Multi-dim. Quantitative Games
Multi-dim. weighted games

- \( \mathbb{Z}^k \)-weighted game arena \( A=(S_1,S_2,s_{\text{init}},E,w) \) with \( w:E \rightarrow \mathbb{Z}^k \)

- Each play \( \rho=s_0e_0s_1e_1 \ldots s_ne_n s_{n+1} \ldots \) in the weighted arena defines an infinite sequence of vectors \( w(e_0) \ w(e_1) \ldots w(e_n) \ldots \in (\mathbb{Z}^k)^\omega \)

- Given a measure function \( f \in \{\text{Inf}, \text{Sup}, \text{Lim-Inf}, \text{Lim-Sup}, \text{MP-Inf}, \text{MP-Sup}\} \), we associate a vector in \( \mathbb{R}^k \) by applying \( f \) on each dimension, i.e.

\[
\text{given a play } \rho=s_0e_0s_1e_1 \ldots s_ne_n s_{n+1} \ldots ,
\text{with weight vectors } w(e_0) \ w(e_1) \ldots w(e_n) \ldots \in (\mathbb{Z}^k)^\omega ,
\]

\[
f(\rho) \in \mathbb{R}^k, \text{ s.t. for all dimensions } i, 1 \leq i \leq k ,
\]

\[
f(\rho)(i)=f(w(e_0)(i) \ w(e_1)(i) \ldots w(e_n)(i) \ldots)
\]
Threshold problems multi-dim.

Given a \( \mathbb{Z}^k \)-weighted game arena \( A \), a state \( s \), a measure \( f \), and a vector of thresholds \( v \in \mathbb{Q}^k \), the threshold problem asks

\[
\text{if Player 1 has a strategy } \lambda_1 \in \Lambda_1 \text{ such that } \quad \text{Outcome}_A(s, \lambda_1) \subseteq \{ \rho \mid f(\rho) \geq v \}
\]

or equivalently, if Player 1 has a strategy \( \lambda_1 \in \Lambda_1 \) s.t.

for all strategies \( \lambda_2 \in \Lambda_2 \) of Player 2, for each dim. \( i \),

\[
f(\text{Outcome}_A(s, \lambda_1, \lambda_2))(i) \geq v(i)
\]

For a state \( s \) of a multi-dimensional weighted arenas, there is not necessarily a unique minimal/maximal value vector that Player 1 can force. So the right concept for those games is not the notion of optimal value but rather the notion of \textbf{Pareto optimal} value and the notion of \textbf{Pareto curve}. 
Multi-Dimensional
Inf, Sup, Lim-Inf, Lim-Sup
Games
Multi-dimensional Sup games

- Given a $\mathbb{Z}^k$-weighted game arena $A$ and a threshold $v \in \mathbb{Q}^k$, Player 1 has a strategy to win the objective

$$\{ \rho \in \text{Plays}(A) \mid \text{Sup}(\rho) \geq v \}$$

if he can force to \textit{reach, for each dimension} $i$ ($1 \leq i \leq k$), an edge with value $w(e)(i) \geq v(i)$

- It is thus equivalent to a \textit{conjunction} of reachability objectives

- \textbf{Generalized reachability games} are defined as follows:

Given a game arena with $\{T_1, T_2, \ldots, T_k\}$, where each $T_i \subseteq S$, the generalized reachability problem is defined by $A=(S_1, S_2, E, s_{init}, \text{Win}_1 = \cap_{i=1,2,\ldots,k} \text{reach}(T_i))$, and asks: if there exists a strategy $\lambda_1$ for Player 1 that forces a visit to each set $T_i \subseteq S$
From multi-reach. to reachability

• Idea: record which sets $T_i$ has been visited so far

• So the state space of the new game arena is $S \times 2^{\{1,2,\ldots,k\}}$, and the target set is $\{(s,\{1,2,\ldots,k\}) \mid s \in S\}$

• It can be shown that if Player 1 can win a generalized reachability objective then he can ensure a visit to each set $T_i$ in at most $k \times |S|$ steps

• As a corollary, we get that the generalized reachability problem can be solved in $\text{APTIME}(O(|S|))$, or equivalently it is in $\text{PSPACE}$

• $\text{PSPACE-Hardness}$ can be established with a reduction from QBF

• Details about PSpace completeness can be found in https://www.cis.upenn.edu/~alur/Concur03g.pdf or http://arxiv.org/pdf/1010.2420.pdf
Multi-dimensional Inf games

• Given a $\mathbb{Z}^k$-weighted game arena $A$ and a threshold $v \in \mathbb{Q}^k$, Player 1 has a strategy to win the objective

$$\{\rho \in \text{Plays}(A) \mid \text{Inf}(\rho) \geq v\}$$

if he can avoid edges in $\bigcup_{i=1,2,...,k} \{e \mid w(e)(i) < v(i)\}$

• It is thus equivalent to a (simple) safety objective

• … so multi-dimensional Inf games can be solved in linear time
Multi-dimensional **Lim-Sup** and **Lim-Inf** games

- Exercises:
  - Show how to reduce a multi **Lim-Sup** game to a generalized **Büchi** game
  - Show how to solve generalized **Büchi** game in polynomial time
    
    **Hint:** compute for each Büchi objective the winning states of Player 1, intersects, and recurse on the sub-arena defined by the intersection
  
  - How much memory does Player 1 need to win a generalized Büchi game? What is the complexity of the recursive algorithm?
  
  - Show how to reduce a multi-dim. **Lim-Inf** game to a **coBüchi** game
Multi-dim. Mean-payoff and Energy Games
∃(C₁, C₂) ∈ \mathbb{N}^2 \text{ and } \lambda_1 \text{ s. t. } \text{Outcome}_A(q_0, \lambda_1, (C_1, C_2)) \models \square \ EL_1 \geq 0 \land EL_2 \geq 0.
Multi-dim. Energy Games

For any \((C_1, C_2) \geq (2, 1)\), Player 1 has a winning strategy.

Player 1 needs memory! How much?

\[\exists (C_1, C_2) \in \mathbb{N}^2 \text{ and } \lambda_1 \text{ s. t. } \mathbf{Outcome}_A(q_0, \lambda_1, (C_1, C_2)) \models \Box \mathbf{EL}_1 \geq 0 \land \mathbf{EL}_2 \geq 0.\]
Complexity of Multi-dim. Energy Games [CDHR10]
**Theorem.** If Player 2 has a winning strategy in a multi-energy game then he has a **memoryless** winning strategy.

**Proof.** By induction on the number of states in which Player 2 has a choice.

Let us consider a state $s \in S_2$ with a choice (between L(eft) and R(ight)). By induction hypothesis, in $A_L$ (the arena without the choice R in $s$), and $A_R$ (the arena without the choice L in $s$), Player 2 has a winning strategy iff he has a **memoryless** winning strategy. If Player wins either $A_L$ or $A_R$, his memoryless strategy in either $A_L$ or $A_R$, is also winning in $A$, and we are done. So we can assume that Player 2 has no winning strategy in $A_L$ nor in $A_R$. Now, let us consider the game $A$ and let us show that if he has no memoryless strategy in $A$ then he has no winning strategy at all.
Let $C_L$ and $C_R$ be the initial vectors of energy that are sufficient for Player 1 to win in $A_L$ and $A_R$ with strategies $\lambda_L$ and $\lambda_R$ respectively.

We claim that Player 1 has a winning strategy in $A$ against all possible strategies of Player 2 from initial energy level $C_L+C_R$. To prove that we construct a strategy $\lambda_1$ from $\lambda_R$ and $\lambda_L$ that wins from that initial credit:

- play first $\lambda_L$ up to a first visit to $s$,
- during $L$ periods, play as $\lambda_L$ and consider as histories the concatenation of the previous $L$ periods only,
- during $R$ periods, play as $\lambda_R$ and consider as histories the concatenation of the first $L$ period and the subsequent $R$ periods.

Clearly, as $\lambda_L$ wins in $A_L$, during $L$ periods, the running sum stays always above $-C_L$, and as $\lambda_R$ wins in $A_R$ during $R$ the running sum also always stays above $-C_R$. So globally, the running sum stays always above $-C_L-C_R$, and so Player 1 wins from initial credit $C_L+C_R$. 
**Theorem** [Kosaraju, Sullivan 88]. Given a $\mathbb{Z}^k$-weighted graph $G$, it is decidable in deterministic polynomial time if $G$ contains a state $s$ which is reachable from itself with a (not necessarily simple) cycle with zero effect on all dimensions.
Lemma. The unknown initial credit problem in MEGs is in coNP.

Proof.

- **Memoryless** strategies are sufficient for Player 2 to win a multi-dim. EG.
- Let $\lambda_2 \in \Sigma_2$ be a memoryless strategy for Player 2, and $A(\lambda_2)$ be the arena $A$ where only the choices of Player 2 that are compatible with $\lambda_2$ are kept, then $A(\lambda_2)$ is a $\mathbb{Z}^k$-weighted graph.
- $\lambda_2$ is winning for Player 2 (and so losing for Player 1) \textbf{iff} $A(\lambda_2)$ does not contain a reachable cycle (not necessarily simple) with non-negative effect on all dimensions (this later problem can solved in polynomial time).
- So, to show that Player 1 cannot win, we guess a memoryless strategy for Player 2 and verify in polynomial time that this strategy is winning for Player 2.
Lemma. The unknown initial credit problem in MEGs is \textbf{coNP-Hard}. \\

Proof. We show that deciding whether Player 1 has a winning strategy is as hard as deciding if a 3CNF formula is \textbf{unsatisfiable}. \\

Let $\psi$ be a 3CNF formula with clauses $C_1, C_2, \ldots, C_k$ over variables $\{x_1, x_2, \ldots, x_n\}$. We construct from $\psi$ the following game structure with weight in $\mathbb{Z}^{2n}$:
Complexity of multi-dim. EG

Ex: $\Phi = (x \lor \neg y \lor z) \land (\neg x \lor y \lor \neg z)$
Complexity of multi-dim. EG

We define the weight labelling as follows:

- every edge is labeled by $\{0\}^2n$ with the exception of edges going from literals back to initial state.
- for a literal $y$ and an edge back to the initial state, the weight vector contains:
  - 1 in the dimension of $y$
  - -1 in the dimension of the complement of $y$
  - 0 otherwise

Ex: $\Phi = (x \lor \neg y \lor z) \land (\neg x \lor y \lor \neg z)$
Complexity of multi-dim. EG

Let \( v \) be s.t. \( v \models \Phi \). We construct \( \lambda_2 \) as follows: in each clause \( C_i \), \( \lambda_2 \) chooses \( l_{ij} \) s.t. \( v \models l_{ij} \). Now, take any \( \lambda_1 \) and consider the play consistent with \( \lambda_1 \) and \( \lambda_2 \). There must exist \( C_i \) that appears \( \infty \)-often along this play: the dimension that correspond to \( \neg l_{ij} \) is decreased \( \infty \)-often without ever being increased! There is no initial credit that can help Player 1! So Player 2 wins.

Ex: \( \Phi = ( x \lor \neg y \lor z ) \land ( \neg x \lor y \lor \neg z ) \)

\( v(x)=1, v(y)=1, v(z)=0 \)

\( v \models \Phi \)

\( \lambda_2(C_1)=x \)

\( \lambda_2(C_2)=y \)
Φ is unsatisfiable implies Player 1 is winning
or equivalently Φ is unsatisfiable implies Player 2 is not winning

As Φ is unsatisfiable, when Player 2 chooses one literal per clause (we know that he can play optimally without memory), it has to choose two literals that are complementary. Let assume that the choice of Player 2 are complementary for clauses $C_i$ and $C_j$. In that case, the counter-strategy for Player 1 is to alternate between $C_i$ and $C_j$. This strategy is winning for a initial credit of 1 in all dimension.

$$\Phi = (\neg y) \land (x \lor y) \land (\neg x)$$
Complexity of multi-dim. EG

Φ is unsatisfiable implies Player 1 is winning
or equivalently Φ is unsatisfiable implies Player 2 is not winning

As Φ is unsatisfiable, when Player 2 chooses one literal per clause (we know that he can play optimally without memory), it has to choose two literals that are complementary. Let assume that the choice of Player 2 are complementary for clauses Cᵢ and Cⱼ. In that case, the counter-strategy for Player 1 is to alternate between Cᵢ and Cⱼ. This strategy is winning for a initial credit of 1 in all dimension.

\[
\Phi = (\neg y \land (x \lor y) \land \neg x)
\]

(1 -1 0 0)
+ (-1 1 0 0)
= (0 0 0 0)
Complexity of multi-dim. EG

Φ is unsatisfiable implies Player 1 is winning
or equivalently Φ is unsatisfiable implies Player 2 is not winning

As Φ is unsatisfiable, when Player 2 chooses one literal per clause (we know that he can play optimally without memory), it has to choose two literals that are complementary. Let assume that the choice of Player 2 are complementary for clauses $C_i$ and $C_j$. In that case, the counter-strategy for Player 1 is to alternate between $C_i$ and $C_j$. This strategy is winning for a initial credit of 1 in all dimension.

$\Phi = (\neg y) \land (x \lor y) \land (\neg x)$

$(1 -1 0 0) + (-1 1 0 0) = (0 0 0 0)$
Complexity of multi-dim. EG

To summarize:

**Theorem.** The unknown initial credit problem in MEGs is \textbf{coNP-C}. 
Exponential Memory is Sufficient for Player I in Multi-dim. EG [CRR12]
Player 1 - Finite Memory Strategies

Lemma. Finite memory strategies are sufficient for Player 1 to win in MEGs.
Lemma. Finite memory strategies are sufficient for Player 1 to win in MEGs.

Proof. First, \((\mathbb{N}^k, \leq)\) is a well-quasi ordered set, i.e.:

1. \(\leq\) is a partial order (so a pre-order)

2. for all infinite sequences of elements \(m_0 m_1 m_2 \ldots m_n \ldots\) in \((\mathbb{N}^k)^\omega\),

   there exists \(i < j\) such that \(m_i \leq m_j\)
** Lemma.** Finite memory strategies are sufficient for Player 1 to win in MEGs.

**Proof.** First, \((\mathbb{N}^k, \leq)\) is a well-quasi ordered set, i.e.:

Let \(\lambda_1\) be winning

On each branch

Then \(\lambda'_1\) is winning and finite memory

stop and play as from \(L_1\)!
Lemma. Finite memory strategies are sufficient for Player 1 to win in MEGs.

Proof. First, \((\mathbb{N}^k, \leq)\) is a well-quasi ordered set, i.e.:

Then \(\lambda_1'\) is winning and finite memory

Finite tree = winning strategy:

① play according to the choices made in tree

② in leaf, go to ancestor with lower or equal energy

wqo + Koenig’s lemma
Finite $\rightarrow$ Exponential memory

Then $\lambda_1'$ is winning and finite memory

$wqo+Koenig\text{'s lemma}$
Then $\lambda'$ is winning and finite memory...

$\Rightarrow$ wqo+Koenig's lemma...

Arbitrary weights:

1Exp

[BJK10]

Self-Covering Tree [BJK10]

$\Rightarrow$ 2Exp

3Exp

$\Rightarrow$ S

Self-Covering Tree

[BJK10]

$\Rightarrow$ 2Exp

3Exp
Then \( \lambda' \) is winning and finite memory...

Arbitrary weights:

\( 1 \text{Exp} \) [BJK10]

Self-Covering Tree [BJK10]

Depth: single exponential - encoding of arbitrary weights into \{-1,0,1\} does not add choices to the adversary.

Width: only energy level important (DAG).
Then $\lambda'$ is winning and finite memory...

$\text{1Exp}$

$\text{[BJK10]}$

Arbitrary weights:

$\text{2Exp}$

$\text{1Exp}$

$\text{3Exp}$ $\text{1Exp}$

**Depth:** single exponential - encoding of arbitrary weights into $\{-1,0,1\}$ does not add choices to the adversary.

**Width:** only energy level important (DAG).

Works also with parity
Finite $\rightarrow$ Exponential memory

•① Exponential memory is sufficient.
  - Use extensions of technics à la Rackoff (Petri nets) - refinements of [BJK10]

•② Exponential memory is needed

•③ Leads to symbolic and incremental algorithms
Finite memory \[\rightarrow\] Exponential memory

- Exponential memory is sufficient.
- Use extensions of technics à la Rackoff (Petri nets) - refinements of [BJK10]
- Exponential memory is needed
- Leads to symbolic and incremental algorithms

Lemma 3.

There exists a family of multi energy games $i G_i K_{1,1}$, $i S_1, S_2, s_{init}, E, k = t \cdot K, w_k$ s.t. for any initial credit, $P_1$ needs exponential memory to win.

The idea is the following: if $P_1$ does not remember the exact choices of $P_2$ which require an exponential size Moore machine, there will exist some sequence of choices of $P_2$ such that $P_1$ cannot counteract a decrease in energy. Thus, by playing this sequence long enough, $P_2$ can force $P_1$ to lose whatever his initial credit.

Fig. 2. Family of games requiring exponential memory:

Corollary 1.

Exponential memory is both sufficient and, in general, necessary for finite-memory winning on multi mean-payoff games. Synthesizing a winning strategy (if one exists) can be done in time exponential in the size of the game.

Proof. Thanks to [szn] Theorem we have equivalence between finite-memory winning for multi energy and multi mean-payoff games. The rest follows from straightforward application of Theorem and Lemma.

4 Randomness as a substitute for finite-memory

Throughout previous sections, we have been interested in pure finite-memory strategies for MEPGs and MMPPGs, as they are of importance for practical applications. As introduced in Section, another widely studied class of strategies is the class of randomized strategies. In this section, we answer a fundamental question regarding the nature of strategies: "When and how pure finite-memory can be traded for randomized memorylessness?" Specifically, we study on which kind of games $P_1$ can replace a pure finite-memory winning strategy by an equally powerful, yet conceptually simpler, randomized memoryless one and...
Finite \(\rightarrow\) Exponential memory

•① Exponential memory is sufficient.
  - Use extensions of technics à la Rackoff (Petri nets) - refinements of [BJK10]

•② Exponential memory is needed

•③ Leads to symbolic and incremental algorithms
Finite memory $\rightarrow$ Exponential memory

- Exponential memory is sufficient.
- Use extensions of technics à la Rackoff (Petri nets) - refinements of [BJK10]
- Exponential memory is needed.
- Leads to symbolic and incremental algorithms.

Lemma 3.

There exists a family of multi-energy games $G_i^{s_1, s_2, \text{init}, E, k}$ such that for any initial credit, $P_1$ needs exponential memory to win.

The idea is the following: if $P_1$ does not remember the exact choices of $P_2$ which require an exponential size Moore machine, there will exist some sequence of choices of $P_2$ such that $P_1$ cannot counteract a decrease in energy. Thus, by playing this sequence long enough, $P_2$ can force $P_1$ to lose whatever his initial credit.

Corollary 1.

Exponential memory is both sufficient and, in general, necessary for finite-memory winning on multi-mean-payoff parity games. Synthesizing a winning strategy (if one exists) can be done in time exponential in the size of the game.

Proof.

Thanks to Theorem we have equivalence between finite-memory winning for multi-energy and multi-mean-payoff games. The rest follows from straightforward application of Theorem and Lemma.

Randomness as a substitute for finite-memory

Throughout previous sections, we have been interested in pure finite-memory strategies for MEPGs and MMPPGs, as they are of importance for practical applications. As introduced in Section another widely studied class of strategies is the class of randomized strategies. In this section, we answer a fundamental question regarding the nature of strategies: "When and how pure finite-memory can be traded for randomized memorylessness?" Specifically, we study on which kind of games $P_1$ can replace a pure finite-memory winning strategy by an equally powerful yet conceptually simpler randomized memoryless one and

\[ C_{\text{max. constant appearing in the SCT}} \]

Incremental and symbolic algorithm

Minimal energy vectors that are winning (within $[0,2C]^k$)
Multi-dim. Mean-payoff Games
Two variants: LimSup - LimInf

- Lim Inf
- Lim Sup

- In the one dimension case, it does **not** make a difference because memoryless optimal strategies always exist, so outcomes can be considered as **ultimately periodic** (and so the limit exists and MP-Sup and MP-Inf coincide when the limit exists).

- In the multi-dim. case, it **makes a difference** because optimal strategies may require **infinite memory**.
MMPGs - Infinite Memory

• **MP-Inf**: define the mean-payoff in each dimension as follows:

- Let \( \pi : \mathbb{N} \rightarrow \mathbb{Z}^2 \), we associate to \( \pi \) the pair \((u,v)\) where:
  
  - \( u = \lim \inf_{i \to \infty} \frac{1}{n} \times \sum_{i=0}^{n} \pi(i) \downarrow 1 \) %MP on first dim.
  
  - \( v = \lim \inf_{i \to \infty} \frac{1}{n} \times \sum_{i=0}^{n} \pi(i) \downarrow 2 \) %MP on second dim.

Consider the strategy that alternates visits to \( q_a \) and \( q_b \) such that after the \( n^{th} \) alternation, the self-loop on the visited state \( q \) (\( q \in \{q_a,q_b\} \)) is taken \( n \) times. This strategy achieves threshold \((1, 1)\) for **MP-Inf**, as the frequency of edges with \((0,0)\) goes to 0.
MMPGs - Infinite Memory

**MP-Inf**: define the mean-payoff in each dimension as follows:

Let $\pi : \mathbb{N} \rightarrow \mathbb{Z}^2$, we associate to $\pi$ the pair $(u,v)$ where:

$$u = \liminf_{i \to \infty} \frac{1}{n} \times \sum_{i=0..n} \pi(i) \downarrow 1 \quad \% \text{MP on first dim.}$$

$$v = \liminf_{i \to \infty} \frac{1}{n} \times \sum_{i=0..n} \pi(i) \downarrow 2 \quad \% \text{MP on second dim.}$$
MMPGs - Infinite Memory

• **MP-Sup**: define the mean-payoff in each dimension as follows:

• Let \( \pi : \mathbb{N} \rightarrow \mathbb{Z}^2 \), we associate to \( \pi \) the pair \((u,v)\) where:

  - \( u = \limsup_{i \to \infty} \frac{1}{n} \times \sum_{i=0}^{n} \pi(i) \downarrow 1 \)  %MP on first dim.

  - \( v = \limsup_{i \to \infty} \frac{1}{n} \times \sum_{i=0}^{n} \pi(i) \downarrow 2 \)  %MP on second dim.

Consider the strategy that alternates visits to \( q_a \) and \( q_b \) such that after the \( n^{th} \) alternation, the self-loop on the visited state \( q \) (\( q \in \{q_a,q_b\} \)) is taken so many times that the average frequency of \( q \) gets larger than \((n-1)/n\) in the current finite prefix of the play. This is always possible and achieves threshold \((2, 2)\) for **MP-Sup**.
MMPGs - Infinite Memory

• **MP-Sup**: define the mean-payoff in each dimension as follows:

• Let $\pi : \mathbb{N} \rightarrow \mathbb{Z}^2$, we associate to $\pi$ the pair $(u,v)$ where:

  - $u = \limsup_{i \to \infty} \frac{1}{n} \times \sum_{i=0..n} \pi(i) \downarrow 1$  \quad \%MP on first dim.

  - $v = \limsup_{i \to \infty} \frac{1}{n} \times \sum_{i=0..n} \pi(i) \downarrow 2$  \quad \%MP on second dim.
MMPGs - Lim-sup

Lemma (strategy switch). If for all states $s \in S_1 \cup S_2$, for all $i$, $1 \leq i \leq k$, Player 1 has a winning strategy for winning MP-Sup for dimension $i$, then for all states $s \in S_1 \cup S_2$, Player I has a winning strategy from $s$ for the conjunction of all $k$ mean-payoff objectives.

Intuition: play each of the $k$ winning strategies one after the other for longer and longer time intervals.

Exercise. Write a detailed proof of the lemma.
Consider the following algorithm:

1. Compute $W_i =$ set of states where Player 1 wins the 1 dim. game defined by dim. $i$

2. Let $W$ be the intersection of all $W_i$'s

3. Remove states that are not in $W$

**Repeat** until no states are removed. Let Win be the states that survived this process

Win dim. 1

Win dim. 2
MP-Sup - Algorithm

Win dim. 1

Win dim. 2
MP-Sup - Algorithm
MP-Sup - Algorithm

Intersection

Recurse
MP-Sup - Algorithm

... Win
**Theorem.** From all states in Win, Player 1 has a winning strategy for **MP-Sup**, this strategy may require *infinite memory*. From all states that are not in Win, Player 2 has a *memoryless* winning strategy.

**Proof (sketch).** For Player 1, Lemma strategy switch establishes the existence of a winning strategy that uses infinite memory (this is unavoidable in the example that we have developed above). For Player 2, if s has been removed during the first phase of the algorithm, then Player 2 has a memoryless winning strategy to force a negative MP-Sup for at least one dimension. If s has been removed during phase i of the algorithm, then Player 2 has a memoryless winning strategy to either reach a state that has been removed at an earlier phase or to win for at least one dimension.

**Exercice.** Write the details of the proof above.

**Corollary.** Deciding the threshold problem for multi-dim. **MP-Sup** is in **NP∩coNP**.

**Proof.** The set Win can be constructed by a polynomial number of calls to an **NP∩coNP** oracle that solves one dimensional MP games. **NP∩coNP** is *closed* under polynomial number of oracle calls.
Theorem. In a multi-dim. MP-Inf games Player 1 may need to use infinite memory while Player 2 has memoryless optimal strategies. Deciding the threshold problem for multi-dim. MP-Inf is coNP-complete.

- The proof for coNP hardness for multi-dim. energy game can be easily adapted to the MP-Inf case.

- The proof for coNP membership is a variation of the proof for multi-dim. energy games.

- Memoryless for Player 2 is a consequence of a general result due to Kopczynski that establishes the existence of optimal memoryless strategies for Player 2 whenever the objective of Player 1 is both prefix-independent and convex:


An objective $\phi$ is prefix-independent if for all plays $\pi$ and $\pi'$ such that $\rho'=\pi.\rho$, where $\pi$ is a finite prefix, we have $\rho\in\phi$ iff $\rho'\in\phi$. A play $\rho$ is a combination of two plays $\rho_1=uuu\ldots$ and $\rho_2=uuu\ldots$, where $u$'s are finite prefixes, if $\rho=uuuuuu\ldots$. An objective $\phi$ is convex if it is closed under combination.
## Summary

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>EG</td>
<td>Memoryless</td>
<td>OK</td>
<td>Memoryless</td>
<td>(NP \cap \text{coNP})</td>
</tr>
<tr>
<td>MP</td>
<td>Memoryless</td>
<td>OK</td>
<td>Memoryless</td>
<td>(NP \cap \text{coNP})</td>
</tr>
<tr>
<td>MEG</td>
<td>Exponential</td>
<td>OK</td>
<td>Memoryless</td>
<td>(\text{coNP-})C</td>
</tr>
<tr>
<td>MEG-finite</td>
<td>-</td>
<td>OK</td>
<td>Memoryless</td>
<td>(\text{NP-})C</td>
</tr>
<tr>
<td>MMPG - Sup</td>
<td>Infinite</td>
<td>OK</td>
<td>Memoryless</td>
<td>(NP \cap \text{coNP})</td>
</tr>
<tr>
<td>MMPG - Inf</td>
<td>Infinite</td>
<td>?</td>
<td>Memoryless</td>
<td>(\text{coNP-})C</td>
</tr>
<tr>
<td>MMPG - Mix</td>
<td>Infinite</td>
<td>?</td>
<td>Memoryless</td>
<td>(\text{coNP-})C</td>
</tr>
</tbody>
</table>
Qualitative games


References and further readings

Mean-payoff games


Energy games


Fixpoint algorithm for MP

References and further readings

**Multi-dimension MP and EG**


