Introduction to Infinite Games

Jan Křetínský

Never-ending games seminar
May 6, 2013

inspired by lectures by Wolfgang Thomas and books
Infinite words and *Automata, Logics, and Infinite games*

Motivation I

Alonzo Church, 1957

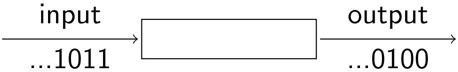
"Given a requirement which a circuit is to satisfy, we may suppose the requirement expressed in some suitable logistic system which is an extension of restricted recursive arithmetic. The synthesis problem is then to find recursion equivalences representing a circuit that satisfies the given requirement (or alternatively, to determine that there is no such circuit)."

Motivation I

Alonzo Church, 1957

"Given a requirement which a circuit is to satisfy, we may suppose the requirement expressed in some suitable logistic system which is an extension of restricted recursive arithmetic. The synthesis problem is then to find recursion equivalences representing a circuit that satisfies the given requirement (or alternatively, to determine that there is no such circuit)."

Given a requirement on a bit stream transformation



fill the box by a machine with output, satisfying the requirement (or state that the requirement is not satisfiable).

Motivation II

Given Kripke structure K and formula ϕ (e.g. of modal μ-calculus), the model checking problem is to decide whether $K \models \phi$.

Motivation II

Given Kripke structure K and formula ϕ (e.g. of modal μ-calculus), the model checking problem is to decide whether $K \models \phi$.

Solution: Combine K and ϕ into a "product game graph" $K \times \phi$ with the a "parity winning condition" such that

$$K \models \phi$$

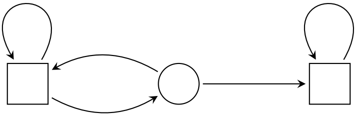
iff

from a designated vertex of $K \times \phi$ player 0 has "winning strategy".

Infinite games

Game $G = (A, W)$

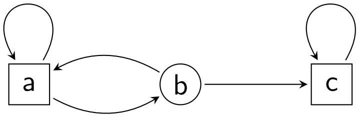
Arena A is an oriented graph (V, E) with partioning $V = V_0 \uplus V_1$



Winning condition $W \subseteq V^\omega$

Playing and winning a game

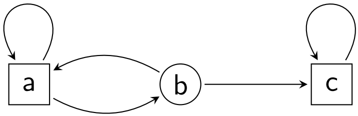
Example: how to win from b?



$$W = V^* c^\omega$$

$\rho = bc^\omega$ is winning for Player 0
$\rho = (ba)^\omega$ is winning for Player 1

Example: how to win from b?



$W = V^* c^\omega$

$\rho = bc^\omega$ is winning for Player 0
$\rho = (ba)^\omega$ is winning for Player 1

ρ conforms to a strategy $\sigma : V^* \to V$ if $\rho[i+1] = \sigma(\rho[1] \cdots \rho[i])$ whenever $\rho[i] \in V_0$
σ is winning if all runs conforming to σ are winning

Winning conditions

- Reachability $F \subseteq V$:

$$\exists i : \rho[i] \in F$$

- Büchi $F \subseteq V$:

$$\mathrm{Inf}(\rho) \cap F \neq \emptyset$$

- Muller $\mathcal{F} \subseteq 2^V$:

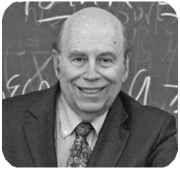
$$\mathrm{Inf}(\rho) \in \mathcal{F}$$

- Rabin $\{(F_1, I_1), \ldots, (F_n, I_n)\}$:

$$\exists i : \quad \mathrm{Inf}(\rho) \cap F_i = \emptyset \quad \& \quad \mathrm{Inf}(\rho) \cap I_i \neq \emptyset$$

- Parity $c : V \to C$:

$$\max_{v \in \mathrm{Inf}(\rho)} c(v) \text{ is even} \qquad \text{(often min instead)}$$

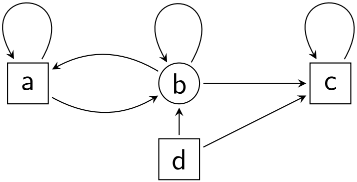


The problem is to compute the winning region Win

- Reachability
- Büchi
- Muller
- Rabin
- Parity

Controllable predecessor:

$$cpred(X) = \{v \in V_0 \mid \exists (v,x) \in E : x \in X\} \cup \{v \in V_1 \mid \forall (v,x) \in E : x \in X\}$$



$F = \{c\}$

Reachability games

Controllable predecessor:

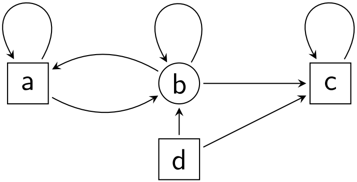
$$\begin{aligned} cpred(X) = &\{v \in V_0 \mid \exists (v,x) \in E : x \in X\} \\ \cup &\{v \in V_1 \mid \forall (v,x) \in E : x \in X\} \end{aligned}$$

Attractor construction: $\text{Attractor}_0(F)$ is the fixpoint of

$$\text{Attractor}_0{}^{0}(F) = F$$

$$\text{Attractor}_0{}^{i+1}(F) = \text{Attractor}_0{}^{i}(F) \cup cpred(\text{Attractor}_0{}^{i}(F))$$

$$Win(Reach(F)) = \text{Attractor}_0(F)$$

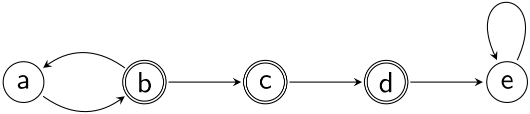


$F = \{c\}$

Accepting states on "controllable" cycles are the fixpoint C of

$$C_0 = F$$
$$C_{i+1} = C_i \cap cpred(\text{Attractor}_0(C_i))$$

$$Win(Buchi(F)) = \text{Attractor}_0(C)$$



$F = \{b, c, d\}$

Muller games I - the difficulty

DJW game:

Arena: repeat

1. Player 1 picks A, B, C, or D
2. Player 0 picks 1, 2, 3, or 4

Winning condition: ρ is winning if

- the highest number in $\mathrm{Inf}(\rho)$ is the number of letters in $\mathrm{Inf}(\rho)$

Muller games II - latest appearance record

Muller game $(V, E), \mathcal{F} \longrightarrow$ parity game $(V', E'), c$

arena (V', E') must be different

Muller games II - latest appearance record

Muller game $(V, E), \mathcal{F} \longrightarrow$ parity game $(V', E'), c$

- Parity condition can be expressed as a Muller condition
- Winning strategies need memory in Muller games
- Winning strategies need no memory in parity games

$\Rightarrow$ arena (V', E') must be different

Muller games II - latest appearance record

Muller game $(V, E), \mathcal{F} \longrightarrow$ parity game $(V', E'), c$

- Parity condition can be expressed as a Muller condition
- Winning strategies need memory in Muller games
- Winning strategies need no memory in parity games

$\Rightarrow$ arena (V', E') must be different

$V' = \{(i_1, \dots i_{h-1}, \underline{i_h}, i_{h+1} \dots, i_{|V|}) \mid$ permutation of V with a "pointer"$\}$

E' contains $(i_1, \dots, i_{|V|}) \rightarrow (i_m, i_1, \dots \underline{i_{m-1}}, i_{m+1} \dots, i_{|V|})$ for $(i_1, i_m) \in E$

Muller games II - latest appearance record

Muller game $(V, E), \mathcal{F} \longrightarrow$ parity game $(V', E'), c$

- Parity condition can be expressed as a Muller condition
- Winning strategies need memory in Muller games
- Winning strategies need no memory in parity games

$\Rightarrow$ arena (V', E') must be different

$V' = \{(i_1, \ldots i_{h-1}, \underline{i_h}, i_{h+1} \ldots, i_{|V|}) \mid$ permutation of V with a "pointer"$\}$

E' contains $(i_1, \ldots, i_{|V|}) \rightarrow (i_m, i_1, \ldots \underline{i_{m-1}}, i_{m+1} \ldots, i_{|V|})$ for $(i_1, i_m) \in E$

Latest appearance record $LAR(V, E) := (V', E')$

$V' = \{(i_1, \ldots i_{h-1}, \underline{i_h}, i_{h+1} \ldots, i_{|V|}) \mid$ permutation of V with a "pointer"$\}$
E' contains $(i_1, \ldots, \underline{i_{|V|}}) \to (i_m, i_1, \ldots \underline{i_{m-1}}, i_{m+1} \ldots, i_{|V|})$ for $(i_1, i_m) \in E$

Muller games III - coloring

$V' = \{(i_1, \dots i_{h-1}, \underline{i_h}, i_{h+1} \dots, i_{|V|}) \mid$ permutation of V with a "pointer"$\}$
E' contains $(i_1, \dots, i_{|V|}) \to (i_m, i_1, \dots \underline{i_{m-1}}, i_{m+1} \dots, i_{|V|})$ for $(i_1, i_m) \in E$

- Run $i_1, i_2 \dots$ over (V, E) corresponds to run $(i_1, \dots), (i_2, \dots) \dots$ over $LAR(V, E)$
- k := maximal position of underlining used ∞-often
- Eventually, the states $i_{k+1}, \dots, i_{|V|}$ stay fixed and are never visited again
- and precisely $i_1, \dots, i_k$ will be visited infinitely often
- Player 0 wins if $\{i_1, \dots, i_k\} \in \mathcal{F}$

Muller games III - coloring

$V' = \{(i_1, \dots i_{h-1}, \underline{i_h}, i_{h+1} \dots, i_{|V|}) \mid$ permutation of V with a "pointer"$\}$
E' contains $(i_1, \dots, \underline{i_{|V|}}) \to (i_m, i_1, \dots \underline{i_{m-1}}, i_{m+1} \dots, i_{|V|})$ for $(i_1, i_m) \in E$

- Run $i_1, i_2 \dots$ over (V, E) corresponds to run $(i_1, \dots), (i_2, \dots) \dots$ over $LAR(V, E)$
- k := maximal position of underlining used ∞-often
- Eventually, the states $i_{k+1}, \dots, i_{|V|}$ stay fixed and are never visited again
- and precisely $i_1, \dots, i_k$ will be visited infinitely often
- Player 0 wins if $\{i_1, \dots, i_k\} \in \mathcal{F}$

$$c(i_1 \dots \underline{i_h} \dots i_{|V|}) := \begin{cases} 2h - 1 & \text{if } \{i_1, \dots, i_h\} \notin \mathcal{F} \\ 2h & \text{if } \{i_1, \dots, i_h\} \in \mathcal{F} \end{cases}$$

Muller games III - coloring

$V' = \{(i_1, \ldots i_{h-1}, \underline{i_h}, i_{h+1} \ldots, i_{|V|}) \mid$ permutation of V with a "pointer"$\}$
E' contains $(i_1, \ldots, \underline{i_{|V|}}) \to (i_m, i_1, \ldots \underline{i_{m-1}}, i_{m+1} \ldots, i_{|V|})$ for $(i_1, i_m) \in E$

- Run $i_1, i_2 \ldots$ over (V, E) corresponds to run $(i_1, \ldots), (i_2, \ldots) \ldots$ over $LAR(V, E)$
- $k :=$ maximal position of underlining used ∞-often
- Eventually, the states $i_{k+1}, \ldots, i_{|V|}$ stay fixed and are never visited again
- and precisely $i_1, \ldots, i_k$ will be visited infinitely often
- Player 0 wins if $\{i_1, \ldots, i_k\} \in \mathcal{F}$

$$c(i_1 \ldots \underline{i_h} \ldots i_{|V|}) := \begin{cases} 2h - 1 & \text{if } \{i_1, \ldots, i_h\} \notin \mathcal{F} \\ 2h & \text{if } \{i_1, \ldots, i_h\} \in \mathcal{F} \end{cases}$$

Proposition: $\mathrm{Inf}(\rho) \in \mathcal{F}$ iff $\max_{v \in \mathrm{Inf}(\rho')} c(v)$ is even

Conclusion

Infinite games with finite arenas and various ω-regular winning conditions:

- Reachability - solution using the attractor construction
- Büchi - iterating attractors
- Muller - reduction to parity using the last appearance record
- Rabin - via Muller or directly
- Parity - various methods