As Soon as Possible but Rationally

Christophe Grandmont

UMONS Université de Mons

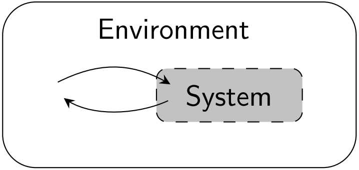
ULB UNIVERSITÉ LIBRE DE BRUXELLES

April 26 2024

Work with Véronique Bruyère and Jean-François Raskin

Topic: Reactive Systems

A **system** (player 0) interacting with the **environment** (player 1).



- System is **controllable** (coffee machine, elevator, autopilot, ...)
- Environment is not (humans, other independent systems, ...)

$\rightsquigarrow$ How to guarantee a **specification** Ω_0 in such reactive systems ?

Rational Synthesis

Guarantee Ω_0, in which condition ?

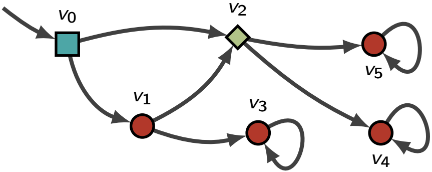
- **Zero-sum** ? Environment completely antagonistic... $\rightsquigarrow$ Too simple !
- Instead, assume that the environment is composed of **multiple components** $1, \ldots, t$, each with a specification Ω_i.
  $\rightsquigarrow$ The environment behaves **"rationally"** !

Rational Synthesis

The **Rational Synthesis** problem asks to decide whether there exists a strategy of player 0 such that every *rational* response of the environment *to this strategy* satisfies a goal Ω_0 of player 0.

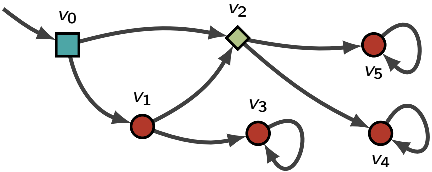
Game played on graphs

$\rightsquigarrow$ Model interactions with **games played on graphs** ...



... where the environment **plays "rationally"** !

Game played on graphs



$$\left.\begin{array}{l}\text{Directed graph: } (V, E) \\ \text{Set of players: } \mathcal{P} = \{0, \dots, t\} \\ \text{Partition of } V\text{: } (V_i)_{i \in \mathcal{P}}\end{array}\right\} \text{Arena } \mathcal{A} = (V, E, \mathcal{P}, (V_i)_{\in \mathcal{P}})$$

- Play: $\pi \in \text{Plays} \subseteq V^\omega$ consistent with E, history: $h \in V^*$,
- Strategy for $i \in \mathcal{P}$: function $\sigma_i : V^* V_i \to V$, $hv \mapsto \sigma_i(hv)$.

Game played on graphs: Reachability objective

Associate an objective with each player i: **Quantitative reachability** (T_i, w_i):

$T_i \subseteq V$, $w_i : E \to \mathbb{N}$, with $\text{cost}_i : \text{Plays} \to \mathbb{N} \cup \{+\infty\}$:

- for a play $\pi = \pi_0\pi_1 \dots$ and $n = \inf\{k \in \mathbb{N} \mid \pi_k \in T_i\}$,

$$\text{cost}_i(\pi) = \begin{cases} \sum_{k=1}^{n} w_i((\pi_{k-1}, \pi_k)) & \text{if } n < +\infty, \\ +\infty & \text{otherwise.} \end{cases}$$

(When $w_i(e) = 0$ for all $e \in E$ $\rightsquigarrow$ **Qualitative reachability**)

$\rightsquigarrow$ Goal of the system: $\text{cost}_0(\pi) \leq c$, for a given threshold $c \in \mathbb{N}$.

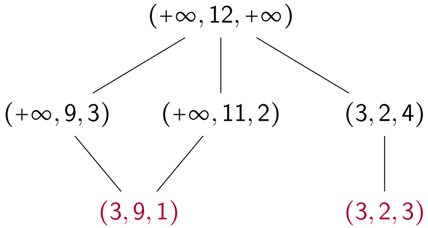
$\rightsquigarrow$ Goal of each player of the environment: **Minimize** cost_i, then be antagonistic towards the system.

What does "minimize" mean (i.e., be rational) ?

Pareto-Optimality

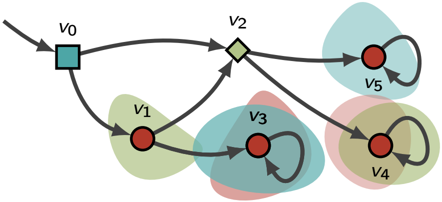
Stackelberg game: player 0 fixes σ_0, and the environment behaves rationally according to σ_0.
Here, players $1, \ldots, t$ agree to get ~~the~~ a lowest $\text{cost}_{\text{env}} = (\text{cost}_1, \ldots, \text{cost}_t)$:
Pareto-Optimality !

$\rightsquigarrow$ Partial order $\preceq$ on $\mathbb{N}^t$, e.g.



$P_{\sigma_0} = \min\{\text{cost}_{\text{env}}(\pi) \mid \pi \text{ play consistent with } \sigma_0\}$.

Pareto-Optimality: Example



- All weights constant, $w_i = 1$,
- $T_0 = \{v_3, v_4\}$,
- $T_1 = \{v_3, v_5\}$,
- $T_2 = \{v_1, v_4\}$.

For σ_0 such that $\sigma_0(v_1) = v_2$:

$$\left.\begin{array}{llll} v_0 v_1 v_2 v_5 & : \text{player 0: } +\infty, & \text{env: } (3,1) \\ v_0 v_1 v_2 v_4 & : \text{player 0: } 3, & \text{env: } (+\infty,1) \\ v_0 v_2 v_5 & : \text{player 0: } +\infty, & \text{env: } (2,+\infty) \\ v_0 v_2 v_4 & : \text{player 0: } 2, & \text{env: } (+\infty,2) \end{array}\right\} P_{\sigma_0} = \{(3,1),(2,+\infty)\}$$

For σ_0' such that $\sigma_0'(v_1) = v_3$:

$$\left.\begin{array}{llll} v_0 v_1 v_3 & : \text{player 0: } 2, & \text{env: } (2,1) \\ v_0 v_2 v_5 & : \text{player 0: } +\infty, & \text{env: } (2,+\infty) \\ v_0 v_2 v_4 & : \text{player 0: } 2, & \text{env: } (+\infty,2) \end{array}\right\} P_{\sigma_0} = \{(2,1),(+\infty,2)\}$$

Pareto Synthesis

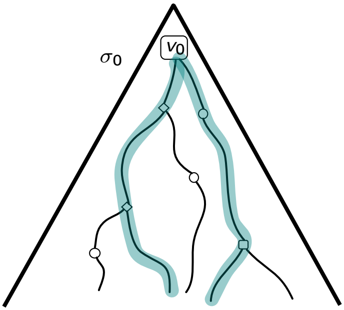
Let $\mathcal{G} = (V, E, V_0, (T_i, w_i)_{0 \le i \le k})$ be a game and $c \in \mathbb{N}$ be a threshold,

Non-Cooperative Pareto Synthesis (NCPS) problem:

$$\exists \sigma_0,\ \forall \pi \in \mathsf{Plays}_{\sigma_0},\quad \mathsf{cost}_{\mathsf{env}}(\pi) \in P_{\sigma_0} \Rightarrow \mathsf{cost}_0(\pi) \le c.$$

Cooperative Pareto Synthesis (CPS) problem:

$$\exists \sigma_0,\ \exists \pi \in \mathsf{Plays}_{\sigma_0},\quad \mathsf{cost}_{\mathsf{env}}(\pi) \in P_{\sigma_0} \land \mathsf{cost}_0(\pi) \le c.$$



Pareto Results

	Coop. synthesis	Non-coop. synthesis
Pareto, weights	PSPACE-**complete**	NEXPTIME-complete[1]
Pareto, qualitative	PSPACE-**complete**	NEXPTIME-complete[2]

Verification variants: given a strategy σ_0 defined by a **deterministic** Mealy machine, is σ_0 a solution to a synthesis problem ? (Is every σ_0 defined by a **nondeterministic** Mealy machine a solution ?)

	Non-coop. verif. (det.)	Non-coop. verif. (nondet.)
Pareto, weights	Π_2^P-**complete**	PSPACE-**complete**
Pareto, qualitative	Π_2^P-**complete**	PSPACE-**complete**

[1] Brihaye, Bruyère, and Reghem, "Quantitative Reachability Stackelberg-Pareto Synthesis is NEXPTIME-Complete", RP 2023

[2] Bruyère, Raskin, and Tamines, "Stackelberg-Pareto Synthesis", CONCUR 2021

Cooperative Synthesis: Sketch

PSPACE membership of the CPS problem:

$$\exists \sigma_0,\ \exists \pi \in \mathsf{Plays}_{\sigma_0},\quad \mathsf{cost}_{\mathsf{env}}(\pi) \in P_{\sigma_0} \wedge \mathsf{cost}_0(\pi) \leq c.$$

Approach by 3 steps:

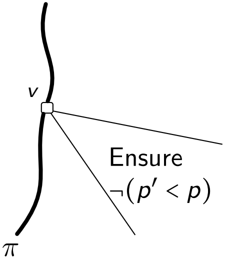
1. **Guess** a lasso $\pi = \mu(\nu)^\omega$, **(Is is sufficient ? Which size ?)**
2. **Check** whether $\mathsf{cost}_0(\pi) \leq c$, (Run through $\mu\nu$)
3. **Check** whether $\mathsf{cost}_{\mathsf{env}}(\pi) \in P_{\sigma_0}$ (for σ_0 to determine). **(How ?!)**

Lemma

For Step 1, polynomial-length lasso is sufficient.

Cooperative Synthesis: Sketch

3. Given $\pi = \mu(\nu)^\omega$, check whether $p = \mathsf{cost}_{\mathsf{env}}(\pi) \in P_{\sigma_0}$ for σ_0 to determine.



For each prefix h of π, player 0 must ensure

$$\Omega^{(h)} = \{\pi' \in \mathsf{Plays} \mid \neg(\mathsf{cost}_{\mathsf{env}}(h\pi') < p)\}, \text{ i.e.,}$$

$$\left(\forall 1 \le i \le t,\ p'_i \ge p_i\right) \ \vee\ \left(\exists 1 \le i \le t,\ p'_i > p_i\right).$$

It amounts to solve a zero-sum game $(\mathcal{G}, \Omega)$ with

$$\Omega = \left(\bigcap_{1\le i\le t} \mathsf{Safe}_{\ge d_i}(T_i)\right) \cup \left(\bigcup_{1\le i\le t} \mathsf{Safe}_{\ge d_i+1}(T_i)\right),$$

with **bounded safety objectives** $\mathsf{Safe}_{\ge d}(T) = \{\pi \mid \mathsf{cost}(\pi) \ge d\}$ $(d \in \mathbb{N})$

Cooperative Synthesis: Sketch

Proposition

Deciding the winning of a two-player zero-sum game $(\mathcal{G}, \Omega)$ with

$$\Omega = \left(\bigcap_{1 \leq i \leq t} \mathrm{Safe}_{\geq d_i}(T_i) \right) \cup \left(\bigcup_{1 \leq i \leq t} \mathrm{Safe}_{\geq d_i + 1}(T_i) \right).$$

is in PSPACE.

$\rightsquigarrow$ **The Cooperative Synthesis problem belongs to PSPACE !**

Non-cooperative Verification (deterministic): Sketch

The NCPV problem is in Π_2^{P} iff the coNCPV problem is in $\Sigma_2^{\mathsf{P}} = \mathsf{NP}^{\mathsf{NP}}$.

The coNCPV problem amounts to solve, in some game $\mathcal{G}'$ **where the environment is the only player**:

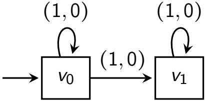
$$\exists \pi \in \mathsf{Plays}, \quad \mathsf{cost}_{\mathsf{env}}(\pi) \in P \wedge \mathsf{cost}_0(\pi) > c.$$

Goal: algorithm in $\mathsf{NP}^{\mathsf{NP}}$

1. **Guess** a lasso $\pi = \mu(\nu)^\omega$, **(Exponential length ☹)**
2. **Check** whether $\mathsf{cost}_0(\pi) > c$, **(Exponential length ☹)**
3. **Check** whether $\mathsf{cost}_{\mathsf{env}}(\pi) \in P$. **(Environment is alone, in coNP ☺)**

Non-cooperative Verification (deterministic): Sketch

Player 1 is the only one to play, $T_0 = T_1 = \{v_1\}$, $w_0 = 1$, and $w_1 = 0$.



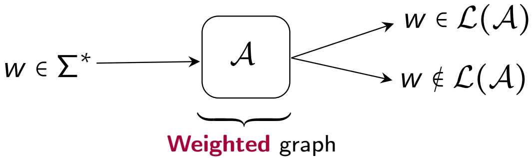
Solutions to the coNCPV problem are $\pi = (v_0)^k(v_1)^\omega$ for $k > c$, but c given in binary $\rightsquigarrow$ $|(v_0)^k v_1|$ **exponential.**

Non-cooperative Verification (deterministic): Sketch

Goal: Steps 1 and 2 with an NP algorithm:

1. Guess a lasso $\pi = \mu(\nu)^\omega \rightsquigarrow$ NP ?
2. Check whether $\text{cost}_0(\pi) > c$,
3. Check whether $\text{cost}_{\text{env}}(\pi) \in P_{\sigma_0} \rightsquigarrow$ coNP !

Solution: split π into sequences that we **succinctly guess** using **Parikh Automata**.



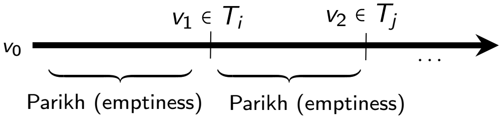
$\rightsquigarrow$ Parikh: w is accepted iff there exists a run ending in a final state $q \in F$ and the accumulated weight is in $C \subseteq \mathbb{N}^k$ (even with $C = \{\bar{c}\}$).

Non-cooperative Verification (deterministic): Sketch

Lemma

The nonemptiness problem of Parikh automata is NP*-complete.*

- We want π **such that** $\text{cost}(\pi) = \bar{c}$
- $\rightsquigarrow$ **Guess a lasso π of exponential length**:
  1. guess markers $v_1, \ldots, v_n$ belonging to some distinct target sets,
  2. guess costs $\bar{c}^{(i)}$ between v_i and v_{i+1},
  3. use Parikh automata to guess sequences $\rho^{(i)}$ from v_i and v_{i+1} with $\text{cost}(\rho^{(i)}) = \bar{c}^{(i)}$.



Non-cooperative Verification (deterministic): Sketch

To sum up:

1. **Succinctly** guess $\pi = \mu(\nu)^\omega$, $\text{cost}_0(\pi)$ such that $\text{cost}_0(\pi) > c$, and a cost tuple $p = \text{cost}_{\text{env}}(\pi)$ through multiple sequences, **(poly.)**
2. Check that $\text{cost}_{\text{env}}(\pi) \in P_{\sigma_0}$. **(NP Oracle)**

$\rightsquigarrow$ **Belongs to** $\text{NP}^{\text{NP}} = \Sigma_2^{\text{P}}$, i.e., the NCPV problem belongs to Π_2^{P}.

All results

Nash variants: instead of asking $\text{cost}_{\text{env}}(\pi) \in P_{\sigma_0}$, ask that π is the outcome of a **Nash Equilibrium**.

Restricted environment: e.g. one player in the environment.

	Coop. synthesis	Non-coop. synthesis	Non-coop. verif. (det.)	Non-coop. verif. (nondet.)
Pareto, weights	PSPACE-**complete**	NEXPTIME-complete[3]	Π_2^P-**complete**	PSPACE-**complete**
Pareto, qualitative	PSPACE-**complete**	NEXPTIME-complete[4]	Π_2^P-**complete**	PSPACE-**complete**
Nash, weights	NP-**complete**	Unknown (EXPTIME-**hard**)	coNP-**complete**	coNP-**complete**
Nash, qualitative	NP-complete[5]	PSPACE-complete[5]	coNP-**complete**[6]	coNP-**complete**[6]
Nash, weights, 1-env	NP-**complete**	EXPTIME **(PSPACE-hard)**	coNP	coNP-**complete**

Thank you !

[3] Brihaye, Bruyère, and Reghem, "Quantitative Reachability Stackelberg-Pareto Synthesis is NEXPTIME-Complete", RP 2023

[4] Bruyère, Raskin, and Tamines, "Stackelberg-Pareto Synthesis", CONCUR 2021

[5] Condurache et al., "The Complexity of Rational Synthesis", ICALP 2016

[6] Christophe Grandmont, Master's Thesis 2023