Church Synthesis Problem for Noisy Input

Yaron Velner [1]
The Blavatnik School of Computer Science
Tel Aviv University, Israel

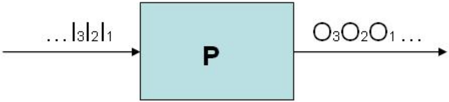
January 29, 2011

[1] Under the supervision of Prof. Alexander Rabinovich

Church Synthesis Problem

- Input: A specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$
- Task: Find a program P which implements L, i.e.,

$$\forall IN \in \{0,1\}^\omega, (IN, OUT = P(IN)) \in L$$



Gale-Stewart Game

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$.
- In every round
  - Player INPUT plays with 0 or 1
  - Player OUTPUT responds with 0 or 1
- Infinite play forms two ω words IN, OUT.
- OUTPUT wins if $(IN, OUT) \in L$.

Gale-Stewart Game

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$.
- In every round
  - Player INPUT plays with 0 or 1
  - Player OUTPUT responds with 0 or 1
- Infinite play forms two ω words IN, OUT.
- OUTPUT wins if $(IN, OUT) \in L$.
- OUTPUT winning strategy is a program for specification L.

Gale-Stewart Game

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$.
- In every round
  - Player INPUT plays with 0 or 1
  - Player OUTPUT responds with 0 or 1
- Infinite play forms two ω words IN, OUT.
- OUTPUT wins if $(IN, OUT) \in L$.
- OUTPUT winning strategy is a program for specification L.
- Theorem Büchi-Landweber theorem 1969
  - Assume that L is ω regular.
    - It is decidable who is the winner.
    - A finite memory strategy for the winner can be constructed.

Extensions of Büchi-Landweber theorem

- Non regular winning conditions
  - Mean payoff condition Ehrenfeucht & Mycielski 1979
  - Context free ω language
  - ...

Extensions of Büchi-Landweber theorem

- Non regular winning conditions
  - Mean payoff condition Ehrenfeucht & Mycielski 1979
  - Context free ω language
  - ...
- Conjunction of winning conditions
  - Mean payoff parity game Chatterjee, Henzinger & Jurdzinski 05
  - Energy parity game Chatterjee & Doyen 10
  - $\bigwedge \mathit{MultiDimensionalMeanPayoff}$ game - partially solved Chatterjee, Doyen, Henzinger & Raskin 10

Extensions of Büchi-Landweber theorem

- Non regular winning conditions
  - Mean payoff condition Ehrenfeucht & Mycielski 1979
  - Context free ω language
  - ...
- Conjunction of winning conditions
  - Mean payoff parity game Chatterjee, Henzinger & Jurdzinski 05
  - Energy parity game Chatterjee & Doyen 10
  - $\bigwedge$ *MultiDimensionalMeanPayoff* game - partially solved Chatterjee, Doyen, Henzinger & Raskin 10
- Games with lookahead degree Holtmann, Kaiser & Thomas 10
  - Possibility to get a lookahead on the moves of the opponent.

Extensions of Büchi-Landweber theorem

- Non regular winning conditions
  - Mean payoff condition Ehrenfeucht & Mycielski 1979
  - Context free ω language
  - ...
- Conjunction of winning conditions
  - Mean payoff parity game Chatterjee, Henzinger & Jurdzinski 05
  - Energy parity game Chatterjee & Doyen 10
  - $\bigwedge MultiDimensionalMeanPayoff$ game - partially solved Chatterjee, Doyen, Henzinger & Raskin 10
- Games with lookahead degree Holtmann, Kaiser & Thomas 10
  - Possibility to get a lookahead on the moves of the opponent.
- Games with imperfect information
  - Observation based strategies.

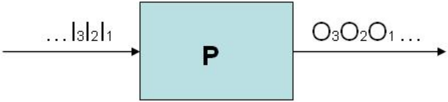
Extensions of Büchi-Landweber theorem

- Non regular winning conditions
  - Mean payoff condition Ehrenfeucht & Mycielski 1979
  - Context free ω language
  - ...
- Conjunction of winning conditions
  - Mean payoff parity game Chatterjee, Henzinger & Jurdzinski 05
  - Energy parity game Chatterjee & Doyen 10
  - $\bigwedge MultiDimensionalMeanPayoff$ game - partially solved Chatterjee, Doyen, Henzinger & Raskin 10
- Games with lookahead degree Holtmann, Kaiser & Thomas 10
  - Possibility to get a lookahead on the moves of the opponent.
- Games with imperfect information
  - Observation based strategies.
  - Games with errors.

Plan

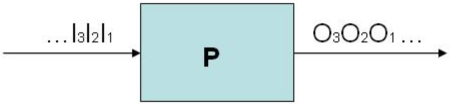
1. Synthesis problem for noisy input.
2. Games with (detected) errors.
3. Regular games with errors.
4. Mean payoff games with errors and a reduction to $\bigwedge \mathit{MultiDimensionalMeanPayoff}$ games.
5. How to determine the winner of $\bigwedge \mathit{MultiDimensionalMeanPayoff}$ games.
6. Games with (undetected) errors.
7. Conclusion & open questions.

Synthesis for noisy input



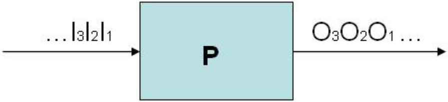
- The input signal is noisy
  - We consider two kinds of errors (noises) in the input signal
    - Detected error - The received signal is $z \notin \{0, 1\}$. The real signal may be any $a \in \{0, 1\}$.
    - Undetected error - The received signal is $a \in \{0, 1\}$. The real signal is $b \in \{0, 1\} - \{a\}$. The program cannot detect whether the signal has an error.

Synthesis for noisy input



- The input signal is noisy
  - We consider two kinds of errors (noises) in the input signal
    - Detected error - The received signal is $z \notin \{0, 1\}$. The real signal may be any $a \in \{0, 1\}$.
    - Undetected error - The received signal is $a \in \{0, 1\}$. The real signal is $b \in \{0, 1\} - \{a\}$. The program cannot detect whether the signal has an error.
- The program must produce an output which correspond the real input signal

Synthesis for noisy input



- The input signal is noisy
  - We consider two kinds of errors (noises) in the input signal
    - Detected error - The received signal is $z \notin \{0, 1\}$. The real signal may be any $a \in \{0, 1\}$.
    - Undetected error - The received signal is $a \in \{0, 1\}$. The real signal is $b \in \{0, 1\} - \{a\}$. The program cannot detect whether the signal has an error.
- The program must produce an output which correspond the real input signal
- However, the amount of allowed errors is limited.
  - If there are "too many" errors in the input signal the program behavior is undefined.

Games with detected errors

Games with detected errors

- Two players game.

Games with detected errors

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$

Games with detected errors

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$
- In every round

Games with detected errors

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$
- In every round
  - Player INPUT plays with 0, 1 or z (detected error).

Games with detected errors

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$
- In every round
  - Player INPUT plays with 0, 1 or z (detected error).
  - Player OUTPUT responds with 0 or 1

Games with detected errors

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$
- In every round
  - Player INPUT plays with 0, 1 or z (detected error).
  - Player OUTPUT responds with 0 or 1
- Infinite play forms (IN, OUT).

Games with detected errors

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$
- In every round
  - Player INPUT plays with 0, 1 or z (detected error).
  - Player OUTPUT responds with 0 or 1
- Infinite play forms (IN, OUT).
- OUTPUT wins if one the following holds

Games with detected errors

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$
- In every round
  - Player INPUT plays with 0, 1 or z (detected error).
  - Player OUTPUT responds with 0 or 1
- Infinite play forms (IN, OUT).
- OUTPUT wins if one the following holds
  - IN has "too many" errors.

Games with detected errors

- Two players game.
- Specification $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$
- In every round
  - Player INPUT plays with 0, 1 or z (detected error).
  - Player OUTPUT responds with 0 or 1
- Infinite play forms (IN, OUT).
- OUTPUT wins if one the following holds
  - IN has "too many" errors.
  - $\forall X$ generated from IN by replacing every z with 0 or 1, we have $(X, OUT) \in L$.

Games with detected errors - cont

How many errors are ”too many”?

Games with detected errors - cont

How many errors are "too many"?

- Scales
  - Error count: $EC(X, n)$ = number of zs in X until position n.
  - Error rate: $ER(X) = \limsup_{n\to\infty} \frac{1}{n} EC(X, n)$

Games with detected errors - cont

How many errors are "too many"?

- Scales
  - Error count: $EC(X, n)$ = number of zs in X until position n.
  - Error rate: $ER(X) = \limsup_{n\to\infty} \frac{1}{n} EC(X, n)$
- Error thresholds

Games with detected errors - cont

How many errors are "too many"?

- Scales
  - Error count: $EC(X, n) =$ number of zs in X until position n.
  - Error rate: $ER(X) = \limsup_{n\to\infty} \frac{1}{n} EC(X, n)$
- Error thresholds
  - Error rate (most interesting):
    - DE_δ are games with detected errors, with error rate threshold $\delta \in [0, 1]$

Games with detected errors - cont

How many errors are "too many"?

- Scales
  - Error count: $EC(X, n)$ = number of zs in X until position n.
  - Error rate: $ER(X) = \limsup_{n\to\infty} \frac{1}{n} EC(X, n)$
- Error thresholds
  - Error rate (most interesting):
    - DE_δ are games with detected errors, with error rate threshold $\delta \in [0, 1]$
  - Finite number of errors:
    - DE_{fin} - number of errors must be finite.

Games with detected errors - cont

How many errors are "too many"?

- Scales
  - Error count: $EC(X, n)$ = number of zs in X until position n.
  - Error rate: $ER(X) = \limsup_{n\to\infty} \frac{1}{n} EC(X, n)$
- Error thresholds
  - Error rate (most interesting):
    - DE_δ are games with detected errors, with error rate threshold $\delta \in [0, 1]$
  - Finite number of errors:
    - DE_{fin} - number of errors must be finite.
  - Fixed number of errors:
    - DE_n - at most n errors in input.

Games with detected errors - cont

How many errors are "too many"?

- Scales
  - Error count: $EC(X, n) =$ number of zs in X until position n.
  - Error rate: $ER(X) = \limsup_{n\to\infty} \frac{1}{n} EC(X, n)$
- Error thresholds
  - Error rate (most interesting):
    - DE_δ are games with detected errors, with error rate threshold $\delta \in [0, 1]$
  - Finite number of errors:
    - DE_{fin} - number of errors must be finite.
  - Fixed number of errors:
    - DE_n - at most n errors in input.
- Bounded number of errors problem:
  - Is there exists $n \in \mathbb{N}$ s.t INPUT is the winner of DE_n?

Regular games with detected errors

Theorem

Regular games with detected errors

Theorem

- It is decidable who is the winner of DE_δ, DE_{fin} and DE_n games.

Regular games with detected errors

Theorem

- It is decidable who is the winner of DE_δ, DE_{fin} and DE_n games.
- The bounded number of errors problem is decidable.

Regular games with detected errors

Theorem

- It is decidable who is the winner of DE_δ, DE_{fin} and DE_n games.
- The bounded number of errors problem is decidable.
- Sketch of proof

Regular games with detected errors

Theorem

- It is decidable who is the winner of DE_δ, DE_{fin} and DE_n games.
- The bounded number of errors problem is decidable.
- Sketch of proof
  - Immediate reduction from DE_n and DE_{fin} to regular games, and from DE_δ to mean payoff parity games.

Regular games with detected errors

Theorem

- It is decidable who is the winner of DE_δ, DE_{fin} and DE_n games.
- The bounded number of errors problem is decidable.
- Sketch of proof
  - Immediate reduction from DE_n and DE_{fin} to regular games, and from DE_δ to mean payoff parity games.
    - It is decidable who is the winner in mean payoff parity game Chatterjee, Henzinger & Jurdzinski 05

Regular games with detected errors

Theorem

- It is decidable who is the winner of DE_δ, DE_{fin} and DE_n games.
- The bounded number of errors problem is decidable.
- Sketch of proof
  - Immediate reduction from DE_n and DE_{fin} to regular games, and from DE_δ to mean payoff parity games.
    - It is decidable who is the winner in mean payoff parity game Chatterjee, Henzinger & Jurdzinski 05
  - There exists a computable $m \in \mathbb{N}$ s.t: INPUT wins DE_{fin} $\Leftrightarrow$ INPUT wins DE_m

Mean payoff games (reminder)

- Game arena $G = (V, E, w : E \to \mathbb{Z})$

Mean payoff games (reminder)

- Game arena $G = (V, E, w : E \to \mathbb{Z})$
- Finite path has a *total payoff* value.
  - For finite path $\pi = e_0 e_1 \dots e_n$.
    - $TP(\pi) = \sum_{i=0}^{n} w(e_i)$

Mean payoff games (reminder)

- Game arena $G = (V, E, w : E \to \mathbb{Z})$
- Finite path has a *total payoff* value.
  - For finite path $\pi = e_0 e_1 \dots e_n$.
    - $TP(\pi) = \sum_{i=0}^{n} w(e_i)$
- Infinite path has a *mean payoff* value.
  - For infinite path π
    - $\overline{MP}(\pi) = \lim_{n\to\infty} \sup\{\frac{1}{k} TP(\pi[0,k]) | k \geq n\}$
    - $\underline{MP}(\pi) = \lim_{n\to\infty} \inf\{\frac{1}{k} TP(\pi[0,k]) | k \geq n\}$

Mean payoff games (reminder)

- Game arena $G = (V, E, w : E \rightarrow \mathbb{Z})$
- Finite path has a *total payoff* value.
  - For finite path $\pi = e_0 e_1 \ldots e_n$.
    - $TP(\pi) = \sum_{i=0}^{n} w(e_i)$
- Infinite path has a *mean payoff* value.
  - For infinite path π
    - $\overline{MP}(\pi) = \lim_{n\rightarrow\infty} \sup\{\frac{1}{k} TP(\pi[0,k]) | k \geq n\}$
    - $\underline{MP}(\pi) = \lim_{n\rightarrow\infty} \inf\{\frac{1}{k} TP(\pi[0,k]) | k \geq n\}$
- Mean payoff condition
  - $\mathrm{MeanPayoffSup}^{\geq}(0)$ - $\overline{MP}$ value of play ≥ 0
  - $\mathrm{MeanPayoffInf}^{\geq}(0)$ - $\underline{MP}$ value of play ≥ 0

Mean payoff games with detected errors

Mean payoff games with detected errors

Decidability results

Mean payoff games with detected errors

Decidability results

- It is decidable who is the winner of DE_n, $\mathrm{DE}_{\mathit{fin}}$ and of DE_δ for $\delta = 0$.

Mean payoff games with detected errors

Decidability results

- It is decidable who is the winner of DE_n, DE_{fin} and of DE_δ for $\delta = 0$.
- Sketch of proof
  - INPUT is the winner of DE_δ for $\delta = 0 \Leftrightarrow$ INPUT is the winner of DE_{fin}.

Mean payoff games with detected errors

Decidability results

- It is decidable who is the winner of DE_n, DE_{fin} and of DE_δ for $\delta = 0$.
- Sketch of proof
  - INPUT is the winner of DE_δ for $\delta = 0 \Leftrightarrow$ INPUT is the winner of DE_{fin}.
  - INPUT is the winner of $\mathrm{DE}_{fin} \Leftrightarrow$ INPUT is the winner of DE_n for $n = 2^{|V|}$

Mean payoff games with detected errors

Decidability results

- It is decidable who is the winner of DE_n, DE_{fin} and of DE_δ for $\delta = 0$.
- Sketch of proof
  - INPUT is the winner of DE_δ for $\delta = 0 \Leftrightarrow$ INPUT is the winner of DE_{fin}.
  - INPUT is the winner of $\mathrm{DE}_{fin} \Leftrightarrow$ INPUT is the winner of DE_n for $n = 2^{|V|}$
  - Reduction from DE_n game to $\bigwedge$ *MultiDimensionalMeanPayoff* game.

Mean payoff games with detected errors

Decidability results

- It is decidable who is the winner of DE_n, DE_{fin} and of DE_δ for $\delta = 0$.
- Sketch of proof
  - INPUT is the winner of DE_δ for $\delta = 0 \Leftrightarrow$ INPUT is the winner of DE_{fin}.
  - INPUT is the winner of $\mathrm{DE}_{fin} \Leftrightarrow$ INPUT is the winner of DE_n for $n = 2^{|V|}$
  - Reduction from DE_n game to $\bigwedge$ *MultiDimensionalMeanPayoff* game.

Undecidability result

- For $\delta > 0$, it is undecidable who is the winner of DE_δ.
  - Immediate reduction to universality problem of non deterministic mean payoff automaton.

$\bigwedge$ *MultiDimensionalMeanPayoff* games

- The game arena is a graph with multi dimensional weight function.
  - $w : E \to \mathbb{Z}^k$.
- Every play π has $2k$ dimensional mean payoff **vector** $\overrightarrow{MP} = (\underline{MP}(\pi)_1, \dots, \underline{MP}(\pi)_k, \overline{MP}(\pi)_1, \dots, \overline{MP}(\pi)_k)$

$\bigwedge$ *MultiDimensionalMeanPayoff* games

- The game arena is a graph with multi dimensional weight function.
  - $w : E \rightarrow \mathbb{Z}^k$.
- Every play π has $2k$ dimensional mean payoff **vector** $\overrightarrow{MP} = (\underline{MP}(\pi)_1, \ldots, \underline{MP}(\pi)_k, \overline{MP}(\pi)_1, \ldots, \overline{MP}(\pi)_k)$
- Winning conditions

$\bigwedge$ *MultiDimensionalMeanPayoff* games

- The game arena is a graph with multi dimensional weight function.
  - $w : E \to \mathbb{Z}^k$.
- Every play π has $2k$ dimensional mean payoff **vector** $\overrightarrow{MP} = (\underline{MP}(\pi)_1, \ldots, \underline{MP}(\pi)_k, \overline{MP}(\pi)_1, \ldots, \overline{MP}(\pi)_k)$
- Winning conditions
  - $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ games
    - OUTPUT must ensure $\bigwedge_{i=1}^{k}(\underline{MP}(\pi)_i \geq 0)$.

$\bigwedge$ *MultiDimensionalMeanPayoff* games

- The game arena is a graph with multi dimensional weight function.
  - $w : E \to \mathbb{Z}^k$.
- Every play π has $2k$ dimensional mean payoff **vector** $\overrightarrow{MP} = (\underline{MP}(\pi)_1, \ldots, \underline{MP}(\pi)_k, \overline{MP}(\pi)_1, \ldots, \overline{MP}(\pi)_k)$
- Winning conditions
  - $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ games
    - OUTPUT must ensure $\bigwedge_{i=1}^{k}(\underline{MP}(\pi)_i \geq 0)$.
  - $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ games
    - OUTPUT must ensure $\bigwedge_{i=1}^{k}(\overline{MP}(\pi)_i \geq 0)$.

$\bigwedge$ *MultiDimensionalMeanPayoff* games

- The game arena is a graph with multi dimensional weight function.
  - $w : E \to \mathbb{Z}^k$.
- Every play π has $2k$ dimensional mean payoff **vector** $\overrightarrow{MP} = (\underline{MP}(\pi)_1, \ldots, \underline{MP}(\pi)_k, \overline{MP}(\pi)_1, \ldots, \overline{MP}(\pi)_k)$
- Winning conditions
  - $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ games
    - OUTPUT must ensure $\bigwedge_{i=1}^{k}(\underline{MP}(\pi)_i \geq 0)$.
  - $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ games
    - OUTPUT must ensure $\bigwedge_{i=1}^{k}(\overline{MP}(\pi)_i \geq 0)$.
  - $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0) \wedge \bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ games
    - For $S \subseteq \{1, \ldots, k\}$ OUTPUT must ensure $\bigwedge_{i \in S}(\underline{MP}(\pi)_i \geq 0) \wedge \bigwedge_{i \in \overline{S}}(\overline{MP}(\pi)_i \geq 0)$.

$\bigwedge$ *MultiDimensionalMeanPayoff* games - cont

- Chatterjee, Doyen, Henzinger & Raskin 10:
  - When OUTPUT is restricted to finite memory strategy
    - Objectives $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ and $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ coincide.
    - Deciding whether OUTPUT is the winner is coNP complete.
    - If INPUT is the winner, he has a **memoryless** winning strategy.

$\bigwedge$ *MultiDimensionalMeanPayoff* games - cont

- Chatterjee, Doyen, Henzinger & Raskin 10:
  - When OUTPUT is restricted to finite memory strategy
    - Objectives $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ and $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ coincide.
    - Deciding whether OUTPUT is the winner is coNP complete.
    - If INPUT is the winner, he has a **memoryless** winning strategy.
  - Open question: Decidability and complexity of who is the winner of $\bigwedge$ *MultiDimensionalMeanPayoff* (when strategy of OUTPUT is not restricted).

$\bigwedge$ *MultiDimensionalMeanPayoff* games - cont

- Chatterjee, Doyen, Henzinger & Raskin 10:
  - When OUTPUT is restricted to finite memory strategy
    - Objectives $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ and $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ coincide.
    - Deciding whether OUTPUT is the winner is coNP complete.
    - If INPUT is the winner, he has a **memoryless** winning strategy.
  - Open question: Decidability and complexity of who is the winner of $\bigwedge$ *MultiDimensionalMeanPayoff* (when strategy of OUTPUT is not restricted).
- We answer this question.

$\bigwedge$ *MultiDimensionalMeanPayoff* games

Theorem

- For $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ games
  - Deciding whether OUTPUT is the winner is in NP $\cap$ coNP.

$\bigwedge$ *MultiDimensionalMeanPayoff* games

Theorem

- For $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ games
  - Deciding whether OUTPUT is the winner is in NP $\cap$ coNP.
- For $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ games
  - Deciding whether OUTPUT is the winner is coNP complete.

$\bigwedge$ *MultiDimensionalMeanPayoff* games

Theorem

- For $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ games
  - Deciding whether OUTPUT is the winner is in NP $\cap$ coNP.
- For $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ games
  - Deciding whether OUTPUT is the winner is coNP complete.
- For $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0) \wedge \bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ games
  - Deciding whether OUTPUT is the winner is coNP complete.

$\bigwedge$ *MultiDimensionalMeanPayoff* games

Theorem

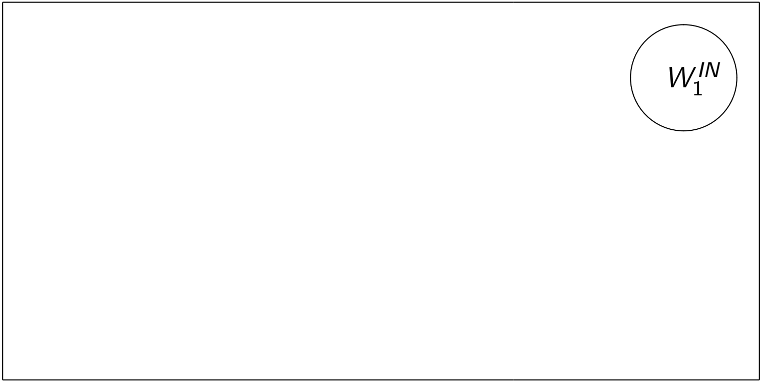
- For $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ games
  - Deciding whether OUTPUT is the winner is in NP $\cap$ coNP.
- For $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ games
  - Deciding whether OUTPUT is the winner is coNP complete.
- For $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0) \wedge \bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ games
  - Deciding whether OUTPUT is the winner is coNP complete.
- For all above games
  - If INPUT is the winner, he has a memoryless winning strategy.

Determine the winner of $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ game

Proof by example for 2 dimensions MP game

Determine the winner of $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ game

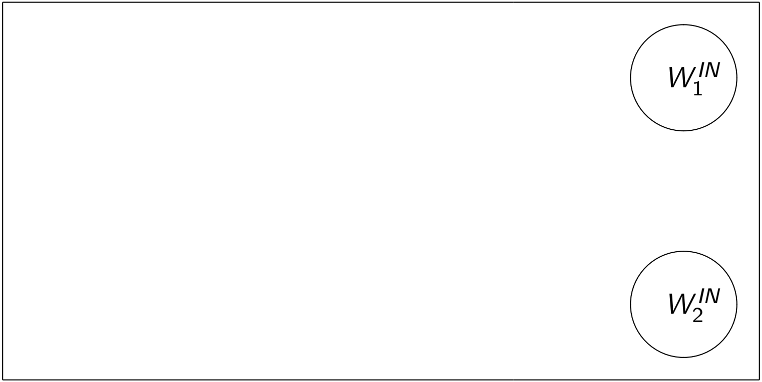
Proof by example for 2 dimensions MP game



- W_1^{IN} - INPUT winning region for dimension 1

Determine the winner of $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ game

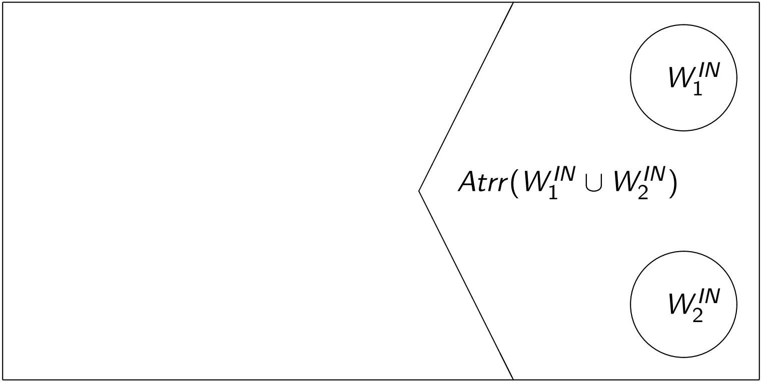
Proof by example for 2 dimensions MP game



- W_1^{IN} - INPUT winning region for dimension 1
- W_2^{IN} - INPUT winning region for dimension 2

Determine the winner of $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ game

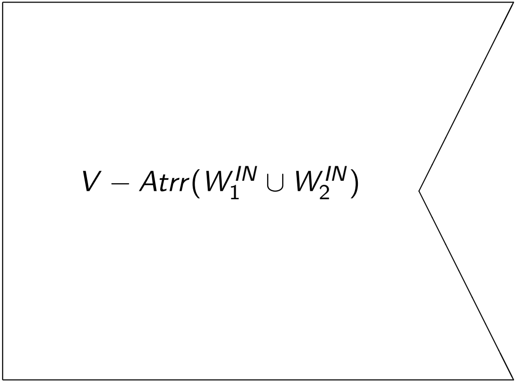
Proof by example for 2 dimensions MP game



- W_1^{IN} - INPUT winning region for dimension 1
- W_2^{IN} - INPUT winning region for dimension 2
- INPUT wins in $Atrr(W_1^{IN} \cup W_2^{IN})$

Determine the winner of $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ game

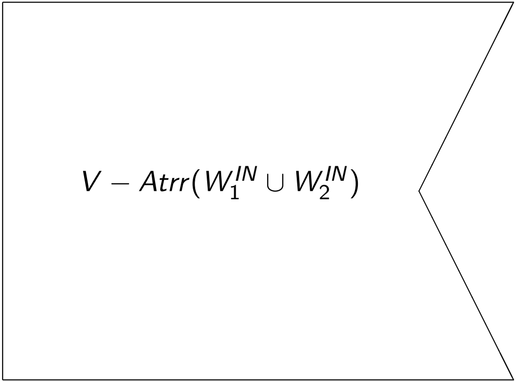
Proof by example for 2 dimensions MP game



- W_1^{IN} - INPUT winning region for dimension 1
- W_2^{IN} - INPUT winning region for dimension 2
- INPUT wins in $Atrr(W_1^{IN} \cup W_2^{IN})$
- Continue computation on subgame $V - Atrr(W_1^{IN} \cup W_2^{IN})$

Determine the winner of $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ game

Proof by example for 2 dimensions MP game



- W_1^{IN} - INPUT winning region for dimension 1
- W_2^{IN} - INPUT winning region for dimension 2
- INPUT wins in $Atrr(W_1^{IN} \cup W_2^{IN})$
- Continue computation on subgame $V - Atrr(W_1^{IN} \cup W_2^{IN})$
  - If $Atrr(W_1^{IN} \cup W_2^{IN}) = \emptyset$, OUTPUT wins in V

Determine the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ game

Determine the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ game

- **Lemma 1**: OUTPUT is the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ $\Leftrightarrow$ For every $\alpha > 0$, OUTPUT has a finite memory wining strategy in $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(-\alpha)$.

Determine the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ game

- **Lemma 1**: OUTPUT is the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ $\Leftrightarrow$ For every $\alpha > 0$, OUTPUT has a finite memory wining strategy in $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(-\alpha)$.
- When OUTPUT is restricted to finite memory strategies
  - **Lemma 2**: If INPUT is the winner he has a memoryless winning strategy Chatterjee, Doyen, Henzinger & Raskin 10.

Determine the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ game

- **Lemma 1**: OUTPUT is the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ $\Leftrightarrow$ For every $\alpha > 0$, OUTPUT has a finite memory wining strategy in $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(-\alpha)$.
- When OUTPUT is restricted to finite memory strategies
  - **Lemma 2**: If INPUT is the winner he has a memoryless winning strategy Chatterjee, Doyen, Henzinger & Raskin 10.
- **Lemma 3**: If INPUT is the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$, it has a memoryless winning strategy.

Determine the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ game

- **Lemma 1**: OUTPUT is the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ $\Leftrightarrow$ For every $\alpha > 0$, OUTPUT has a finite memory wining strategy in $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(-\alpha)$.
- When OUTPUT is restricted to finite memory strategies
  - **Lemma 2**: If INPUT is the winner he has a memoryless winning strategy Chatterjee, Doyen, Henzinger & Raskin 10.
- **Lemma 3**: If INPUT is the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$, it has a memoryless winning strategy.
- **Lemma 4**: One can verify in polynomial time if INPUT memoryless strategy is a winning strategy in $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$.

Determine the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ game

- **Lemma 1**: OUTPUT is the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ $\Leftrightarrow$ For every $\alpha > 0$, OUTPUT has a finite memory wining strategy in $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(-\alpha)$.
- When OUTPUT is restricted to finite memory strategies
  - **Lemma 2**: If INPUT is the winner he has a memoryless winning strategy Chatterjee, Doyen, Henzinger & Raskin 10.
- **Lemma 3**: If INPUT is the winner of $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$, it has a memoryless winning strategy.
- **Lemma 4**: One can verify in polynomial time if INPUT memoryless strategy is a winning strategy in $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$.
- **Corollaries**:
  - Deciding whether OUTPUT is the winner is in coNP
  - coNP hardness follows from Chatterjee, Doyen, Henzinger & Raskin 10 $\Rightarrow$ The problem is coNP complete.

Restricted weights

When weights are restricted to $\{-1, 0, +1\}$

- Deciding whether OUTPUT is the winner for $\bigwedge \mathrm{MeanPayoffSup}^{\geq}(0)$ condition is in P
- Deciding whether OUTPUT is the winner for $\bigwedge \mathrm{MeanPayoffInf}^{\geq}(0)$ condition is coNP hard

Games with undetected errors

- Two players game.
- ω language $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$
- In every round
  - Player INPUT plays with 0, 1 (every move is possibly an undetected error).
  - Player OUTPUT responds with 0 or 1
- Infinite play forms (IN, OUT).

Games with undetected errors

- Two players game.
- ω language $L \subseteq \{0,1\}^\omega \times \{0,1\}^\omega$
- In every round
  - Player INPUT plays with 0, 1 (every move is possibly an undetected error).
  - Player OUTPUT responds with 0 or 1
- Infinite play forms (IN, OUT).
- OUTPUT wins if for every $X \in \{0,1\}^\omega$:
  - X has "too many errors" ($X(i)$ has error if $X(i) \neq IN(i)$).
  - $(X, OUT) \in L$

Games with undetected errors - cont

- Error count: $EC(IN, X, n)$ number of positions until position n where $X(i) \neq IN(i)$
- Error rate: $ER(IN, X) = \lim_{n \to \infty} \sup\{\frac{1}{k} EC(IN, X, k) | k\}$
- Thresholds
  - UDE_δ - error rate δ
  - UDE_{fin} - finite number of errors
  - UDE_n - up to n errors
- Bounded number of errors problem - is there exists $n \in \mathbb{N}$ s.t INPUT is the winner of UDE_n?

Regular games with undetected errors

- Theorem

Regular games with undetected errors

- Theorem
  - It is decidable who is the winner of UDE_{fin} and UDE_n
    - Immediate reduction to regular games.

Regular games with undetected errors

- Theorem
  - It is decidable who is the winner of UDE_{fin} and UDE_n
    - Immediate reduction to regular games.
  - For $\delta > 0$ it is undecidable who is the winner of UDE_δ
    - Reduction to the universality problem of non deterministic mean payoff automaton.

Regular games with undetected errors

- Theorem
  - It is decidable who is the winner of UDE_{fin} and UDE_n
    - Immediate reduction to regular games.
  - For $\delta > 0$ it is undecidable who is the winner of UDE_δ
    - Reduction to the universality problem of non deterministic mean payoff automaton.
- Open questions
  - Deciding the winner of UDE_δ for $\delta = 0$
  - The bounded number of errors problem.

Conclusion

1. We proved decidability and complexity of who is the winner of $\bigwedge$ *MultiDimensionalMeanPayoff* games.
2. We obtained the following results for error games:

	Bounded		Fin		$\delta = 0$		$\delta \in (0, 1)$		$\delta = 1$	
	Par	MP	Par	MP	Par	MP	Par	MP	Par	MP
DE	✓	✓	✓	✓	✓	✓	✓	X	✓	X
UDE	?	?	✓	?	?	?	X	X	✓	X

✓ - decidable. X - undecidable. ? - Open.

Thank you