Fixpoints in VASS: Results and Applications

Arnaud Sangnier

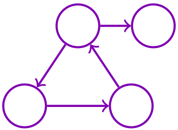
IRIF - Université Paris Diderot

joint works with : *Parosh A. Abdulla, Radu Ciobanu, Richard Mayr and Jeremy Sproston*

Gandalf'16 - 16th September 2016

Model-checking

Does **a system** **satisfies** **a specification ?**



Model

$\stackrel{?}{\models}$ φ

Formula

Challenges:

- Find expressive models
- Find logics to express interesting properties
- Find algorithms to solve the model-checking problem

Trade-off between efficiency and expressiveness

Examples of Models and Specification Languages

Models

- Finite State Systems
- Infinite State Systems
  - Türing machines
  - Timed Automata
  - Pushdown systems
  - Petri nets or Vector Addition System with States (VASS)

Logics

- **Linear Time Logics**
  - Linear Time Temporal Logic (LTL)
  - Büchi automata
  - Linear μ-calculus
  - First order logic over words
- **Branching Time Logics**
  - Computational Tree Logic (CTL)
  - μ-calculus

Examples of Models and Specification Languages

Models

- Finite State Systems
- Infinite State Systems
  - Türing machines
  - Timed Automata
  - Pushdown systems
  - **Petri nets or Vector Addition System with States (VASS)**

Logics

- **Linear Time Logics**
  - Linear Time Temporal Logic (LTL)
  - Büchi automata
  - Linear μ-calculus
  - First order logic over words
- **Branching Time Logics**
  - Computational Tree Logic (CTL)
  - μ-calculus

Modelling Uncertainty

Adding probabilities to models

- In pure probabilistic systems, like in Markov Chains, non-determinism is cancelled
- In some systems, like Markov Decision Processes (MDP), probabilities and non-determinism cohabit
  - These systems can be seen as a one and half player game
  - The first player, aka **scheduler**, resolves non-determinism and the other player is the probabilistic player

Specification in probabilistic systems

- **Qualitative specification**
  - Probabilities are only compared with 0 or 1
  - Is a state reached with probability 1 ?
  - Is the probability of seeing infinitely ofter a state strictly positive ?
- **Quantitative specification**
  - Is the probability of an event bigger then 0.6 ?

A Small Problem

- I have a certain number of mystery **black** balls
- When shining a ball, it becomes **red** or **green** with **probability one half each**
- I need at least **10** green balls to win
- At each round I can pick a ball and shine it
- **Question :** Is there an initial number of balls which allows me to win with probability one ?
- **Question :** What if at each round I can choose to increment the number of balls or to pick a ball ?

A Small Problem

- I have a certain number of mystery **black** balls
- When shining a ball, it becomes **red** or **green** with **probability one half each**
- I need at least **10** green balls to win
- At each round I can pick a ball and shine it
- **Question :** Is there an initial number of balls which allows me to win with probability one ?
- **Question :** What if at each round I can choose to increment the number of balls or to pick a ball ?

A Small Problem

- I have a certain number of mystery **black** balls
- When shining a ball, it becomes **red** or **green** with **probability one half each**
- I need at least **10** green balls to win
- At each round I can pick a ball and shine it
- **Question :** Is there an initial number of balls which allows me to win with probability one ?
- **Question :** What if at each round I can choose to increment the number of balls or to pick a ball ?

Which Ingredients to Solve the Problem ?

- I have a certain number of mystery **black** balls
  - ⇒ **Counting + non-deterministic guess**
- When shining a ball, it becomes **red** or **green** with **probability one half each**
  - ⇒ **Probabilities**
- I need at least **10** green balls to win
  - ⇒ **Test if a counter is greater than 10**

Which Ingredients to Solve the Problem ?

- I have a certain number of mystery **black** balls

  ⇒ **Counting + non-deterministic guess**

- When shining a ball, it becomes **red** or **green** with **probability one half each**

  ⇒ **Probabilities**

- I need at least **10** green balls to win

  ⇒ **Test if a counter is greater than 10**

Vector Addition System with States - Markov Decision Processes

Outline

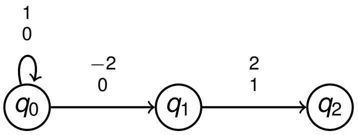
Outline

Vector Addition System with States (VASS)

VASS

A n-dim VASS $S = \langle Q, E, q_0 \rangle$:

- Q : finite set of control states
- $E \subseteq Q \times \mathbb{Z}^n \times Q$
- $q_0 \in Q$: initial control states



- **Set of configurations:** $Q \times \mathbb{N}^n$ - **No negative value allowed!!!**
- **Example of runs:**
$(q_0, (0, 0)) \rightarrow (q_0, (1, 0)) \rightarrow (q_0, (2, 0)) \rightarrow (q_1, (0, 0)) \rightarrow (q_2, (2, 1))$

Why to study VASS ?

- Models *equivalent* to Petri nets
- Infinite state systems with resources that can be incremented and decremented
- Many problems are decidable for VASS
- Methods developed for this model have been reused in other context
- Many theoretical tools available to analyse this model
- Extending VASS leads quickly to undecidable verification problems
- Strong link with some other formalisms like for instance logics with data

Classical Problems for VASS

Control State Reachability (aka *Coverability)*

- **Input:** A n-dim VASS S and a control state q_F
- **Output:** Does there exist $\mathbf{v} \in \mathbb{N}^n$ such that $(q_0, \mathbf{0}) \rightarrow^* (q_F, \mathbf{v})$?

Reachability

- **Input:** A n-dim VASS S and a configuration $(q_F, \mathbf{v}_F)$
- **Output:** Do we have $(q_0, \mathbf{0}) \rightarrow^* (q_F, \mathbf{v}_F)$?

Repeated Control State Reachability

- **Input:** A n-dim VASS S and a control state q_F
- **Output:** Does there exist infinite $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_i, \ldots \in \mathbb{N}^n$ such that $(q_0, \mathbf{0}) \rightarrow^* (q_F, \mathbf{v}_1) \rightarrow^+ (q_F, \mathbf{v}_2) \rightarrow^+ \cdots \rightarrow^+ (q_F, \mathbf{v}_i) \cdots$?

Well Quasi Ordering : a Nice Tool for VASS

Well Quasi Ordering (wqo)

$(X, \leq)$ is a well-quasi ordering if for all infinite sequences $s_1, s_2, \ldots$, there exists $i < j$ such that $s_i \leq s_j$.

Upward closed set

A set $Y \subseteq X$ is upward closed w.r.t $(X, \leq)$ if $y \in Y$ and $y \leq y'$ implies $y' \in Y$.

- Upward closure of $Y \subseteq X$: $\uparrow Y = \{x \in X \mid \exists y \in Y \ \wedge \ y \leq x\}$

Lemma

If $(X, \leq)$ is a wqo and if $Y \subseteq X$ is upward closed w.r.t. $(X, \leq)$, then there exists a finite set $B \subseteq X$ s.t. $Y = \uparrow B$.

Stabilization Lemma

If $(X, \leq)$ is a wqo and $(Y)_{i \in \mathbb{N}}$ is a sequence of upward-closed sets such that $Y_i \subseteq Y_{i+1}$, then there exists j such that $Y_{k+1} = Y_k$ for all $k > j$.

Properties of VASS

- Order on configurations of VASS:

$$(q, \mathbf{v}) \sqsubseteq (q', \mathbf{v}') \text{ iff } q = q' \text{ and } \mathbf{v} \leq \mathbf{v}'$$

Dickson's Lemma

$(Q \times \mathbb{N}^n, \sqsubseteq)$ is a wqo.

Monotonicity Lemma

If $(q_1, \mathbf{v}_1) \rightarrow (q_2, \mathbf{v}_2)$ and if $\mathbf{v}_1 \leq \mathbf{v}'_1$ then there exists $\mathbf{v}_2 \leq \mathbf{v}'_2$ such that $(q_1, \mathbf{v}'_1) \rightarrow (q_2, \mathbf{v}'_2)$

Consequences:

- For a set $C \subseteq Q \times \mathbb{N}^n$

$$Pre(C) = \{(q, \mathbf{v}) \mid \exists (q', \mathbf{v}') \in C \,.\, (q, \mathbf{v}) \rightarrow (q', \mathbf{v}')\}$$

- **If C is upward closed, then $Pre(C)$ is upward-closed**

Solving Control State Reachability in VASS

- Compute the following sequence of upward-closed sets
  - $C_0 = \uparrow \{(q_F, \mathbf{0})\}$
  - $C_{i+1} = C_i \cup Pre(C_i)$
- This computation is possible by reasoning always on the minimal elements (which are finite).
- By the Stabilization Lemma, there is $j \in \mathbb{N}$ such $C_{k+1} = C_k$ for all $k \geq j$.
- Test if $(q_0, \mathbf{0}) \in C_j$.

Solving Control State Reachability in VASS

- Compute the following sequence of upward-closed sets
  - $C_0 = \uparrow \{(q_F, \mathbf{0})\}$
  - $C_{i+1} = C_i \cup Pre(C_i)$
- This computation is possible by reasoning always on the minimal elements (which are finite).
- By the Stabilization Lemma, there is $j \in \mathbb{N}$ such $C_{k+1} = C_k$ for all $k \geq j$.
- Test if $(q_0, \mathbf{0}) \in C_j$.

This method is not optimal from the complexity point of view

Results

Theorem [Lipton'76,Rackoff'78]

Control State Reachability in VASS is EXPSPACE-complete.

- Use short sequences of doubly exponential length to witness control state reachability

Theorem [Kosaraju'82; Mayr'84]

Reachability in VASS is decidable.

- Non-primitive recursive algorithm
- Exact complexity is an open problem
- Shorter proof provided in [Leroux'11]

Theorem [Habermehl'97]

Repeated Control State Reachability in VASS is EXPSPACE-complete.

Linear Temporal Logics (LTL)

Syntax

$$\phi ::= q \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \mathrm{U} \phi \mid \mathrm{X}\phi$$

where $q \in Q$

- Models of LTL are infinite words ρ over a finite alphabet Q

Satisfaction relation

$$\begin{array}{rcl} \rho, i \models q & \stackrel{\text{def}}{\Leftrightarrow} & q_i = q \\ \rho, i \models \mathrm{X}\phi & \stackrel{\text{def}}{\Leftrightarrow} & i+1 < |\rho| \text{ and } \rho, i+1 \models \phi \\ \rho, i \models \phi_1 \mathrm{U} \phi_2 & \stackrel{\text{def}}{\Leftrightarrow} & \text{for some } i \leq j < |\rho|, \rho, j \models \phi_2 \\ & & \text{and for all } i \leq k < j, \rho, k \models \phi_1 \end{array}$$

Example of properties:

- **Liveness:** There is a run that visits infinitely often q
- **Safety:** The state q is never visited

Model-checking LTL in VASS

Model-Checking of LTL

- **Input:** A VASS and an LTL formula φ
- **Output:** Does there exist an infinite run ρ such that $\rho, 0 \models \varphi$?

Theorem [Habermehl'97]

Model-checking LTL on VASS is EXPSPACE-complete.

- Any LTL formula can be translated into a Büchi automaton [Vardi-Wolper'86] of exponential size
- Check repeated control state reachability in the product of the VASS and the automaton (where the automaton is parsed *on the fly*)

Can We Go Further ?

By extending the model

- VASS have only the ability to test if a counter is bigger than a value
- It is possible to add one counter that is tested to 0
- Doing more leads to undecidability

By considering more expressive linear time logics

- The proof technique presented before works for any temporal logic that recognizes ω-regular languages
- For instance, the model-checking of VASS with linear μ-calculus is EXPSPACE-complete

By considering branching time logics

- Whereas for finite state systems, model-checking CTL is easier than model-checking LTL
- In VASS, most of model-checking problems with branching time logics are undecidable
- **Anyway, we'll see there is some ways to overcome this issue**

An Annoying Undecidability Result

Minsky machine

- Manipulates two counters c_1 and c_2
- Finite set of labeled instructions of the form:
  1. $L : c_i := c_i + 1$; goto L'
  2. L : if $c_i = 0$ goto L' else $c_i := c_i - 1$; goto L''
- An initial label L_0
- A special label L_F with no output instruction

Halting problem: Is the label L_F eventually reached?

Theorem [Minsky'67]

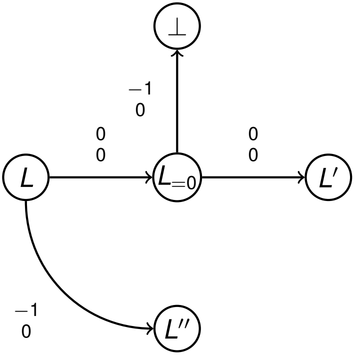
The halting problem for Minsky machines is undecidable.

Remark:

- VASS can simulate easily increment and decrement
- For zero-test, it is not directly possible

Using branching time logics for zero testing

- VASS to encode L : if $c_1 = 0$ goto L' else $c_1 := c_1 - 1$; goto L''



If the branching logic can:

- Test the existence of a run reaching L_F such that at any moment, if the VASS is in state $L_{=0}$ the state $\perp$ is not reachable.

$\Rightarrow$ **the model-checking becomes undecidable**

Outline

1. VASS and their Toolbox
2. **Playing in VASS**
3. Qualitative Analysis of Probabilistic VASS
4. Probabilities and Non-Determinism in VASS
5. Conclusion

Games and Logics

In finite state systems

- There is a connection between **games** and **model-checking problems**
- Model-checking μ-calculus can be reduced to solving a parity game over a system

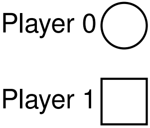
Games and Logics

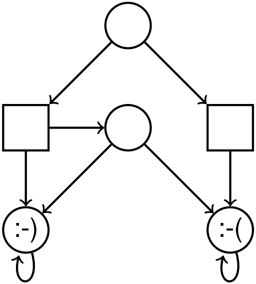
In finite state systems

- There is a connection between **games** and **model-checking problems**
- Model-checking μ-calculus can be reduced to solving a parity game over a system

What about games played on the transition system of a VASS ?

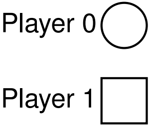
Finite-State Parity Games

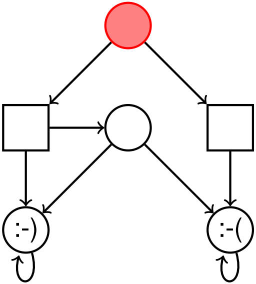




- Finite number of control states
- Every state belongs to Player 0 or Player 1
- Colors (in $\{1, \ldots, k\}$) associated to each state
- **Parity winning condition:** Player 0 wins iff the highest color seen infinitely often is even

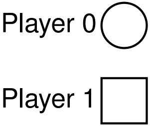
Finite-State Parity Games

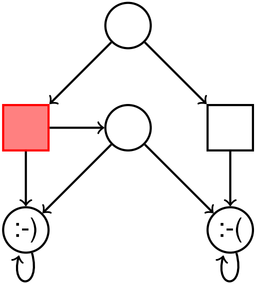




- Finite number of control states
- Every state belongs to Player 0 or Player 1
- Colors (in $\{1, \ldots, k\}$) associated to each state
- **Parity winning condition:** Player 0 wins iff the highest color seen infinitely often is even

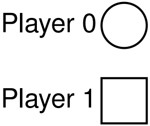
Finite-State Parity Games

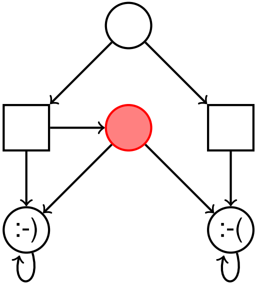




- Finite number of control states
- Every state belongs to Player 0 or Player 1
- Colors (in $\{1, \ldots, k\}$) associated to each state
- **Parity winning condition:** Player 0 wins iff the highest color seen infinitely often is even

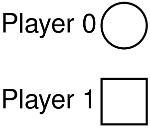
Finite-State Parity Games

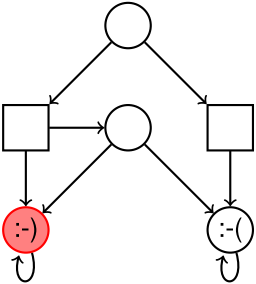




- Finite number of control states
- Every state belongs to Player 0 or Player 1
- Colors (in $\{1, \ldots, k\}$) associated to each state
- **Parity winning condition:** Player 0 wins iff the highest color seen infinitely often is even

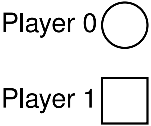
Finite-State Parity Games

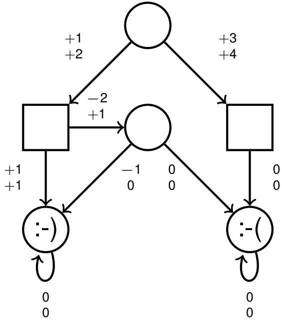




- Finite number of control states
- Every state belongs to Player 0 or Player 1
- Colors (in $\{1, \ldots, k\}$) associated to each state
- **Parity winning condition:** Player 0 wins iff the highest color seen infinitely often is even

Integer Vector Games

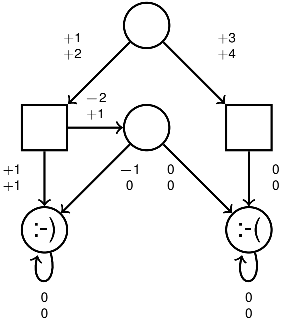




- Adding counters $C_1, \ldots, C_n$ to the game
- Transitions can decrement and increment the counter values
- Configurations are pairs $(q, \mathbf{v})$ with:
    - q : control state
    - $\mathbf{v} \in \mathbb{Z}^n$: values for the counters

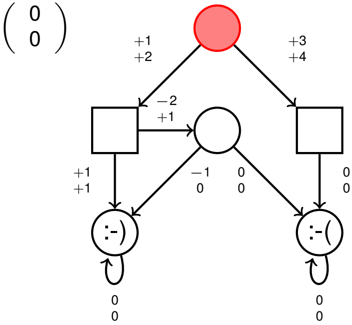
Which role play the counters in the winning condition and in the enabledness of transitions ?

Energy Semantics



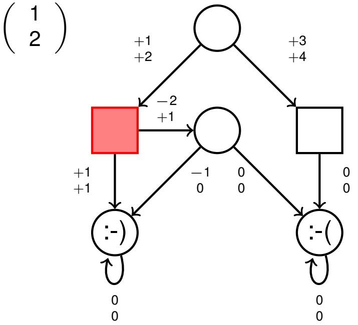
- Transitions are always enabled regardless of the counter values
- If the value of one counter drops below 0, Player 1 wins
- Player 0 wins if no counter drops below 0 and if the parity condition is respected
- Higher value of the counters are always better for Player 0

Energy Semantics



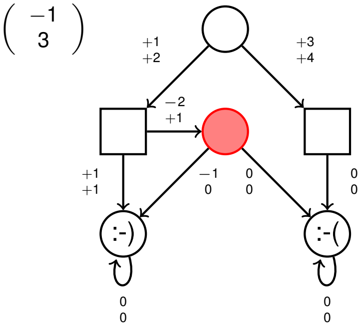
- Transitions are always enabled regardless of the counter values
- If the value of one counter drops below 0, Player 1 wins
- Player 0 wins if no counter drops below 0 and if the parity condition is respected
- Higher value of the counters are always better for Player 0

Energy Semantics



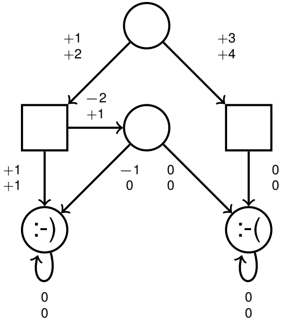
- Transitions are always enabled regardless of the counter values
- If the value of one counter drops below 0, Player 1 wins
- Player 0 wins if no counter drops below 0 and if the parity condition is respected
- Higher value of the counters are always better for Player 0

Energy Semantics



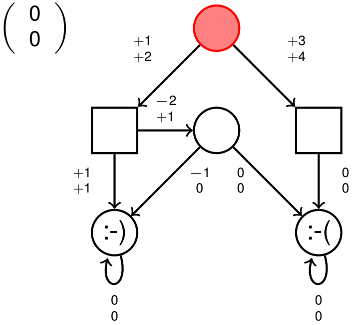
- Transitions are always enabled regardless of the counter values
- If the value of one counter drops below 0, Player 1 wins
- Player 0 wins if no counter drops below 0 and if the parity condition is respected
- Higher value of the counters are always better for Player 0

VASS Semantics



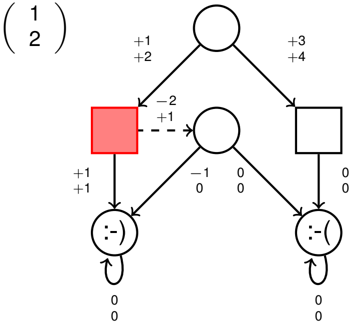
- Configurations : $(q, \mathbf{v})$ with $\mathbf{v} \in \mathbb{N}^n$
- Transitions that make a counter drop below 0 are disabled
- Player 0 wins if the parity condition is respected
- **Higher value of the counters are NOT always better for Player 0**

VASS Semantics



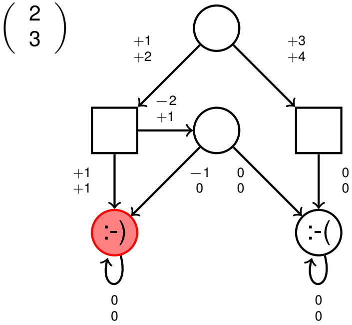
- Configurations : $(q, \mathbf{v})$ with $\mathbf{v} \in \mathbb{N}^n$
- Transitions that make a counter drop below 0 are disabled
- Player 0 wins if the parity condition is respected
- **Higher value of the counters are NOT always better for Player 0**

VASS Semantics



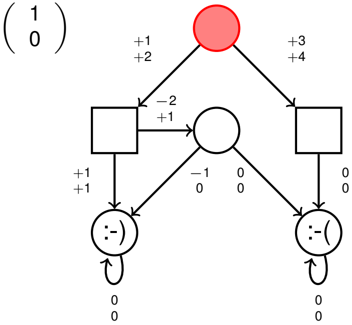
- Configurations : $(q, \mathbf{v})$ with $\mathbf{v} \in \mathbb{N}^n$
- Transitions that make a counter drop below 0 are disabled
- Player 0 wins if the parity condition is respected
- **Higher value of the counters are NOT always better for Player 0**

VASS Semantics



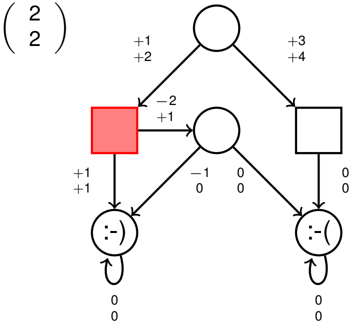
- Configurations : $(q, \mathbf{v})$ with $\mathbf{v} \in \mathbb{N}^n$
- Transitions that make a counter drop below 0 are disabled
- Player 0 wins if the parity condition is respected
- **Higher value of the counters are NOT always better for Player 0**

VASS Semantics



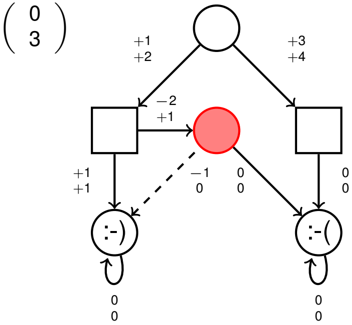
- Configurations : $(q, \mathbf{v})$ with $\mathbf{v} \in \mathbb{N}^n$
- Transitions that make a counter drop below 0 are disabled
- Player 0 wins if the parity condition is respected
- **Higher value of the counters are NOT always better for Player 0**

VASS Semantics



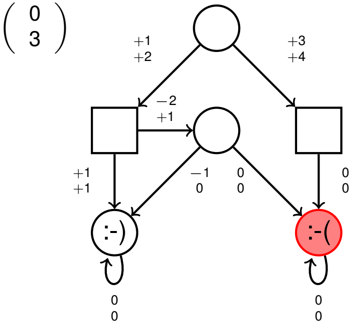
- Configurations : $(q, \mathbf{v})$ with $\mathbf{v} \in \mathbb{N}^n$
- Transitions that make a counter drop below 0 are disabled
- Player 0 wins if the parity condition is respected
- **Higher value of the counters are NOT always better for Player 0**

VASS Semantics



- Configurations : $(q, \mathbf{v})$ with $\mathbf{v} \in \mathbb{N}^n$
- Transitions that make a counter drop below 0 are disabled
- Player 0 wins if the parity condition is respected
- **Higher value of the counters are NOT always better for Player 0**

VASS Semantics



- Configurations : $(q, \mathbf{v})$ with $\mathbf{v} \in \mathbb{N}^n$
- Transitions that make a counter drop below 0 are disabled
- Player 0 wins if the parity condition is respected
- **Higher value of the counters are NOT always better for Player 0**

Problems

For $\mathtt{I} \in \{\mathtt{Energy}, \mathtt{VASS}\}$ and a game $\mathcal{G}$:

- $\mathtt{Win}(\mathcal{G}, \mathtt{I}) = \{(q, \mathbf{v}) \in Q \times \mathbb{N}^n \mid$ Player 0 has a winning strategy from $(q, \mathbf{v})\}$

Unknown initial credit problem

- **Input:** A game $\mathcal{G}$ and a semantic $\mathtt{I} \in \{\mathtt{Energy}, \mathtt{VASS}\}$
- **Output:** Is $\mathtt{Win}(\mathcal{G}, \mathtt{I})$ not empty ?

Fixed initial credit problem

- **Input:** A game $\mathcal{G}$,a semantic $\mathtt{I} \in \{\mathtt{Energy}, \mathtt{VASS}\}$ and a configuration $(q, \mathbf{v})$
- **Output:** Do we have $(q, \mathbf{v}) \in \mathtt{Win}(\mathcal{G}, \mathtt{I})$?

Computing the winning set

- **Input:** A game $\mathcal{G}$ and a semantic $\mathtt{I} \in \{\mathtt{Energy}, \mathtt{VASS}\}$
- **Output:** Can we compute (and represent finitely) $\mathtt{Win}(\mathcal{G}, \mathtt{I})$?

Previous Results

Theorem [Chatterjee et al.'12]

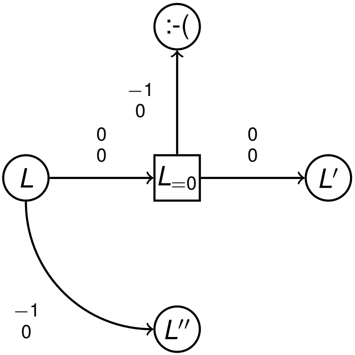
The unknown initial credit problem is $\texttt{coNP}$-complete for energy games.

Theorem [Abdulla et al.'03]

The fixed initial credit problem is undecidable for VASS games (even with reachability objectives).

Why Reachability VASS Games are Undecidable ?

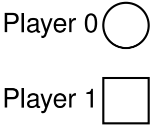
- VASS to encode L : if $c_1 = 0$ goto L' else $c_1 := c_1 - 1$; goto L''

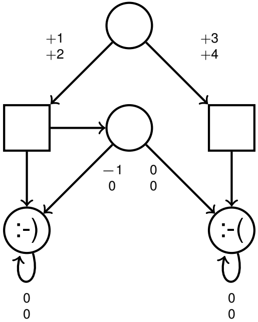


- Player 0 chooses whether the counter is equal to 0
- If she cheats, Player 1 punishes it!!!

Single Sided Games

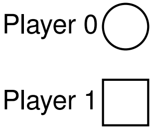
Player 1 cannot change the counter values

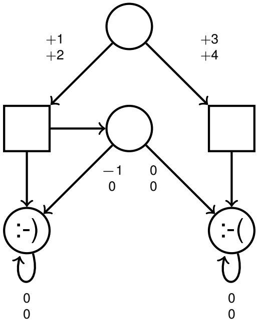




Single Sided Games

Player 1 cannot change the counter values





$$CPre(C) =$$
$$\{(q, \mathbf{v}) \mid q \text{ is } \bigcirc \text{ and } \exists (q', \mathbf{v}') \in C.(q, \mathbf{v}) \rightarrow (q', \mathbf{v}')\} \cup$$
$$\{(q, \mathbf{v}) \mid q \text{ is } \Box \text{ and } (q, \mathbf{v}) \rightarrow (q', \mathbf{v}') \text{ implies } (q, \mathbf{v}') \in C\}$$

If C is upward-closed then $CPre(C)$ is upward-closed

Results for Single Sided Games

Theorem [Raskin et al.'04]

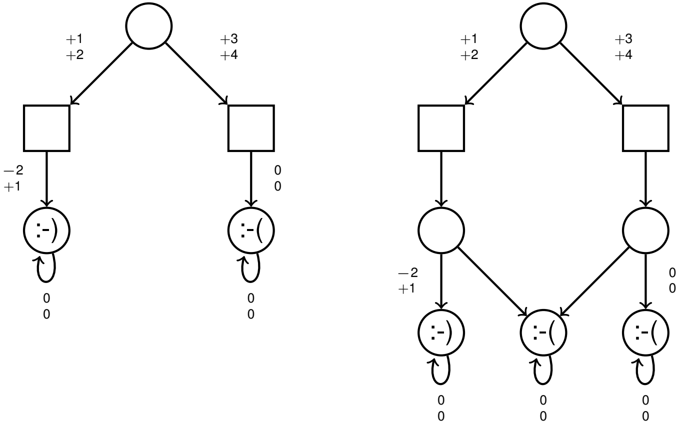
The fixed initial credit problem is decidable for single-sided VASS games with reachability objectives.

Proposition [Abdulla et al.'13]

For energy games and single-sided VASS games, the winning regions are upward closed.

- What Player 0 can achieve with some values, she can achieve it as well with bigger values !

From energy games to single-sided games



Proposition [Abdulla et al.'13]

Energy games and single-sided VASS games are PTIME inter-reducible.

Results

Theorem [Abdulla et al.'13]

For single-sided VASS games, the minimal elements of the winning sets are computable.

- The proof is done by induction on the dimension of the VASS games
- It refined an over-approximation of the winning set at each step
- At each step, it builds a finite state games where the states are labelled with some counter values and ω (standing for any values is accepted)
- It is inspired by the Karp and Miller coverability graph

Corollary [Abdulla et al.'13]

For energy games, the minimal elements of the winning sets are computable.

Hence, for single-sided VASS games and energy games we can solve:

- The unknown initial credit problem
- The fixed initial credit problem

Back to logic

μ-calculus syntax

$$\phi ::= q \mid X \mid \phi \wedge \phi \mid \phi \vee \phi \mid \Diamond\phi \mid \Box\phi \mid \mu X.\phi \mid \nu X.\phi$$

where q is a control state of the VASS.

- Each closed formula ϕ characterizes a set of configurations $[\![\phi]\!]$
- $\mu X.\Diamond X \vee q$ is the least fixpoint of the function $f : X \mapsto [\![\Diamond X]\!] \cup [\![q]\!]$ where:
  - $[\![\Diamond X]\!]$ is the set of configurations that have a sucessor in X
  - $[\![q]\!]$ is the set of configurations $\{(q, \mathbf{v}) \mid \mathbf{v} \in \mathbb{N}^n\}$
- νX stands for greatest fixpoint
- $\Box X$ stands for all successors are in X
- $\mu X.\Diamond X \vee q_F$ represents the configurations that can reach q_F

Model-checking of μ-calculus

- **Input:** A VASS and a closed μ-calculus formula ϕ
- **Output:** Does $(q_0, \mathbf{0}) \in [\![\phi]\!]$?

Results

Theorem

The model-checking μ-calculus on VASS is undecidable.

- Single-Sided VASS $S = \langle Q, Q_0, Q_1, E, q_0 \rangle$:
  - $\langle Q, E, q_0 \rangle$ is VASS
  - $Q_0 \uplus Q_1$ is a partition of Q
  - for all $(q, \mathbf{v}, q') \in E$ if $q \in Q_1$ then $\mathbf{v} = \mathbf{0}$
- Guarded fragment of μ-calculus : replace $\Box X$ with $Q_1 \wedge \Box X$

Theorem [Abdulla et al.'13]

Model-checking the guarded fragment of μ-calculus on single-sided VASS is decidable.

- The model-checking problem can be translated to a Single-Sided VASS parity game
- The translation is similar to the one in finite state systems
- The guard is necessary since states of Player 1 correspond to the operator $\Box X$

Outline

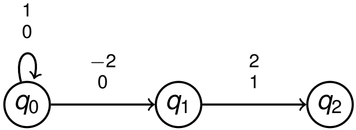
Probabilistic VASS (PVASS)

PVASS

A n-dim PVASS $S = \langle Q, E, q_0, w \rangle$ is a VASS $\langle Q, E, q_0 \rangle$ equipped with a weight function $w : E \mapsto \mathbb{N} \setminus \{0\}$.

- The semantics of a PVASS is a **Markov Chain**
- The probability to go from $(q, \mathbf{v})$ to $(q', \mathbf{v}')$ is equal to

$$\frac{w(q, \mathbf{v}' - \mathbf{v}, q')}{\Sigma_{\{e=(q,\mathbf{v}'',q'')\in E\}} w(e)}$$



- If the weight of each transition is 1
- Then $(q_0, (0,0))$ goes to $(q_0, (1,0))$ with probability 1
- And $(q_0, (2,0))$ goes to $(q_0, (3,0))$ with probability $\frac{1}{2}$ and to $(q_1, (0,0))$ with probability $\frac{1}{2}$

Classical problems in PVASS

- For an event $\mathcal{E}$, characterizing a set of runs, we denote by $\mathbb{P}(\mathcal{E})$ the probability of $\mathcal{E}$ in the Markov Chain of the VASS
- Let $\Diamond q$ be the set of runs reaching q
- Let $\Box\Diamond q$ be the set of runs visiting q infinitely often

Almost Sure Control State Reachability

- **Input:** A PVASS S and a control state q_F
- **Output:** Do we have $\mathbb{P}(\Diamond q_F) = 1$?

Almost Sure Control State Repeated Reachability

- **Input:** A PVASS S and a control state q_F
- **Output:** Do we have $\mathbb{P}(\Box\Diamond q_F) = 1$?

A nice property of PVASS

- Let $\widetilde{q_F}$ represent the configurations that cannot reach q_F
- Let $\Diamond\widetilde{q_F}$ be the runs that reach such a configurations

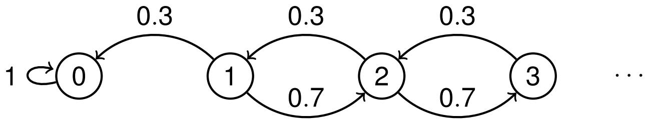
Lemma (Decisive Markov Chains) [Abdulla et al.'07]

In PVASS, we have $\mathbb{P}(\Diamond q_F \vee \Diamond\widetilde{q_F}) = 1$

- In other words, a run that always visits states from which q_F can be reached without visiting q_F has probability 0
- All the states s' in a run in $\neg(\Diamond q_F \vee \Diamond\widetilde{q_F})$ are in $\neg\widetilde{q_F}$
- They never visit q_F nor $\widetilde{q_F}$
- Hence from all this states q_F is reachable in a bounded number of steps (the computing of predecessors is bounded by k)
- Hence from all this states the probability to reach q_F is greater than m^k (where m is the minimum possible probability)
- This allows to deduce that $\mathbb{P}(\neg(\Diamond q_F \vee \Diamond\widetilde{q_F})) \leq (1 - m^k)^\infty = 0$

Which Markov Chains are Decisive ?

- When is the property $\mathbb{P}(\Diamond q_F \vee \Diamond \widetilde{q_F}) = 1$ satisfied ?
- It is true in finite states Markov chains
- It holds for some infinite states Markov chains, for instance Probabilistic Lossy Channel Systems
- It does not hold for all infinite states Markov chains, for instance the gambler ruin :



Probability to reach 0 from 1 strictly smaller than 1.

Results

Theorem [Abdulla et al.'07]

Almost Sure Control State Reachability is decidable in PVASS

- $\mathbb{P}(\Diamond q_F) = 1$ iff from $(q_0, \mathbf{0})$ it is not possible to reach $\widetilde{q_F}$ without seing q_F
- $\widetilde{q_F}$ is the complement of an upward closed set of configurations (the complement of the predecessors of q_F)
- To test this, remove the outgoing edge of q_F and test whether a state in $\widetilde{q_F}$ can be reached
- This reduces to reachability in VASS

Results

Theorem [Abdulla et al.'07]

Almost Sure Control State Reachability is decidable in PVASS

- $\mathbb{P}(\Diamond q_F) = 1$ iff from $(q_0, \mathbf{0})$ it is not possible to reach $\widetilde{q_F}$ without seing q_F
- $\widetilde{q_F}$ is the complement of an upward closed set of configurations (the complement of the predecessors of q_F)
- To test this, remove the outgoing edge of q_F and test whether a state in $\widetilde{q_F}$ can be reached
- This reduces to reachability in VASS

Theorem [Abdulla et al.'07]

Almost Sure Repeated Control State Reachability is decidable in PVASS

Outline

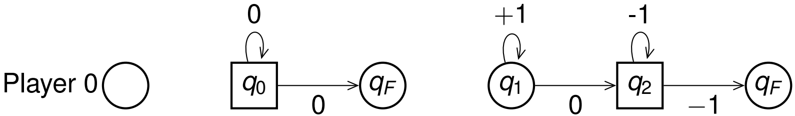
VASS Markov Decision Processes (VASS-MDP)

VASS-MDP

A n-dim VASS-MDP $S = \langle Q, Q_0, Q_P, E, q_0, w \rangle$ is a VASS $\langle Q, E, q_0 \rangle$ equipped with :

- a weight function $w : E \mapsto \mathbb{N} \setminus \{0\}$
- $Q_0 \uplus Q_P$ forms a partition of Q between non-deterministic states Q_0 and probabilistic states Q_P.

- The weight function matters only for transitions leaving probabilistic states.
- The semantics is given in term of a Markov Decision Process
- A scheduler $\sigma \in \Sigma$ resolves non-determinism
  - It assigns to each finite run ending in a state of Q_0 a successor configuration.



Classical Problems in VASS-MDP

- Once a strategy σ is given we get a Markov Chain
- We will denote $\mathbb{P}^\sigma(\mathcal{E})$ the probability of event $\mathcal{E}$ in the Markov chain obtained when considering σ

Almost Sure Control State Reachability

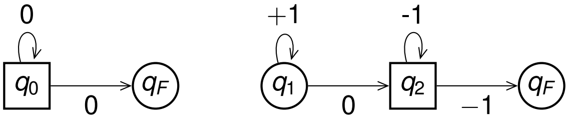
- **Input:** A VASS-MDP S and a control state q_F
- **Output:** Does there exist σ such that $\mathbb{P}^\sigma(\Diamond q_F) = 1$?

Almost Sure Control State Repeated Reachability

- **Input:** A VASS-MDP S and a control state q_F
- **Output:** Does there exist σ such that $\mathbb{P}^\sigma(\Box\Diamond q_F) = 1$?

- There are as well the Limit Sure versions $sup_{\sigma\in\Sigma}\mathbb{P}^\sigma(\Diamond q_F) = 1$ and $sup_{\sigma\in\Sigma}\mathbb{P}^\sigma(\Box\Diamond q_F) = 1$?
- Limit Sure and Almost Sure problems are equivalent in finite state systems **but not in our case!**

Examples



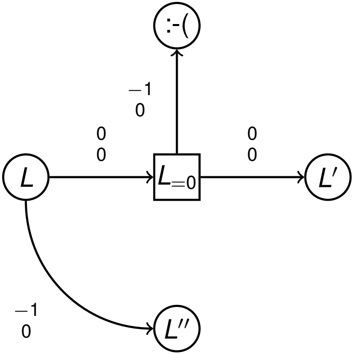
- From $(q_0, 0)$, the state q_F is reached almost surely
- From $(q_1, 0)$, the state q_F is reached limit surely but not almost surely

Always the Same Trick Leads to Undecidability

Theorem

Almost Sure (Repeated) Control State Reachability is undecidable for VASS-MDP.

- VASS-MPD to encode L : if $c_1 = 0$ goto L' else $c_1 := c_1 - 1$; goto L''



- Here each transition has weight 1

How to Regain Decidability ?

In finite states MDP:

- Almost-sure (repeated) reachability is decidable
- One can characterize the winning configurations by a μ-calculus formula [Chatterjee et al.'09]

How to Regain Decidability ?

In finite states MDP:

- Almost-sure (repeated) reachability is decidable
- One can characterize the winning configurations by a μ-calculus formula [Chatterjee et al.'09]

For VASS:

- We just see some restrictions leading to decidability of a μ-calculus fragment

Does this allow us to obtain some results ?

Single Sided VASS-MDP

- In **Single Sided VASS-MDPs**, Player P cannot change the counter values
- Hence the underlying VASS is as well single sided

What needs to be done to get the results ?

1. Prove that there exists a μ-calculus formula characterizing the winning configurations
2. Prove that this formula belongs to the guarded fragment!

One Last Effort

- Take the formula
  $\texttt{InvPre}(X, Y) = (Q_0 \wedge \Diamond(X \wedge Y)) \vee (\Diamond Y \wedge Q_P \wedge \Box X)$
- It represents the set of states from which Player 0 can go to X and Y and Player P can go to Y and cannot go out of X
- Now the formula $\nu X.\mu Y.\left(q_F \vee \texttt{InvPre}(X, Y)\right)$
- It represents the configurations that can reach almost surely q_F.
- **Warning:** This is true only because we have VASS, in other infinite state systems, this does not hold!
- In fact, in this set Player 0 can reach q_F in N steps and Player 0 cannot take him out of this set.

Theorem [Abdulla et al.'16]

Almost Sure (Repeated) Control State Reachability is decidable in Single Sided VASS-MDP.

- We also show that Limit Sure Reachability is decidable in Single Sided VASS-MDP, but with a different proof.

Outline

Now and Then

To sum up

- Restricting the 'universal' or 'probabilistic' player in VASS leads to decidability
- Some techniques of the finite state world can be adapted to VASS due to the 'short path to reach a state' property
- I have a way to solve my original problem automatically :-)

What's next

- What about quantitative verification ... it's an hard problem
- Application to Parameterized Verification
- Other games might be decidable with a limited choice for Player P ... (and reachability objectives not restricted to control states).