U \gamma_2$ iff there is $j \in \mathbb{N}_0$ such that $M, \pi[j] \models \gamma_2$ and $M, \pi[i] \models \gamma_1$ for all $0 \leq i < j$. 

Valentin Goranko
Semantics of QATL* 

Given: $M$ be a GCGMP, $c$ a configuration, $\varphi$ state formula, $\gamma, \gamma_1, \gamma_2$ path formulae, $S^p$ and $S^o$ two classes of strategies.

$M, c \models p$ iff $p \in L(c^s)$;
$M, c \models ac$ iff $c^u \models ac$,

$M, c \models \langle A \rangle \gamma$ iff there is a $S^p$-strategy $s_A$ such that for all $S^o$-strategies $s_A \backslash A$: $M$, outcome-play$^M(c, (s_A, s_A \backslash A)) \models \gamma$.

$M, \pi \models \varphi$ iff $M, \pi[0] \models \varphi$,

$M, \pi \models X \gamma$ iff $M, \pi[1] \models \gamma$,

$M, \pi \models G \gamma$ iff $M, \pi[i] \models \gamma$ for all $i \in \mathbb{N}$,

$M, \pi \models \gamma_1 U \gamma_2$ iff there is $j \in \mathbb{N}_0$ such that $M, \pi[j] \models \gamma_2$ and $M, \pi[i] \models \gamma_1$ for all $0 \leq i < j$.

Ultimately, we define $M, c \models \varphi$ iff $M, c, 0 \models \varphi$. 
Expressing properties in QATL*
Expressing properties in QATL*

▷ QATL* extends ATL*, so it can express all purely qualitative ATL* properties,
Expressing properties in QATL*

▷ QATL* extends ATL*, so it can express all purely qualitative ATL* properties, like

$$\langle A \rangle (G p \land qU r)$$
Expressing properties in QATL*

▷ QATL* extends ATL*, so it can express all purely qualitative ATL* properties, like

\[\llangle A\rrangle (Gp \land qU r)\]

▷ QATL* can also express quantitative properties,
Expressing properties in QATL*

➢ QATL* extends ATL*, so it can express all purely qualitative ATL* properties, like

\( \langle A \rangle (G p \land q U r) \)

➢ QATL* can also express quantitative properties, e.g.:

\( \langle\{a\}\rangle G (v_a > 0) \)

“Player a has a strategy to maintain his accumulated utility positive”,
Expressing properties in QATL*

▷ QATL* extends ATL*, so it can express all purely qualitative ATL* properties, like

\[ \langle A \rangle (G p \land q \land r) \]

▷ QATL* can also express quantitative properties, e.g.:

\[ \langle \{ a \} \rangle G (v_a > 0) \]

“Player \( a \) has a strategy to maintain his accumulated utility positive”,

▷ Moreover, QATL* can naturally express combined qualitative and quantitative properties,
Expressing properties in QATL*

- QATL* extends ATL*, so it can express all purely qualitative ATL* properties, like
  \[ \langle A \rangle (Gp \land qU r) \]

- QATL* can also express quantitative properties, e.g.:
  \[ \langle \{ a \} \rangle G (v_a > 0) \]

“Player a has a strategy to maintain his accumulated utility positive”,

- Moreover, QATL* can naturally express combined qualitative and quantitative properties, e.g.
  \[ \langle \{ a \} \rangle ((a \text{ is happy}) U (v_a \geq 10^6)) \]
Expressing properties in QATL*

- QATL* extends ATL*, so it can express all purely qualitative ATL* properties, like:
  \[ \langle A \rangle (Gp \land qU r) \]

- QATL* can also express quantitative properties, e.g.:
  \[ \langle \{a\} \rangle G(v_a > 0) \]

"Player a has a strategy to maintain his accumulated utility positive",

- Moreover, QATL* can naturally express combined qualitative and quantitative properties, e.g.:
  \[ \langle \{a\} \rangle ((a\text{ is happy}) U (v_a \geq 10^6)) \]

"Player a has a strategy to reach accumulated utility of one million and meanwhile stay in “happy” states."
Expressing properties in QATL*: more examples

In the examples below $p_i$ is true only at $s_i$, for each $i = 1, 2, 3$. 

<table>
<thead>
<tr>
<th>$I$</th>
<th>$II$</th>
<th>$C$</th>
<th>$D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C$</td>
<td>4, 3</td>
<td>0, 2</td>
<td></td>
</tr>
<tr>
<td>$D$</td>
<td>−1, −2</td>
<td>2, 3</td>
<td></td>
</tr>
</tbody>
</table>

**Battle of Sexes**

<table>
<thead>
<tr>
<th>$I$</th>
<th>$II$</th>
<th>$C$</th>
<th>$D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C$</td>
<td>2, 2</td>
<td>−3, 3</td>
<td></td>
</tr>
<tr>
<td>$D$</td>
<td>3, −3</td>
<td>−1, −1</td>
<td></td>
</tr>
</tbody>
</table>

**Prisoners Dilemma**

<table>
<thead>
<tr>
<th>$I$</th>
<th>$II$</th>
<th>$C$</th>
<th>$D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C$</td>
<td>1, 1</td>
<td>−1, −1</td>
<td></td>
</tr>
<tr>
<td>$D$</td>
<td>−1, −1</td>
<td>1, 1</td>
<td></td>
</tr>
</tbody>
</table>
Expressing properties in QATL*: more examples

In the examples below $p_i$ is true only at $s_i$, for each $i = 1, 2, 3$.

1. $\langle\langle I, II \rangle\rangle F (p_1 \land v_I > 100 \land v_{II} > 100)$
Expressing properties in QATL*: more examples

In the examples below $p_i$ is true only at $s_i$, for each $i = 1, 2, 3$.

1. $\langle\{I, II\}\rangle F(p_1 \land v_I > 100 \land v_{II} > 100)$
2. $\langle\{I, II\}\rangle X X \langle\{II\}\rangle (G(p_2 \land v_I = 0) \land F v_{II} > 100)$. 
Expressing properties in QATL*: more examples

In the examples below \( p_i \) is true only at \( s_i \), for each \( i = 1, 2, 3 \).

1. \( \langle\langle \{I, II\} \rangle\rangle F (p_1 \land v_I > 100 \land v_{II} > 100) \)
2. \( \langle\langle \{I, II\} \rangle\rangle X X \langle\langle \{II\} \rangle\rangle (G(p_2 \land v_I = 0) \land F v_{II} > 100) \).
3. \( \neg \langle\langle \{I\} \rangle\rangle G (p_1 \lor v_I > 0) \)
Expressing properties in QATL*: more examples

In the examples below \( p_i \) is true only at \( s_i \), for each \( i = 1, 2, 3 \).

1. \( \langle\langle \{ I, II\} \rangle\rangle F(p_1 \land v_I > 100 \land v_{II} > 100) \)
2. \( \langle\langle \{ I, II\} \rangle\rangle X X \langle\langle \{ II\} \rangle\rangle (G(p_2 \land v_I = 0) \land F v_{II} > 100) \).
3. \( \neg \langle\langle \{ I\} \rangle\rangle G(p_1 \lor v_I > 0) \)
4. \( \neg \langle\langle \{ I, II\} \rangle\rangle F(p_3 \land G(p_3 \land v_I + v_{II} > 0)) \).
Some undecidability results about QATL*
Some undecidability results about QATL*

The framework is very general and easily leads to undecidable MC.
Some undecidability results about QATL*

The framework is very general and easily leads to undecidable MC.

**Lemma** (Reduction from the Halting problem for Minsky machines)
For any Minsky machine (2-counter automaton) $A$ a finite 2-player GCGMP $M^A$ using a proposition $\text{halt}$ can be constructed so that:

A halts on empty input iff
there is a play $\pi$ in $M^A$ which reaches a $\text{halt}$-state.
Some undecidability results about QATL*

The framework is very general and easily leads to undecidable MC.

**Lemma** (Reduction from the Halting problem for Minsky machines)
For any Minsky machine (2-counter automaton) \( A \) a finite 2-player GCGMP \( \mathcal{M}^A \) using a proposition \( \text{halt} \) can be constructed so that:

A halts on empty input iff
there is a play \( \pi \) in \( \mathcal{M}^A \) which reaches a \( \text{halt} \)-state.

**Thm** Model checking in the logic QATL* is undecidable, even for the fragment with no nested cooperation modalities, where \( S^p = S^{\text{mem}} \) and \( S^o = S^{\text{pos}} \), in each of the following cases:
Some undecidability results about QATL*

The framework is very general and easily leads to undecidable MC.

**Lemma** (Reduction from the Halting problem for Minsky machines)
For any Minsky machine (2-counter automaton) $A$ a finite 2-player GCGMP $M^A$ using a proposition $\text{halt}$ can be constructed so that:

A halts on empty input iff
there is a play $\pi$ in $M^A$ which reaches a halt-state.

**Thm** Model checking in the logic QATL* is undecidable, even for the fragment with no nested cooperation modalities, where $S^p = S^{mem}$ and $S^o = S^{pos}$, in each of the following cases:

1. Two players, no arithmetic constraints in the formula.
Some undecidability results about QATL*

The framework is very general and easily leads to undecidable MC.

**Lemma** (Reduction from the Halting problem for Minsky machines)
For any Minsky machine (2-counter automaton) $A$ a finite 2-player GCGMP $\mathcal{M}^A$ using a proposition $\text{halt}$ can be constructed so that:

A halts on empty input iff there is a play $\pi$ in $\mathcal{M}^A$ which reaches a $\text{halt}$-state.

**Thm** Model checking in the logic QATL* is undecidable, even for the fragment with no nested cooperation modalities, where $S^p = S^{mem}$ and $S^o = S^{pos}$, in each of the following cases:

1. Two players, no arithmetic constraints in the formula.
2. Two players, state-based guards.
Some undecidability results about QATL* 

The framework is very general and easily leads to undecidable MC.

**Lemma** (Reduction from the Halting problem for Minsky machines) For any Minsky machine (2-counter automaton) $A$ a finite 2-player GCGMP $\mathcal{M}^A$ using a proposition $\text{halt}$ can be constructed so that:

A halts on empty input iff there is a play $\pi$ in $\mathcal{M}^A$ which reaches a $\text{halt}$-state.

**Thm** Model checking in the logic QATL* is undecidable, even for the fragment with no nested cooperation modalities, where $S^p = S^{\text{mem}}$ and $S^o = S^{\text{pos}}$, in each of the following cases:

1. Two players, no arithmetic constraints in the formula.
2. Two players, state-based guards.
3. Three players, no guards, non-negative payoffs only.
Some decidability results and conjectures about QATL*
Some decidability results and conjectures about QATL* 

Thm: MC in the logic QATL* is decidable in the following cases:
Some decidability results and conjectures about QATL*

**Thm:** MC in the logic QATL* is decidable in the following cases:

1. Many players, all executing bounded memory effective strategies.
Some decidability results and conjectures about QATL*

**Thm:** MC in the logic QATL* is decidable in the following cases:

1. Many players, all executing bounded memory effective strategies.

2. Two-player turn-based GCGMPs, for the fragment with formulae involving only player 1’s accumulated utility.
Some decidability results and conjectures about QATL*

**Thm:** MC in the logic QATL* is decidable in the following cases:

1. Many players, all executing bounded memory effective strategies.

2. Two-player turn-based GCGMPs, for the fragment with formulae involving only player 1’s accumulated utility.

**Conjectures:** Model checking in the logic QATL* is decidable in each of the following cases:
Some decidability results and conjectures about QATL*

**Thm:** MC in the logic QATL* is decidable in the following cases:

1. Many players, all executing bounded memory effective strategies.

2. Two-player turn-based GCGMPs, for the fragment with formulae involving only player 1’s accumulated utility.

**Conjectures:** Model checking in the logic QATL* is decidable in each of the following cases:

1. Two players and non-negative payoffs.
Some decidability results and conjectures about QATL$^*$

**Thm:** MC in the logic QATL$^*$ is decidable in the following cases:

1. Many players, all executing bounded memory effective strategies.

2. Two-player turn-based GCGMPs, for the fragment with formulae involving only player 1’s accumulated utility.

**Conjectures:** Model checking in the logic QATL$^*$ is decidable in each of the following cases:

1. Two players and non-negative payoffs.

2. Many players, no guards, restriction to the quantitative atomic formulae to only allow comparisons between players’ payoffs and constants, i.e. of the type $v_i \circ c$ but not $v_i \circ v_j$, where $\circ \in \{>, \, =, \, <\}$. 
Concluding remarks

We have proposed a logical framework combining qualitative with quantitative reasoning in multi-payer games.
Concluding remarks

We have proposed a logical framework combining qualitative with quantitative reasoning in multi-payer games.

- Three perspectives of research agenda:
  - Logic: Expressiveness, formal reasoning, deduction.
  - Computation: decidability, algorithms and complexity for model checking and synthesis.
  - Game theory: solution concepts, equilibria, extending results from repeated games (e.g., folk theorems), etc.

Many still unexplored directions, including:
  - games with imperfect information,
  - satisfiability testing and model synthesis,
  - stochastic games with probabilistic strategies, etc.

The End
Concluding remarks

We have proposed a logical framework combining qualitative with quantitative reasoning in multi-payer games.

▷ Three perspectives of research agenda:

- **Logic**: Expressiveness, formal reasoning, deduction.
Concluding remarks

We have proposed a logical framework combining qualitative with quantitative reasoning in multi-payer games.

▷ Three perspectives of research agenda:

- **Logic**: Expressiveness, formal reasoning, deduction.
- **Computation**: decidability, algorithms and complexity for model checking and synthesis.
Concluding remarks

We have proposed a logical framework combining qualitative with quantitative reasoning in multi-payer games.

▷ Three perspectives of research agenda:

- **Logic**: Expressiveness, formal reasoning, deduction.
- **Computation**: decidability, algorithms and complexity for model checking and synthesis.
- **Game theory**: solution concepts, equilibria, extending results from repeated games (e.g., folk theorems), etc.
Concluding remarks

We have proposed a logical framework combining qualitative with quantitative reasoning in multi-payer games.

▷ Three perspectives of research agenda:

- **Logic**: Expressiveness, formal reasoning, deduction.
- **Computation**: decidability, algorithms and complexity for model checking and synthesis.
- **Game theory**: solution concepts, equilibria, extending results from repeated games (e.g., folk theorems), etc.

▷ Many still unexplored directions, including:
Concluding remarks

We have proposed a logical framework combining qualitative with quantitative reasoning in multi-payer games.

▷ Three perspectives of research agenda:

- **Logic**: Expressiveness, formal reasoning, deduction.
- **Computation**: decidability, algorithms and complexity for model checking and synthesis.
- **Game theory**: solution concepts, equilibria, extending results from repeated games (e.g., folk theorems), etc.

▷ Many still unexplored directions, including:

- games with imperfect information,
Concluding remarks

We have proposed a logical framework combining qualitative with quantitative reasoning in multi-payer games.

▷ Three perspectives of research agenda:

- **Logic**: Expressiveness, formal reasoning, deduction.

- **Computation**: decidability, algorithms and complexity for model checking and synthesis.

- **Game theory**: solution concepts, equilibria, extending results from repeated games (e.g., folk theorems), etc.

▷ Many still unexplored directions, including:

- games with imperfect information,

- satisfiability testing and model synthesis,
Concluding remarks

We have proposed a logical framework combining qualitative with quantitative reasoning in multi-payer games.

▷ Three perspectives of research agenda:

- **Logic**: Expressiveness, formal reasoning, deduction.
- **Computation**: decidability, algorithms and complexity for model checking and synthesis.
- **Game theory**: solution concepts, equilibria, extending results from repeated games (e.g., folk theorems), etc.

▷ Many still unexplored directions, including:

- games with imperfect information,
- satisfiability testing and model synthesis,
- stochastic games with probabilistic strategies, etc.
Concluding remarks

We have proposed a logical framework combining qualitative with quantitative reasoning in multi-payer games.

▷ Three perspectives of research agenda:

- **Logic**: Expressiveness, formal reasoning, deduction.
- **Computation**: decidability, algorithms and complexity for model checking and synthesis.
- **Game theory**: solution concepts, equilibria, extending results from repeated games (e.g., folk theorems), etc.

▷ Many still unexplored directions, including:

- games with imperfect information,
- satisfiability testing and model synthesis,
- stochastic games with probabilistic strategies, etc.

The End