Games with Finitary Parity and Finitary Streett Objectives

Krishnendu Chatterjee (IST Austria)

Joint work with
Thomas A. Henzinger (IST Austria and EPFL)
Florian Horn (CNRS, LIAFA Universite Paris 7, France)

GASICS/DOTS 2010, Paris, France

Model Checking and Synthesis

- Model checking: given a model (of a system) and a desired set of behaviors (specification) whether the model satisfies the specification.

$$M \models \psi$$

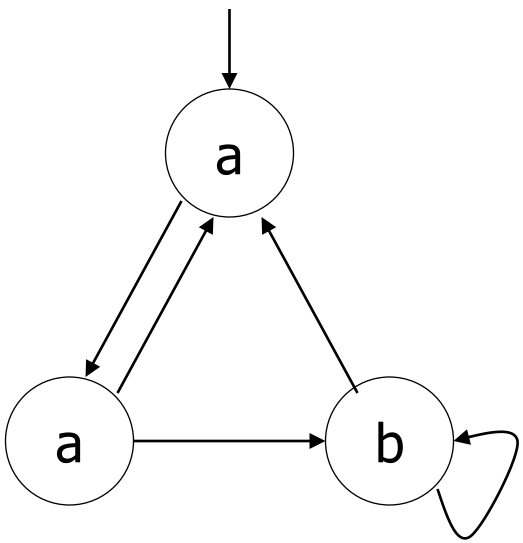
- Synthesis: the model in open environment; system interacts with the environment.

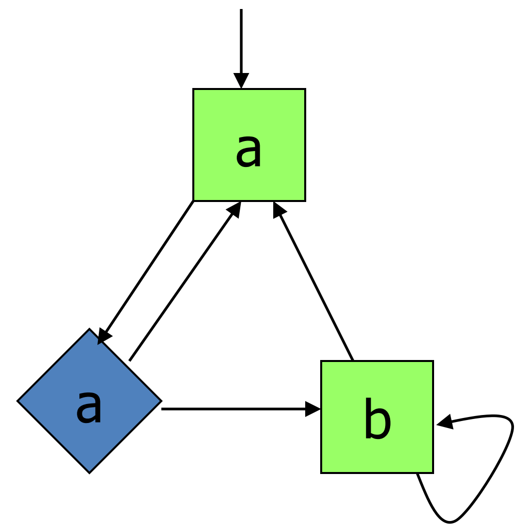
$$M \; || \; E \models \psi$$

Verification and Synthesis

- Model for verification (model checking): labeled transition systems.
- Model for synthesis: games on graphs.
- Specification: an ω–regular set of paths on the graphs.

Labeled Graphs vs. Labeled Games





ω–Regular Objectives

- ω-regular objectives
  - Extension of classical regular languages to infinite strings.
  - Regular: Union ($\cup$), Concatenation (.) and Kleene star (finite repetition (*)).
  - Extension to infinite string by infinite repetition (ω).

$$r ::= \sigma \mid r \cup r \mid r \cdot r \mid (r)^* \mid (r)^\omega \quad \sigma \in \Sigma$$

  - Non-deterministic Buechi automata as compared to finite-automata.
  - Deterministic parity and Streett automata are canonical forms to express ω-regular languages.

ω–Regular Objectives

- ω-regular objectives
  - Extension of classical regular languages to infinite strings: infinite behaviors of non-terminating systems.
  - Express liveness, fairness and most commonly used specifications in verification.
  - Robust and expressive language for specifications.

ω–Regular Objectives

- ω-regular objectives
  - All ω-regular objectives can be decomposed as a safety part and a liveness part [Alpern-Schneider 85].
  - Liveness: something "good" eventually happens.

Strengths of Classical Formulation

- Classical infinitary formulation
  - Strengths
    - Robustness: Independence of granularity of transitions.
    - Simplicity: Abstraction of complicated time bounds.

Weakness of Classical Formulation

- Classical infinitary formulation
  - Weakness
    - Something “good” happens eventually and the time something “good” happens may be unbounded.
    - Everything happens eventually!!!

    - Elevator controller: every request must be eventually granted.

Finitary Formulation

- Finitary fairness [Alur-Henzinger 94]
  - There exists unknown fixed bound b such that something "good" happens within b transitions.
  - Retains simplicity and robustness of ω-regular specifications.
  - Rules out undesired behaviors.
- Synthesis of finitary specifications gives games with finitary objectives.
  - This talk: Games with finitary parity and Streett objectives.

Synthesis of Finitary ω-Regular Conditions

- Game graphs
  - Two-player perfect-information games.
- Winning conditions
  - Finitary parity conditions: canonical form for ω-regular objectives.
  - Finitary Streett conditions: strong fairness.

Game Graphs

Components for Games on Graphs

- Where: the game arena (perfect-informatin turn-based games)
- Why: objectives (finitary parity and finitary Streett)
- How: strategies (pure strategies).

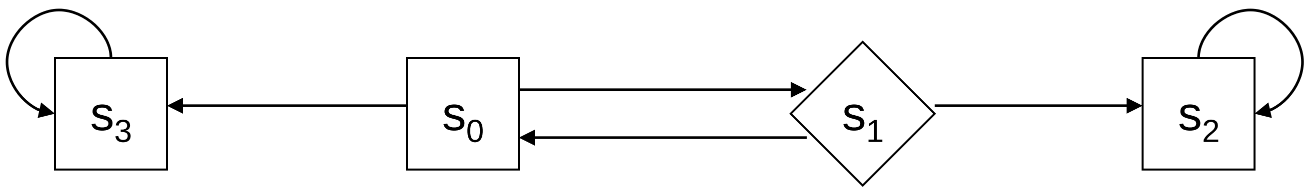
Turn-based Games

- Turn-based deterministic or Perfect-information deterministic Games
  - Game graph $G=((V,E), (V_1,V_2))$.
  - V : finite set of states (vertices).
  - $E \subseteq V \times V$.
  - (V_1,V_2): a partition of the state space.
- The game
  - Played by moving token along edges of the graph.

Play and Strategies

- Play or outcome of a game
  - Infinite sequence of states $(s_1,s_2,\ldots)$ such that for all $i \geq 1$, we have $(s_i,s_{i+1}) \in E$.
  - We denote by Ω the set of all plays.
- Strategies
  - For player 1 given a history of play specifies how to extend the play; $\sigma : V^* \cdot V_1 \to V$ such that for all $x \in V^*$ and $v \in V_1$ we have $(v,\sigma(x \cdot v)) \in E$.
  - Similar definition of strategy π for player 2.

Example: a game graph



- Example of a play: $s_0\ s_1\ s_2^\omega$
- Strategy:
  - Player 1 : $s_0 \to s_1$
  - Player 2 : $s_1 \to s_2$

Play and Strategies

- Strategies
  - Memoryless strategies: Independent of history of the play, i.e., $\sigma : V_1 \to V$.
  - Finite memory strategies: depends only on finite amount of information about the past history, i.e., can be played with a finite automaton.
- Play
  - Unique play: given strategy σ,π and a starting state s there is a unique play denoted as $\Omega_{\sigma,\pi}(s)$.

Objectives

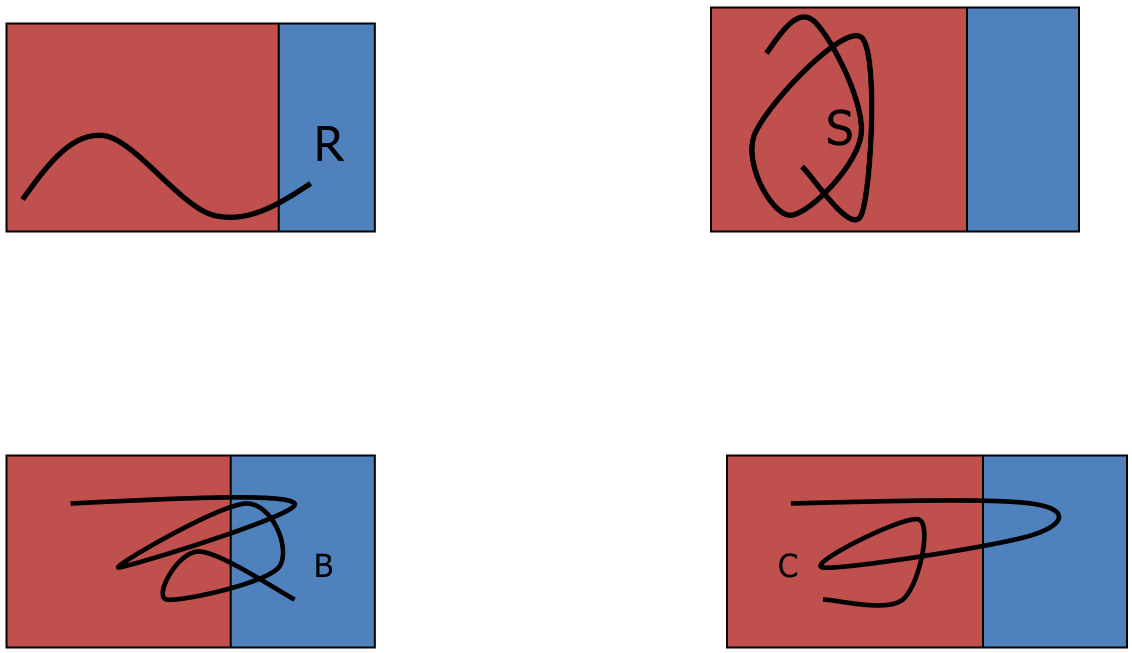
Objectives

- Plays: infinite sequence of states.
- Objectives: subset of plays, $\Psi_1 \subseteq V^\omega$.
- Play is winning for player 1 if it is in Ψ_1.
- Zero-sum game: $\Psi_2 = V^\omega \setminus \Psi_1$.

Reachability, Safety, Buechi and coBuechi Objectives

- Reachability and Safety objectives:
  - Reachability: given a set $R \subseteq V$, the objective requires to visit a state in R.
  - Safety: given a set $F \subseteq V$, the objective requires to remain always in F.
- Buechi and coBuechi Objectives:
  - Buechi: given a set $B \subseteq V$, the objective requires to visit B infinitely often.
  - coBuechi: given a set $C \subseteq V$, the objective requires to visit $V \setminus C$ finitely often.

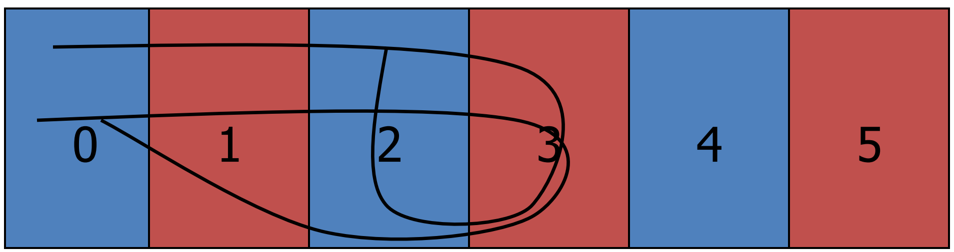
Objectives



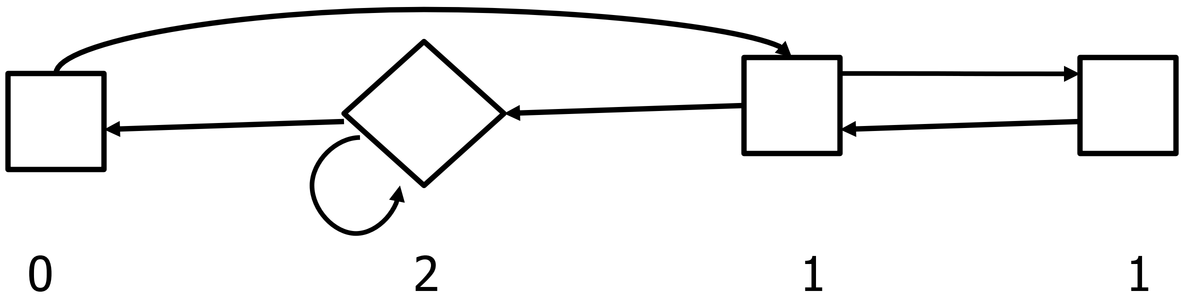
Parity Objectives

- Parity objectives:
  - Consider a priority function $p: V \rightarrow \{0,1,2,\ldots,d\}$.
  - The parity objective requires that the minimum priority visited infinitely often along a play is even.
  - Canonical for ω-regular objectives.

Objectives



Example



Box is Player 1 and winning strategy is to go left.

Streett Objectives

- Let $\{(E_1,F_1), (E_2,F_2),\ldots, (E_d,F_d)\}$ set of state set pairs.

  – Streett: requires for every pair (E_j,F_j) if F_j infinitely often then E_j infinitely often.
    - $\forall$ j. (F_j finitely $\vee$ E_j infinitely) =

      $\forall$ j. (F_j infinitely $\rightarrow$ E_j infinitely)
    - Conjunction of fairness conditions.

Games with ω-Regular Objectives

- Objectives: Φ_1 for player 1 and $\Phi_2 = V^\omega \setminus \Phi_1$ for player 2:
  - Determinacy [Gurevich Harrington, Buechi Landweber]
    For all states s one of the following holds:
    - $\exists \sigma \, \forall \pi. \, \Omega_{\sigma,\pi}(s) \in \Phi_1$;
    - $\exists \pi \, \forall \sigma. \, \Omega_{\sigma,\pi}(s) \in \Phi_2$.
  - Finite-memory determinacy: there always exist witness winning strategies that are finite-memory.

Parity and Streett Games

- Objectives: Φ_1 for player 1 and $\Phi_2 = V^\omega \setminus \Phi_1$ for player 2:
  - Parity objectives: memoryless winning strategies for both players in the respective winning set [Emerson-Jutla].
  - Streett objectives: finite-memory winning strategy for the player with Streett objective and the other player has memoryless winning strategy [Emerson-Jutla].
  - Strett objectives with d-pairs require d! memory [Zielonka , Dziembowski-Jurdzinski-Walukiweicz]

Parity and Streett Games

- Objectives: Φ_1 for player 1 and $\Phi_2 = V^\omega \setminus \Phi_1$ for player 2:
  - Computational complexity:
    - Parity objectives lie in NP $\cap$ co NP [Emerson-Jutla]; also in UP $\cap$ coUP [Jurdzinski].
    - Streett objectives: coNP-complete [Emerson-Jutla].

Finitary Parity Objectives

Finitary Parity Objectives

- Parity objectives requires minimum priority visited infinitely often to be even.

- Finitary parity objectives distance to a better even priority, for odd priorities visited infinitely often, to be bounded.

Finitary Parity Objective

- Given a play $\omega=(s_0,s_1,s_2,\ldots)$ consider the distance sequence as follows:

  – $dist_i(\omega,p)= 0$ if $p(s_i)$ is even; and otherwise $\inf\{ j - i : j > i,\ p(s_j) < p(s_i)$ and $p(s_j)$ is even $\}$.

  – Finitary parity objective requires the distance sequence be bounded in the limit:

    $\limsup_i dist_i(\omega,p)$ to be bounded.

Finitary Parity Objectives

- Finitary coBuechi objectives coincide with coBuechi objectives.

- Finitary Buechi objectives may be strict subset of Buechi objectives.
  - e.g., the distance sequence to Buechi states be 1,2,3,4,…
  - Visit Buechi states infinitely often, but infinitely rarely.

- For a parity objective, the finitary parity objective is in general a strict subset of the plays defined by parity objectives.

Finitary Parity Objectives

- CoBuechi = Finitary CoBuechi.

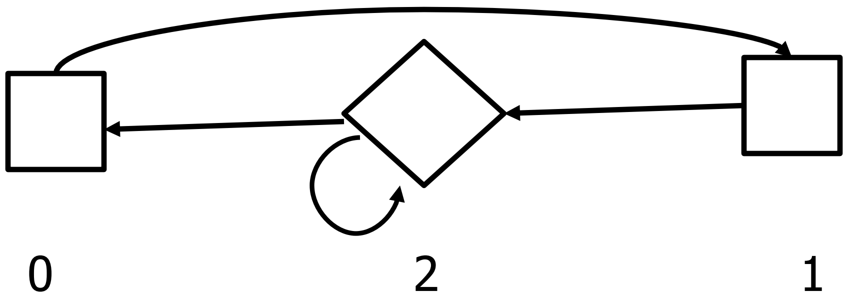
- Buechi $\subsetneq$ Finitary Buechi.

- Parity $\subsetneq$ Finitary Parity.

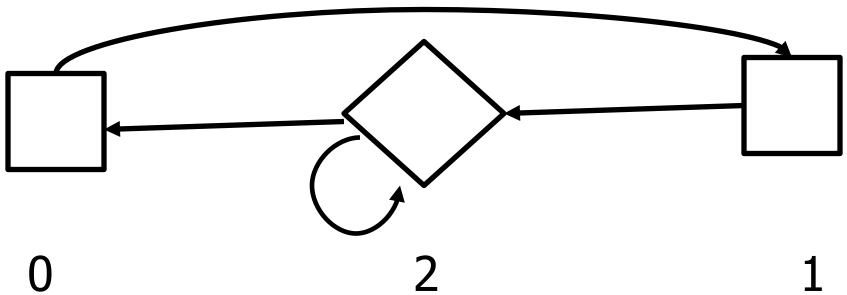
Finitary Parity Objectives

- Finitary coBuechi objectives coincide with coBuechi objectives: so winning sets coincide.
- Finitary Buechi objectives may be strict subset of Buechi objectives.
  - However, it can be shown for games winning sets of finitary Buechi and Buechi objectives coincide (mainly due to existence of memoryless strategies).
- Finitary parity objectives may be strict subset of parity objectives and the winning set may also be strict subset.

Example

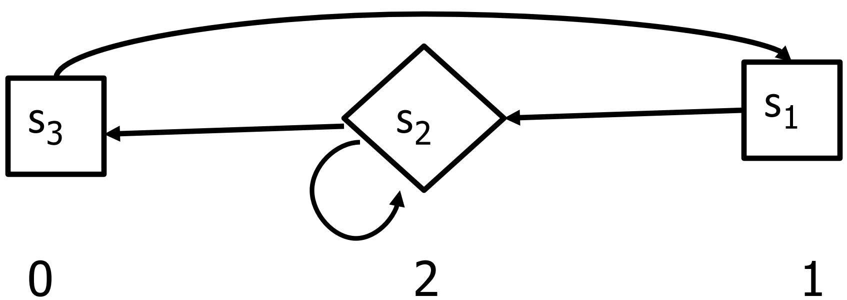


Example



Parity objective: player 1 wins.

Example



Parity objective: player 1 wins.
Finitary parity objective: player 2 play the following strategy in rounds.
In round k stay in s_2 for k times and move to s_3 and proceed to round k+1. The strategy is winning for player 2.
Player 2 winning strategy require infinite memory.

Finitary Parity Objectives

- Winning CoBuechi = Winning Finitary CoBuechi.
- Winning Buechi = Winning Finitary Buechi.
- Winning Parity $\subsetneq$ Winning Finitary Parity.

Finitary Parity and Streett Games

- Winning sets different from classical winning set (generally can be a strict subset).

- Winning strategies very different: may require infinite memory.

- Even if classical and finitary winning set coincide the structure of winning strategies may be different.

Finitary Parity Games

Iterative Algorithm

- We presented an iterative algorithm to compute winning sets for both players for finitary parity objectives [C Henzinger Horn].

- The algorithm quite different from the classical iterative and recursive algorithm for infinitary parity objectives.
  - Recall player 2 requires infinite memory strategies.

Iterative Algorithm

- Three step algorithm.
  - Weak-parity objectives.

    $\Downarrow$
  - Bounded parity objectives.

    $\Downarrow$
  - Finitary parity objectives.

Idea of the Iterative Algorithm

- Weak-parity objectives: requires that the minimum priority that appears is even (contrast to classical parity which require the minimum priority to be even).
  - A linear time algorithm is known to solve games with weak parity objectives: basically iteratively applying solution for reachability games.

- Bounded parity objectives: requires the sequence to be bounded (not in the limit, but bounded).
  - An algorithm that iteratively applies the solution of weak parity games to solve bounded parity games is given in [C Henzinger Horn]: complexity is quadratic.

Idea of the Iterative Algorithm

- An iterative algorithm that uses solution of bounded parity games to solve finitary parity games is given in [C Henzinger Horn]
- It requires $O(n)$ iterations of the bounded parity games and the overall time complexity is $O(n^2. m)$

Idea of the Iterative Algorithm

- Bounded parity is stricter than finitary parity.
- We solve bounded parity for Player 1 and take attractor and remove from the game graph. When we stop we have a game graph where all states are winning for Player 2 for complement of bounded parity.
- We show if all states are winning for Player 2 for complement of bounded parity, then an infinite memory strategy can be constructed for Player 2 to violate finitary parity against all Player 1 strategies.

Iterative Algorithm

- The correctness of our algorithm also proves the determinacy result, i.e., for all finitary parity objectives Φ_1, for all states s one of the following holds:
  - $\exists\, \sigma\, \forall\, \pi.\ \Omega_{\sigma,\pi}(s) \in \Phi_1$;
  - $\exists\, \pi\, \forall\, \sigma.\ \Omega_{\sigma,\pi}(s) \in V^{\omega} \setminus \Phi_1$.

Complexity of the Recursive Algorithm

- Time complexity: Recursive algorithm for finitary parity objectives
  - $O(n^2 . m)$ for game graphs with n states, m edges.

- For infinitary parity objectives:
  - Classical recursive [McNaughton 93]: $O(n^{d-1}. m)$.
  - Small-progress measure [Jurdzinski 00]: $O(n^{d/2} . m)$.
  - Big-step-small-step algorithm [Schewe 07] :$O(n^{d/3}.m)$
  - Sub-exponential algorithm [Jurdzinski et.al. 06] : $O(n^{n^{1/2}})$.

Strategy Complexity

- Memoryless winning strategy exist for player 1.
- Winning strategy for player 2 require infinite memory in general.
- Witnesses for winning strategies:
  - Linear size witnesses exist for winning strategies for player 1.
  - Polynomial (quadratic) size witnesses exist for winning strategies for player 2.
    - Mainly by the reachability and safety characterization.

Comparison Infinitary vs Finitary Parity Objectives

	Classical infinitary parity	**Finitary parity**
Computational Complexity	NP and coNP [EJ88] UP and coUP [Jur98]	PTIME-complete
Algorithmic complexity	$O(n^{d/3}.m)$, $n^{O(\sqrt{n})}$	$O(n^2 . m)$
Strategy complexity	Memoryless for both players	Memoryless for finitary parity; opposing player needs infinite memory.

Finitary Streett Objectives

Finitary Streett Objective

- Natural extension in terms of distances
- Streett objective: F_j inf. often then E_j inf. often.
- Distances: from F_j states to E_j states.
- Finitary Streett: the distance sequences be bounded in the limit.

Service and Request Example

- Requests of two-types R_1 and R_2.
- Requests generated by player 2.
- Once a request of a type is generated further requests are of the type are disabled unless the request is granted. Grants are denoted as G_1 and G_2., respectively.
- Streett objective: R_i inf often, then G_i inf often.

Service and Request Example

- Two strategies: queue service and stack service.

- Both strategies are Streett winning.

- Only queue strategy is finitary Streett winning.
  - Stack strategy is not finitary Streett winning as player 2 can generate a request R_1, and then a sequence of R_2 before R_1 is served and this sequence can grow unbounded.

Finitary Streett Games

Finitary Streett Objectives

- The algorithm is an iterative algorithm on a game graph of exponential size. The iterative algorithm uses solution of generalized Buechi objectives. The solution needs the notion of bounded Streett objectives.

- Bounded Streett objectives: requires the distance sequence to be bounded

Finitary Streett Objectives

- Algorithm ideas:
  - For Parity: weak -> bounded -> finitary.
  - For Streett: bounded -> finitary.

- Winning for bounded Streett is same as winning for Request-response objectives.

- Request-response: every request is eventually granted within bounded steps.

Finitary Streett Objectives

- The solution of bounded Streett objectives: by solving generalized Buechi objectives on a game graph of size $O(2^d .n. m)$ [Wallmeier Hutten Thomas].

- An algorithm that uses $O(n)$ iterations of bounded Streett objectives is given in [C Henzinger Horn].

Finitary Streett Objectives

- The algorithm gives an EXPTIME upper bound.

- Complexity bounds for finitary Streett and Request-response are very related (lower bound of the later gives lower bound for the other).

Request-Response and Finitary Streett Games: Complexity

- The complexity is EXPTIME-complete.
  - Parity (no polynomial) whereas finitary parity is cubic.
  - Streett (coNP-complete) whereas finitary Streett is EXPTIME-complete.

- Two natural extension of Streett objectives gives the high complexity.
  - Reduction from APTM and the objective is used to tackle the exponential number of configurations succinctly.

Request-Response and Finitary Streett Games: Complexity

- Streett:
  - Conjunction (Always Eventually F_i implies Always Eventually E_i).
  - coNP-complete

- Request-response
  - Conjunction (Always (F_i implies Eventually E_i))
  - EXPTIME-complete.

Request-Response and Finitary Streett Games: Complexity

- Computational, algorithmic and strategy complexity.
- Computational complexity: bad news!
- Algorithmic complexity: good news!

Request-Response and Finitary Streett Games: Complexity

- Algorithmic complexity
  - Best known algorithm for Streett: $O(n^d . d! . m)$.
  - Request-response games: $O(n.2^d.m)$.
  - Finitary Streett games: $O(n^2 .2^d.m)$.
  - Complexity in terms of game size is polynomial and the exponent is only the number of pairs.

Strategy Complexity

	Player 1	Player 2
Request-response	Upper bound: d .2^d Lower bound: 2^d -1	2^d
Finitary Streett	Upper bound: d .2^d Lower bound: 2^d -1	Infinite memory
Classical Streett	d!	Memoryless

Comparison Infinitary vs Finitary Streett Objectives

	Classical infinitary Streett	**Finitary Streett**
Computational Complexity	coNP-complete	EXPTIME-complete
Algorithmic complexity	$O(n^d \,.\, d! \,.\, m)$	$O(n^2 \,.2^d.m)$
Strategy complexity	d! for Streett; Memoryless for the opposing player	2^d.d for finitary Streett; opposing player needs infinite memory.

Conclusion

- Conclusion
  - Finitary ω-regular conditions
    - Finitary parity and finitary Streett conditions.
    - Retains simplicity and robusteness of classical specifications and rules out undesired behaviors.
  - Synthesis: Games and stochastic games with finitary parity and Streett conditions.
  - Algorithmic and strategy characterization of games with finitary parity and Streett objectives.

Interesting Directions

- Interesting questions
  - Logical and topological characterization of finitary objectives.
  - Improved algorithm for finitary parity games ($O(n^{1.5}\ m)$ or $O(n\ m)$).
  - More general class of games graphs: extension to turn-based stochastic games already done in [C Henzinger Horn MFCS 09].

Thanks

http://www.ist.ac.at/research/research-groups/chatterjee-group/

I|S|T AUSTRIA

Institute of Science and Technology

Questions ???