Good-For-Games Automata

Udi Boker

Based on work of
Thomas Henzinger, Nir Piterman, and Thomas Colcombet

and joint work with
Denis Kuperberg, Orna Kupferman, Karoliina Lehtinen,
and Michał Skrzypczak

Highlights Spotlight Workshop *2020*

Quick State of Affairs

- Good for games (GFG) automata lie in between deterministic and nondeterministic/alternating automata.
- They were defined in 2006 and have recently got a lot of attention.
- They provide a potential for breakthroughs in formal methods, especially in synthesis and game solving.
- This potential was not yet realized…

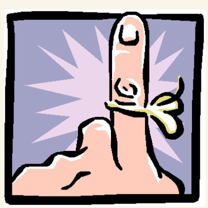
Agenda

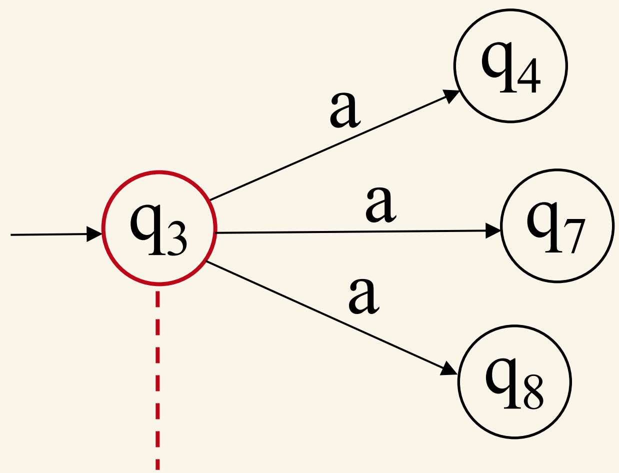
- GFG Automata – Definition(s)
- In the Service of Synthesis
- In the Service of Game Solving
- What We Know and What We Don’t
- The Road Ahead

GFG Automata – Definition(s)

Nondeterminism

- A nondeterministic automaton “decides” which transition to choose at every position.
- The automaton may “see” the whole past and the whole future.

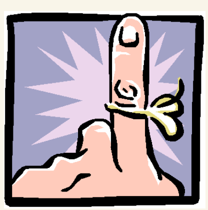


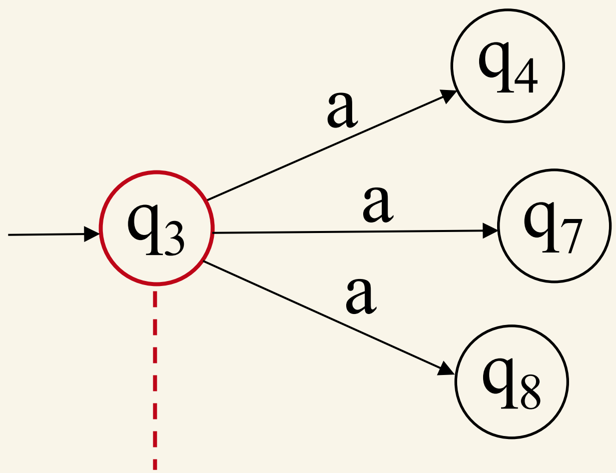


w = “aaababbbbbaaa a abbbabababbababa………….…”

History Determinism

- A nondeterministic automaton *A* is *history-deterministic* if it can "decide" by the past alone: It can accept all words of its language *L(A)*, following a strategy $\boldsymbol{\sigma}$:word-read $\rightarrow$ next state.





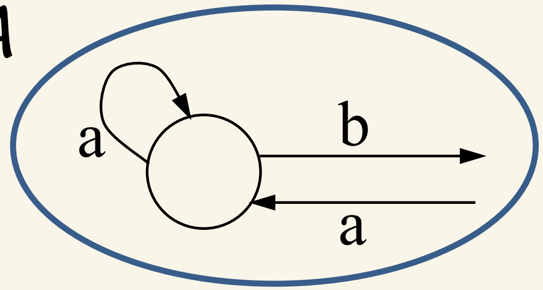
w = "aaababbbbbaaa a

Good For Gameness

- An automaton *A* is *good for games* (GFG) if it composes well with games: For every game G with winning condition L(A), the game G×A has the same winner as G.

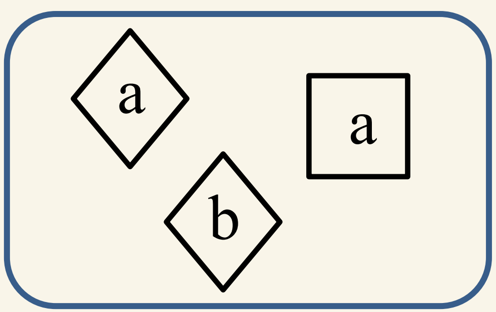
Game *G* between □-Adam and ◊-Eve. Σ-Labeled positions/trans. Eve wins a play if the generated word is in L(A).

A



Automaton *A* over alphabet Σ with acceptance condition α.

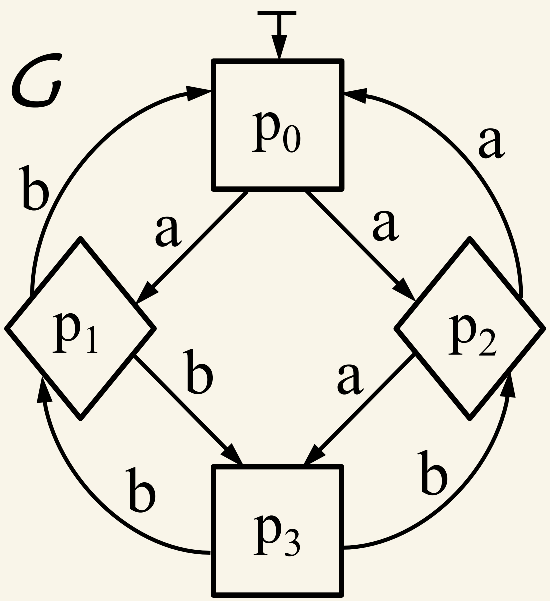
G



G×A

Game between □-Adam and ◊-Eve. Winning condition α.

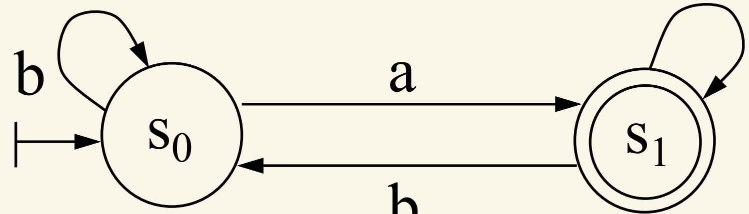
Game × Automaton



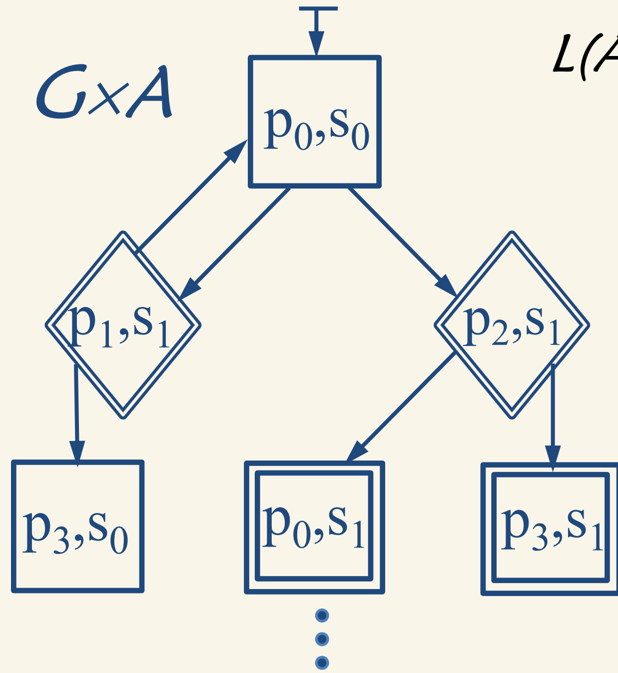
□-Adam and ◊-Eve.
G's (Eve) Winning condition is L(A).

Every deterministic automaton is GFG

Büchi automaton A



L(A) = Infinitely many a's

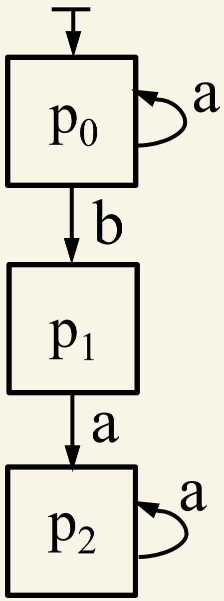


A is GFG: Adam and Eve play the same in G and in G×A. (Looking at the p_i part of positions.)

(G×A)'s winning condition is Büchi

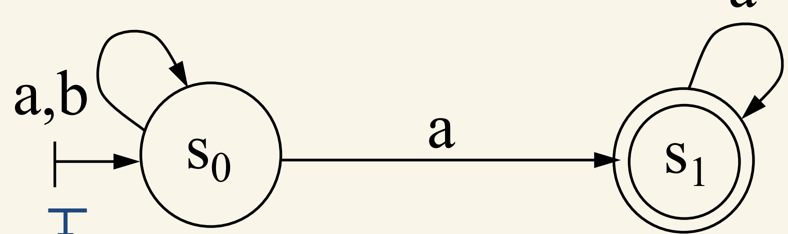
Game × Automaton

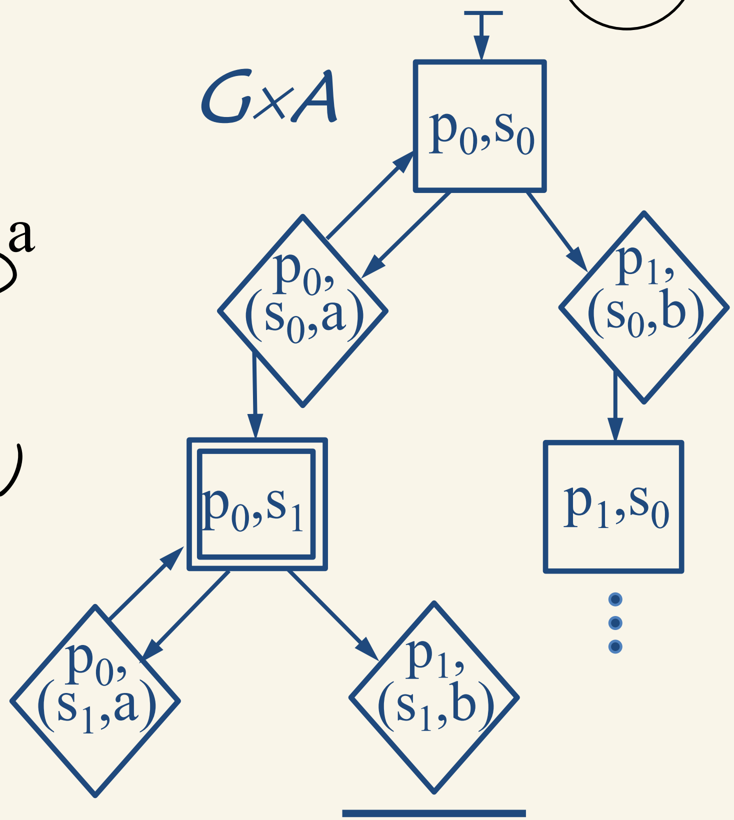
G



G's Winning condition is L(A)

Büchi automaton *A*





L(A) = Finitely many b's

Eve wins *G*, but loses *G×A* !

Not every nondeterministic automaton is GFG

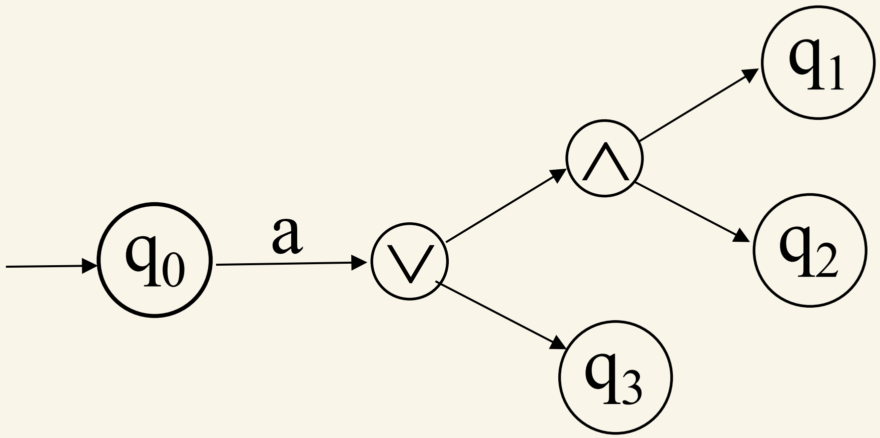
Extending

History Determinism and Good for Gamesness

from nondeterministic to alternating automata

Alternation

- In an alternating automaton there is both nondeterminism and universality.
- A run can be viewed as a game between Eve, who resolves the nondeterminism and wants to accept the word, and Adam who resolves the universality and wants to reject the word.



History Determinism

- An alternating automaton *A* is *history-deterministic* if both Eve and Adam can follow strategies and accept (respectively reject) all words in (respectively out of) the language *L(A)*.

- If only one of them has such a strategy, the automaton is *half-history-deterministic* (with respect to nondeterminism / universality).

Good For Gameness

- The definition is exactly the same as for nondeterminism:
  An automaton *A* is *good for games* (GFG) if it composes well with games: For every game G with winning condition L(A), the game G×A has the same winner as G.
- The definition of the synchronized product G×A is extended – In the product game, Eve controls the nondeterminism within the transitions and Adam controls the universality within them.

Equivalent Definitions

Robust Notion

A nondeterministic/alternating automaton is

History Deterministic [Colcombet, 2009]

iff

Good for Games [Henzinger & Piterman 2006]

iff

Adaquate to Letter Games [Henzinger & Piterman 2006]

iff

Good for Composition with Alternating Automata
[Colcombet, 2013]

iff

Good for Trees [Kupferman & Safra & Vardi 1996]

Equivalence proofs and extensions to alternating automata –
Partly in the above and the rest in [Boker and Lehtinen 2019]

Agenda

✓ GFG Automata – Definition(s)

In the Service of Synthesis

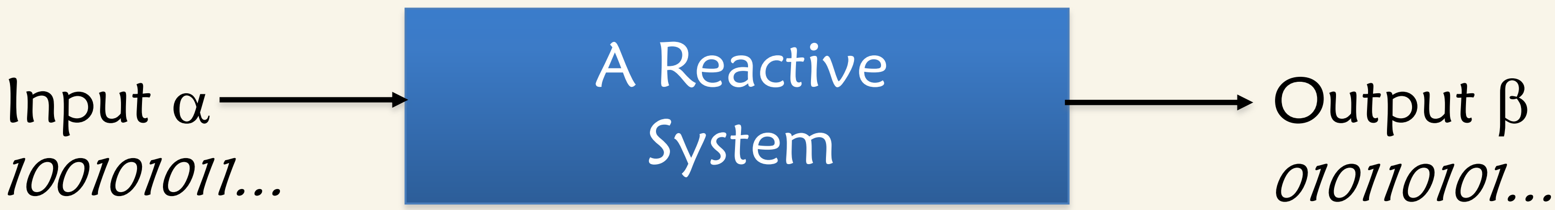
- In the Service of Game Solving
- What We Know and What We Don't
- The Road Ahead

Church's Synthesis Problem

Synthesis: given a specification, decide if there exists a system that *realizes* it, and if there is, automatically construct one.

Reactive synthesis: Synthesis of reactive systems.

- Proposed by Church in 1957; Also called Church's problem.



- The system transforms, letter by letter, an infinite input sequence $\alpha \in (\mathrm{Input})^{\omega}$ into an output sequence $\beta \in (\mathrm{Input})^{\omega}$.
- The pair $(\alpha,\beta) \in (I \times O)^{\omega}$ (or $\in (I \cup O)^{\omega}$) should satisfy the specification, which is usually a logic formula φ.

Synthesis – Complexity

- Büchi and Landweber showed in 1969 that it is decidable for ω-regular specifications.
- Pnueli and Rosner showed in 1989 that for LTL specifications it is 2EXPTIME complete.
- Efficient algorithms for some LTL fragments:
  - Boolean combinations of safety/reachability formulas.
  - Recurrence/persistence formulas.
  - Most notably, GR(1) and some generalizations of it.

- Synthesis has not yet reached a satisfying applicative level.

Solving Synthesis

There are two, fundamentally equivalent, approaches for solving reactive synthesis.

- The first is to reduce the synthesis problem to the problem of solving two-player turn-based games.
  - In this process the specification is first translated to an equivalent deterministic *or GFG* automaton.
- The second is to reduce it to the emptiness problem of tree automata.

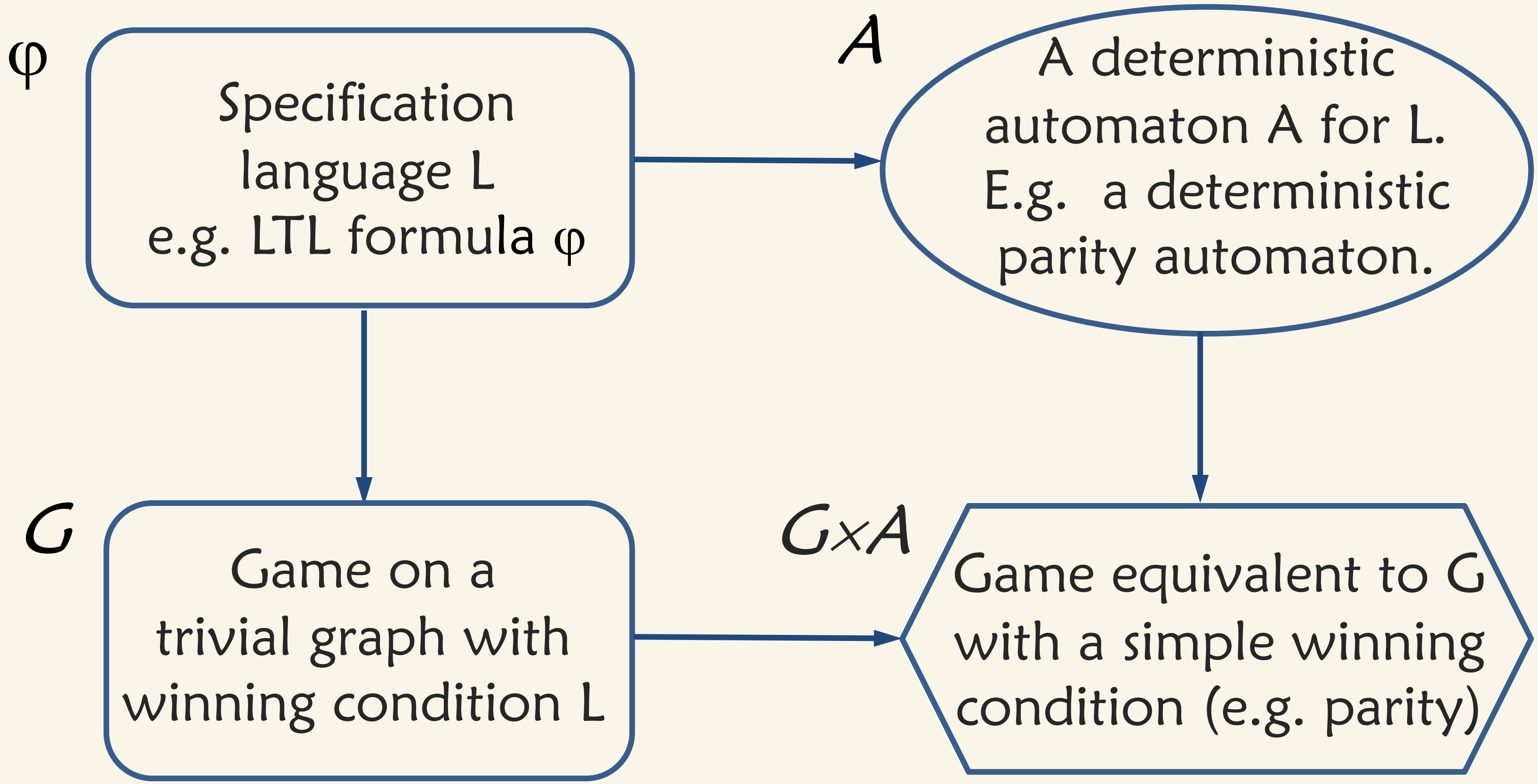
We follow the first approach, in which GFG automata may play a central role,

Synthesis as a Game

Church's synthesis problem is exactly a two-player turn-base win-lose infinite game:

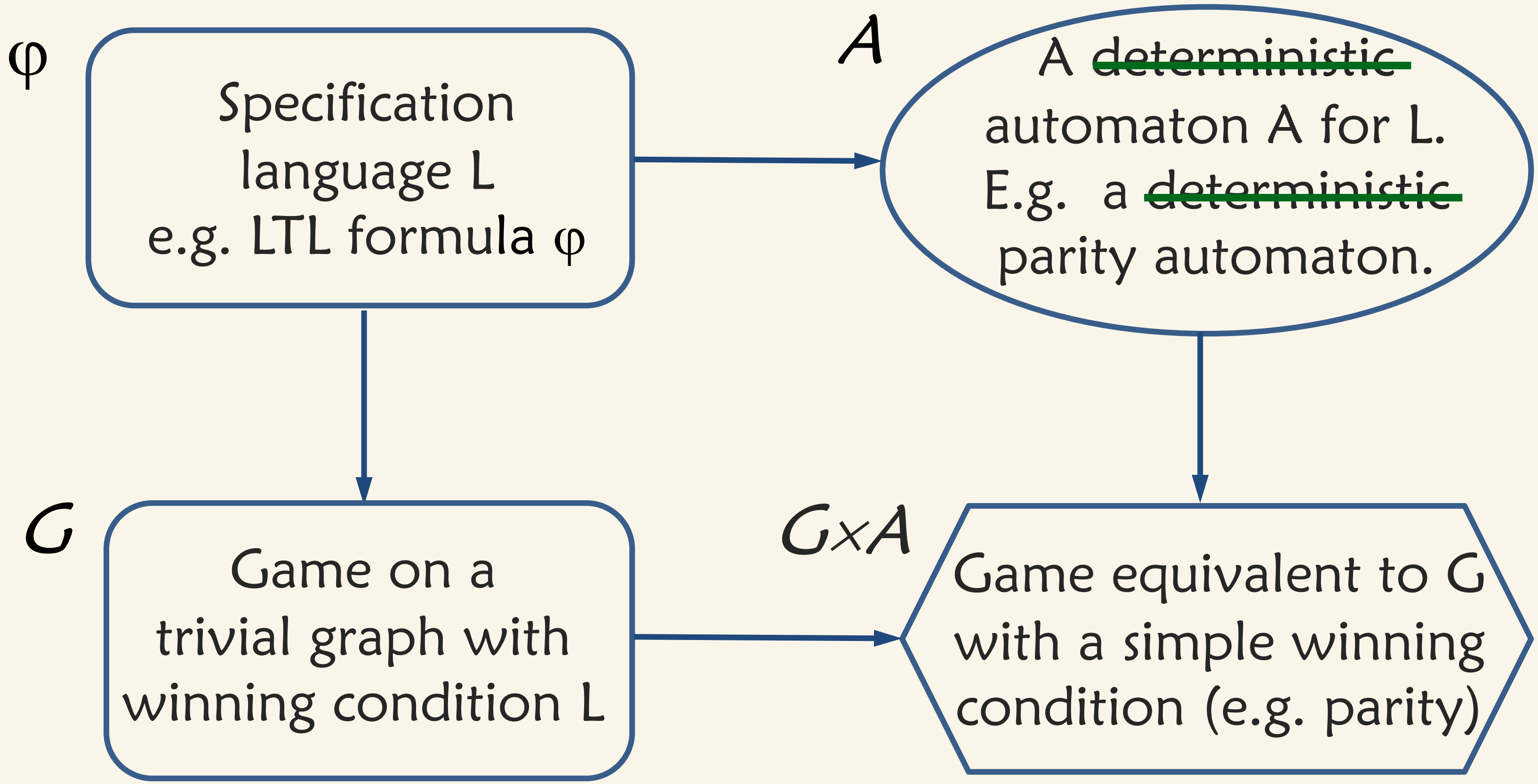
- The environment, producing the input, is Player I – Adam.
- The system, producing the output, is player II – Eve.
- The input and output letters are the players' actions.
- The specification language is the winning condition.
- There exists a transducer iff Eve wins the game.
- The required system is a description of Eve's winning strategy.

Solving the Synthesis Game



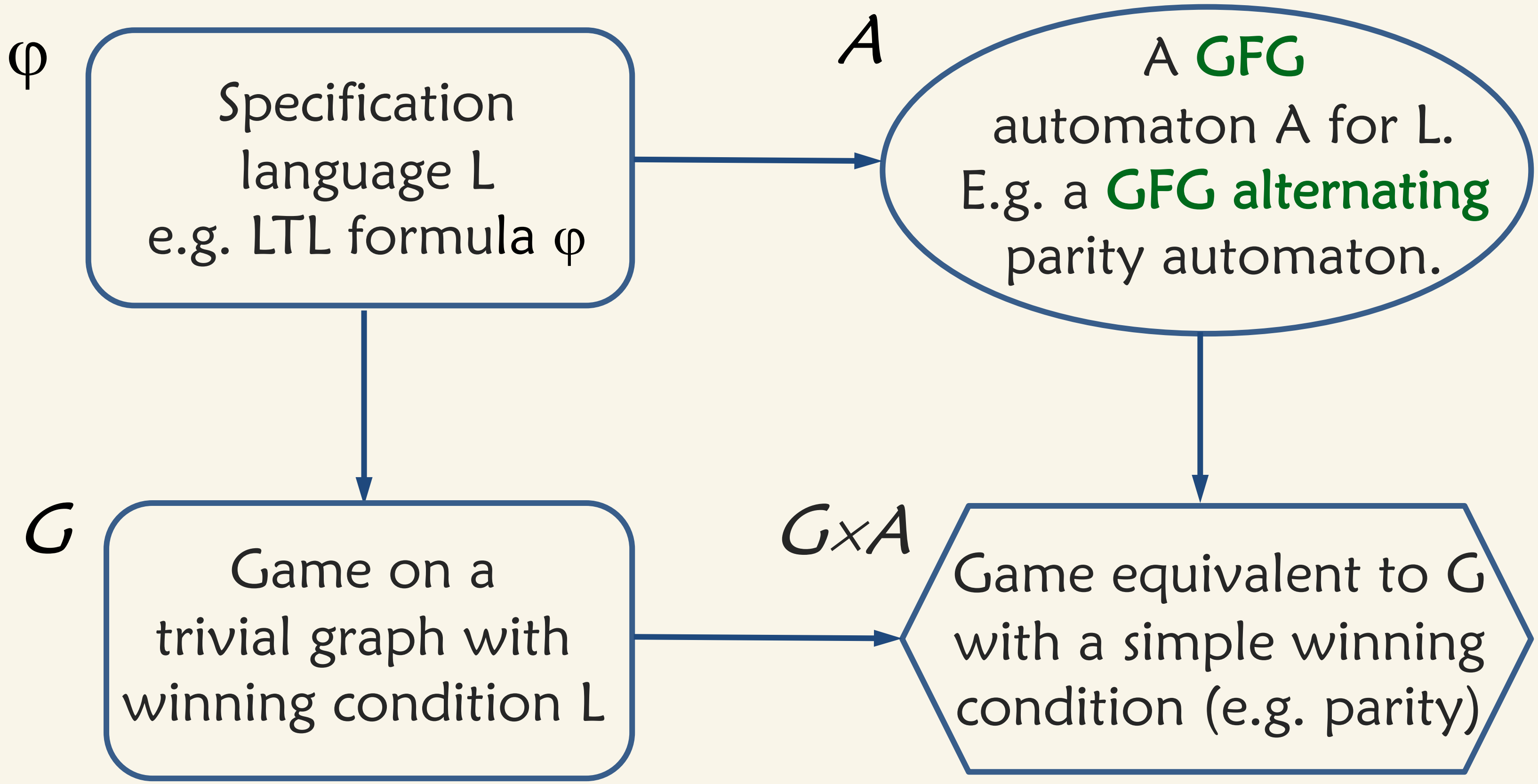
Problem: Deterministic automata are big (double-exp from LTL)

Solving the Synthesis Game



~~Problem: Deterministic automata are big (double-exp from LTL)~~

Solving the Synthesis Game



Challenge: Can GFG automata improve the process?

GFG Automata – Challenges

Can GFG automata be

- Substantially different from deterministic automata?
- More expressive than them?
- More concise?
- Better symbolically represented?
- Concicely translated from useful logics?
  - From interesting fragments of LTL? of μ-calculus?

Agenda

- ✓ GFG Automata – Definition(s)
- ✓ In the Service of Synthesis

In the Service of Game Solving

- What We Know and What We Don't
- The Road Ahead

Quasipolynomial Era

- Since 2017 when Calude, Jain, Khoussainov, Li, and Stephan brought down the upper bound for solving parity games to quasipolynomial, the automata-theoretical aspects of solving parity games have been studied in more depth.
- In particular, Bojańczyk and Czerwiński and Czerwiński et al describe the quasipolynomial algorithms for solving parity games explicitly in terms of deterministic word automata that *separate* some word languages.

Better than Quasipolynomial?

- A polynomial deterministic *or GFG* such automata would imply a polynomial algorithm for parity games (and for other two-player turn-based games on graphs).
- However, Czerwiński at al. showed that the smallest possible such *nondeterministic* automaton is quasipolynomial.
- Can alternating GFG automata allow to use the separation approach for achieving better algorithms?

Good for Small Game

- Lehtinen's recent quasipolynomial algorithm for solving parity games uses an intermediate *nondeterministic* automaton that might not be deterministic nor GFG

- Yet, it is *good for small games*, in the sense that it can be properly composed with games of some bounded size.

- Can alternating good for small games automata allow to further improve improve algorithms for solving two-player turn-based games on graphs?

Agenda

✓ GFG Automata – Definition(s)

✓ In the Service of Synthesis

✓ In the Service of Game Solving

What We Know and What We Don't

- The Road Ahead

Expressiveness

GFG automata vs. deterministic automata

- Regular automata: same expressiveness [KSV96, NW98]
- ω-regular autotomata: same expressiveness [KSV96, NW98]
- Cost automata: GFG are as expressive as nondererminisitc, strictly more than deterministic [Col09]
- ω-PDAs: GFG are more than deterministic and less then nondeterministic. Decidable universality. [LZ20]
- Timed-, limit-average- discounted-sum- register- automata: Preliminary work by Henzinger, Lehtinen and Totzke

Succinctness

- GFG automata over finite words always embody deterministic ones [KSV96]
- When GFG automata were defined [HP06], all examples of GFG automata embodied deterministic ones.
  - The motivation was to have a simpler symbolic representation for these bigger automata.
- Colcombet conjectured in 2012 that

It need not (already Büchi and coBüchi) [BKKM13]

Succinctness – Nondeterministic ω-reg

Are nondeterministic GFG ω-regular automata more succinct than deterministic automata of the same type?

- Weak automata: No.
  They embody deterministic automata [KSV96,Mor03]
- Büchi automata: Still open.
  They are up to quadratically more succinct [KS15]
- Co-Büchi automata: Yes, exponentially!
  [KS15]
- Parity and Streett: Yes, exponentially.
  Deterministic parity and Streett automata are co-Büchi type.
- Rabin and Muller: Still open.
  Deterministic Rabin and Muller are not co-Büchi type.

Succinctness – Alternating ω-reg

Are alternating GFG ω-regular automata more succinct than deterministic automata of the same type?

- Weak automata: No. [BL19]
- Büchi and Co-Büchi : Singly exponential. [BL19]
- Parity : Singly exponential. [BKLS20]
- Street, Rabin and Muller: Still open.

GFGness – Nondeterministic ω-reg

Given a nondeterministic ω-regular automaton, what is the complexity of deciding whether it is GFG?

[KS15]: At least as hard as solving games with the same condition

- Weak automata: Polynomial.
  [Lod11] (Unpublished)
- Büchi automata: Polynomial.
  [BK18] Showing equivalence to solving G2 games
- Co-Büchi automata: Polynomial.
  [KS15]; In [BKLS20a] via G2 games
- Parity: ≤ EXPTIME.
  [HP06]; [BK18]: For fixed priorities – the G2 conjecture.
- Rabin, Street and Muller: ≤ EXPTIME.
  [HP06]: Simulation check against a deterministic automaton

GFGness – Alternating ω-reg

Given an alternating ω-regular automaton, what is the complexity of deciding whether it is GFG?

- Weak automata: Polynomial.
  [BKLS20a] Lifting the G2 game to alternating automata
- Büchi and Co-Büchi automata: $\leq$ EXPTIME.
  [BKLS20a] G2 conjecture $\Rightarrow$ polynomial
- Parity: $\leq$ EXPTIME.
  [BKLS20a] For a fixed index, G2 conjecture $\Rightarrow$ polynomial
  Nondeterministic G2 conjecture $\Leftrightarrow$ Alternating G2 conjecture
- Rabin, Streett and Muller: Open.

Logic Formulas → GFG Automata

What interesting/useful logics or fragments of known logics (sucn as LTL and μ-calculus) are succinctly translated to GFG automata?

- Well… so… you know… look … you see… I mean…

  We still know very little

- Iosti & Kuperberg [IK19]: *SvTL* - Eventually safe μ-calculus.
  - Polynomial translation to GFG nondeterministic Co-Büchi.
- Some preliminary work on fragments of LTL by Kupferman, Sickert, and Abu Radi.

Agenda

- ✓ GFG Automata – Definition(s)
- ✓ In the Service of Synthesis
- ✓ In the Service of Game Solving
- ✓ What We Know and What We Don't

The Road Ahead

The Road Ahead

Thanks

Bring out the potential in GFG automata!

- Expressiveness
  - Can they help with quantitative automata? Other settings?
- Succinctness
  - Alternating ω-regular; stronger conditions; other settings
- Synthesis
  - Interesting logics that succinctly translate to GFG auto.
- Model Checking
  - Interesting logics that succinctly translate to half-GFG auto.
- Game Solving
  - Can GFG or good-for-small game automata help futhur?
- GFGness
  - Complexity? Don't miss Denis' open-problem talk