Games in computer science: a survey

Pierre-Louis Curien

CNRS – Université Paris 7

Globus Seminar, Moscow, April 2005
“Provability” versus “proofs”

• Games to reason about programs

• Programs as strategies
Model-checking 1/2

Satisfiability problem for various logics (modal, temporal, \( \mu \)) for automata or concurrent systems

\[ \iff \]

Existence of winning strategies in associated games.
Model-checking 2/2

Also $\Leftrightarrow$ (non-)emptyness problem for languages recognized by various kinds of automata on (infinite) words or trees.

Also, bisimulation in concurrency theory.
Game semantics 1/2

Strategies as proofs / programs / morphims. Composition corresponds to cut elimination / normalization. Games semantics is very active since a decade.
Game semantics 2/2

New results in the semantics of programming languages: simple and direct semantics for programming features such as control or references,

Full abstraction results connecting denotational and operational semantics tightly.
A theorem on lattices

Joyal (1997) used games to give a nice proof of the following theorem (Whitman 1947): The free lattice $L$ over a partial order $X$ (with $i : X \to L$) is characterized by
- $L$ is a lattice and $i$ is monotonous
- If $u_1 \land u_2 \leq v_1 \lor v_2$, then $u_1 \land u_2 \leq v_1$ or $u_1 \land u_2 \leq v_2$ or $u_1 \leq v_1 \lor v_2$ or $u_2 \leq v_1 \lor v_2$
- If $i(x) \leq v_1 \lor v_2$, then $i(x) \leq v_1$ or $i(x) \leq v_2$
- If $u_1 \land u_2 \leq i(x)$, then $u_1 \leq i(x)$ or $u_2 \leq i(x)$
- If $i(x) \leq i(y)$, then $x \leq y$
- $L$ is generated by $i(X)$
Uniqueness easy. For existence, construct a suitable preorder on the following set of terms:

\[
\frac{x \in X}{x \in T(X)} \quad \frac{V \in T(X)}{F \in T(X)}
\]

\[
\frac{A_1 \in T(X) \quad A_2 \in T(X)}{A_1 \land A_2 \in T(X)} \quad \frac{A_1 \in T(X) \quad A_2 \in T(X)}{A_1 \lor A_2 \in T(X)}
\]
The preorder is defined by: \( A \leq B \) if and only if \((A, B)\) is a \textit{winning position} in some graph game.

The set of nodes is \( T(X) \times T(X) \).
PROLOGUE 5/13

Edges:

\[(A_1 \lor A_2, B) \rightarrow (A_1, B)\]
\[(A, B_1 \land B_2) \rightarrow (A, B_1)\]
\[(A_1 \lor A_2, B) \rightarrow (A_2, B)\]
\[(A, B_1 \land B_2) \rightarrow (A, B_2)\]

\[(A_1 \land A_2, B_1 \lor B_2) \rightarrow (A_1, B_1 \lor B_2)\]
\[(A_1 \land A_2, B_1 \lor B_2) \rightarrow (A_1 \land A_2, B_1)\]
\[(A_1 \land A_2, B_1 \lor B_2) \rightarrow (A_2, B_1 \lor B_2)\]
\[(A_1 \land A_2, B_1 \lor B_2) \rightarrow (A_1 \land A_2, B_2)\]

\[(A_1 \land A_2, F) \rightarrow (A_1, F)\]
\[(V, B_1 \lor B_2) \rightarrow (V, B_1)\]
\[(A_1 \land A_2, F) \rightarrow (A_1 \land A_2, B_1)\]
\[(V, B_1 \lor B_2) \rightarrow (V, B_2)\]

\[(A_1 \land A_2, x) \rightarrow (A_1, x)\]
\[(x, B_1 \lor B_2) \rightarrow (x, B_1)\]
\[(A_1 \land A_2, x) \rightarrow (A_2, x)\]
\[(x, B_1 \lor B_2) \rightarrow (x, B_2)\]
Each node has a polarity $\in \{P, O, N\}$ (Player, Opponent, Neutral).

\[
\begin{align*}
O &= (A_1 \lor A_2, B) \\
O &= (F, B) \\
O &= (A, B_1 \land B_2) \\
O &= (A, V)
\end{align*}
\]

\[
\begin{align*}
P &= (A_1 \land A_2, B_1 \lor B_2) \\
P &= (V, F) \\
P &= (x, B_1 \lor B_2) \\
P &= (x, F) \\
P &= (V, B_1 \lor B_2) \\
P &= (A_1 \land A_2, F) \\
P &= (A_1 \land A_2, x) \\
P &= (V, x) \\
N &= (x, y)
\end{align*}
\]
A strategy is a full subgraph \( S \) s.t.
- If \((A, B) \in S\), then \( S \) contains at least one edge out of \((A, B)\).
- If \((A, B) \in S\), then \( S \) contains all edges of \( G \) out of \((A, B)\).
- If \((x, y) \in S\), then \( x \leq y \) in \( X \).
We say that \((A, B)\) is a winning position if \((A, B)\) belongs to some strategy. We then write \(A \leq B\).
A proof is a strategy which satisfies:

- In the first condition, replace “at least one” by “exactly one”.
- There is a root (an edge from which all other edges can be reached following (oriented) paths of the strategy).
**Lemma 1.** $(A, B)$ is winning iff there is proof rooted in $(A, B)$.

**Lemma 2.** $A_1 \land A_2$ is a greatest lower bound of $A_1$ and $A_2$, etc... .

**Lemma 3.** $\leq$ is transitive.
(1) Easy (induction on formulas)

(2) Use the presentation by proofs

(3) Use the presentation by strategies. The composition of two strategies $S$ and $T$ witnessing $A \leq B$ and $B \leq C$ is:

$$S \circ T = \left\{ (x, z) \mid \exists y \ (x, y) \in S \text{ et } (y, z) \in T \right\}.$$
This example embodies ideas of using games for both
- model-checking (we are interested in the mere existence of strategies for inequality predicates) and
- game semantics: we want a compositional semantics: combine strategies to build other strategies.
The situation proofs / strategies somehow matches the operational / denotational distinction in the semantics of programming languages: Proofs compose by normalization / cut-elimination / interaction, while strategies compose as mathematical functions. (Cf. also functions as relations vs functions as algorithms).
AUTOMATA, LOGICS . . .

Büchi (1962): Two-way correspondence between automata on infinite words and monadic second order logic over infinite words $\alpha$:

$$\forall \alpha \ (\alpha \models \phi \iff A \text{ accepts } \alpha)$$

This logic is decidable.
and GAMES

second order monadic logic \iff automata

\updownarrow parity

parity games

McNaughton, Rabin, Gurevitch-Harrington, Zielonka, Thomas, \ldots
Determinacy

Parity games are determined, and who wins is decidable.

A nice proof of Santocanale goes along the hypothenuse of the above triangle (but the target is a logic of fixed points).
A (partial) game is
- an oriented graph \( G = (G_0, G_1) \)
- the nodes have a polarity (\( \epsilon : G_0 \to \{P, O, N\} \), if \( \epsilon(x) = N \), then \( x \) is terminal)
If \( \epsilon^{-1}(N) = \emptyset \), the game is called total.
Parity automata and fixpoints 2/8

One also gives a set $W_P$ of infinite winning paths for $P$ ($W_O$ is its complement).

Winning strategy for $P$ (resp. $O$) = strategy all of whose infinite paths $\in W_P$ (resp. $\in W_O$). Winning position = belongs to a winning strategy.
Parity automata and fixpoints 3/8

Given \( X \subseteq \varepsilon^{-1}(N) \), given \( S(x) \subseteq G_0 \) and \( OP^x \in \{\land, \lor\} \) for all \( x \in X \), define the games

\[ \mu_S \cdot G[X] \] (short for \( \mu_{S,OP} \cdot G[X] \)) , \( \nu_S \cdot G[X] \):

- add \( x \rightarrow g \) for all \( x \in X, g \in S(x) \),
- change polarity of \( x \in X \) to \( P \) (resp. \( O \)) if \( OP^x = \lor \) (resp. \( OP^x = \land \)).
The two games differ only in the definition of winning:

- $\mu_S.G[X]$: the winning paths of $P$ are those infinite paths in the new graph which eventually are winning for $P$ in the old.
- $\nu_S.G[X]$: (dual) the ... of $O$ in the new graph which eventually ... for $O$ in the old.
Parity automata and fixpoints 5/8

$G[X \cap A]$ defined by changing the polarity of $x \in X$ to $P$ (resp. $O$) if $x \not\in A$ (resp. $x \in A$).

**Lemma 1.** If all games $G[X \cap A]$ are determined, then $\mu_S.G[X]$ (resp. $\nu_S.G[X]$) is determined and its set of winning positions is obtained as a least (resp. greatest) fixed point of a monotonous operator.
A parity game is a (total) game in which the nodes also have a colour \((p : G_0 \rightarrow \{1, \ldots, n\})\) and the colours have a parity \((\chi : \{1, \ldots, n\} \rightarrow \{\text{P}, \text{O}\})\).

\(W_P\) consists of those paths such that if \(m\) is the maximum colour visited infinitely often along the path, then \(\chi(m) = \text{P}\).
Lemma 2. Each parity game $G$ can be written as $Q_{S_n} \cdots Q_{S_1} G_0[X_1] \cdots [X_n]$ where
- $X_i$ is the set of nodes of colour $i$,
- $S_i(x)$ is the set of successors of $x$ in $G$,
- $OP^x = \lor$ (resp. $OP^x = \land$) if $x$ has polarity $P$ (resp. $O$),
- $Q_{S_i} = \mu$ (resp. $Q_{S_i} = \nu$) if $\chi(i) = P$ (resp. $\chi(i) = O$).
Parity automata and fixpoints 8/8

Determinacy of parity games follows from Lemmas 2 and 1.
Proof of lemmas 1 and 2 (hints) 1/3

\[ WP_P[G] =_{def} \{ g \in G_0 | \exists \text{ a winning strategy for } P \text{ containing } g \} \]

**Lemma A:** \( WP_P[G] \cap WP_O[G] = \emptyset \).

**Lemma B:** A path \( \gamma \) that visits \( X \) infinitely often is winning in \( \mu_S.G[X] \).

**Lemma C:** A path that is eventually winning in \( G[X] \) is winning in \( \nu_S.G[X] \).
Proof of lemmas 1 and 2 (hints) 2/3

\[ F_P(A) = \text{def} \{ g \in G_0 \mid (\epsilon g = P \Rightarrow \exists g' (g \rightarrow g' \text{ and } g' \in A)) \text{ and } (\epsilon g = O \Rightarrow \forall g' (g \rightarrow g' \Rightarrow g' \in A) \} \]

When a play reaches \( F_P(A) \), P can force the play to go into \( A \).

The operator of Lemma 1 is \( A \mapsto WP_P[G[X \cap F_P(A)]] \).
A glimpse of the proof of Lemma 1. If \( Z \) is a postfixpoint, i.e., \( Z \subseteq WP_P[G[X \cap F_P(Z)] \), then construct the following strategy: play according to \( G[X \cap F_P(Z)] \), until eventually reaching \( X \cap F_P(Z) \), then force the play to come to \( Z \), and continue to play according to \( G[X \cap F_P(Z)] \), etc…
The goal is to make semantics akin to syntax and to model computation as interaction between

\[ \left\{ \begin{array}{l}
\text{a system} \\
\text{a program} \\
P
\end{array} \right\} \quad \text{and} \quad \left\{ \begin{array}{l}
\text{its environment} \\
\text{its context} \\
O
\end{array} \right\} \]
while keeping a suitable level of mathematical abstraction (categories), and hence the possibility to use powerful reasoning tools.

Abramsky-Jagadeesan-Malacaria, Hyland-Ong (1993)
PRECURSORS

- Dialogue games of Lorenzen, Lorenz, Felscher (1960)
- Sequential algorithms of Berry and Curien (1978) (like M. Jourdain, we did not know that we were talking about games and strategies!)