Algorithms for Model Checking (2IW55) Lecture 7: Solving parity games using small progress measures

Background material:
M. Jurdziński "Small Progress Measures for Solving Parity Games"

Maciej Gazda

using Jeroen Keiren's slides

Technische Universiteit Eindhoven University of Technology
Where innovation starts

/ Department of Mathematics and Computer Science

Recursive algorithm (recap)

Goal: compute winning sets

Relevant concepts:

- Divide and conquer
- Base: empty game
- Step:
  - Compute dominion
  - Compute attractor set
  - Solve remaining subgame
  - Assemble winning sets/strategies from
    - winning sets/strategies of subgames
    - attractor strategy for one of players reaching set of nodes with minimal priority in the game

Algorithms

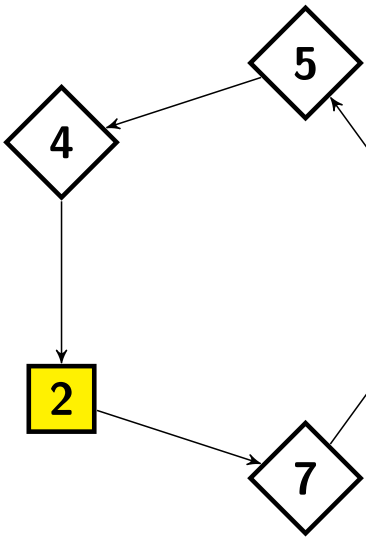
- Recursive
- Small progress measures (iterative)

Small progress measures (intuition)

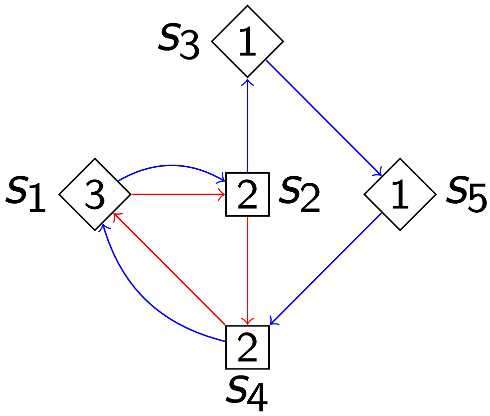
- **Characterise cycles reachable from each vertex. Cycles can be used to decide the winner.**
- Assign a certain measure to each vertex that decreases along the play with each "bad" priority encountered, and can only increase if a "good" value is reached.
- Measure computed using fixed point iteration

Parity games are about odd/even cycles

even [odd] cycle = a cycle in which the lowest priority is even [odd]



Parity games are about odd/even cycles



Player Even [Odd] wins a vertex iff they can force that all cycles appearing in the play are even [odd].

Solitaire game

In a solitaire game, only one player makes (nontrivial) choices.

Definition (Solitaire game)

Parity game $G = (V, E, p, (V_\Diamond, V_\Box))$ is a $\bigcirc$-solitaire game if

$$\forall v \in V_{\overline{\bigcirc}} : v \to w \wedge v \to w' \implies w = w'$$

Given a strategy $\psi_\bigcirc$, parity game $G = (V, E, p, (V_\Diamond, V_\Box))$ can be turned into solitaire game $G_{\psi_\bigcirc} = (V, E', p, (V_\Diamond, V_\Box))$, where

$$E' = \{(v, w) \in E \mid v \in V_\bigcirc \wedge w = \psi_\bigcirc(v)\} \\ \cup \{(v, w) \in E \mid v \in V_{\overline{\bigcirc}}\}$$

Cycles vs winning strategies

Let $G = (V, E, p, (V_\Diamond, V_\Box))$ be a parity game, with:

- $W \subseteq V$
- strategy $\psi_\Diamond$ closed on W.

Consider solitaire game $G_{\psi_\Diamond} \cap W$.

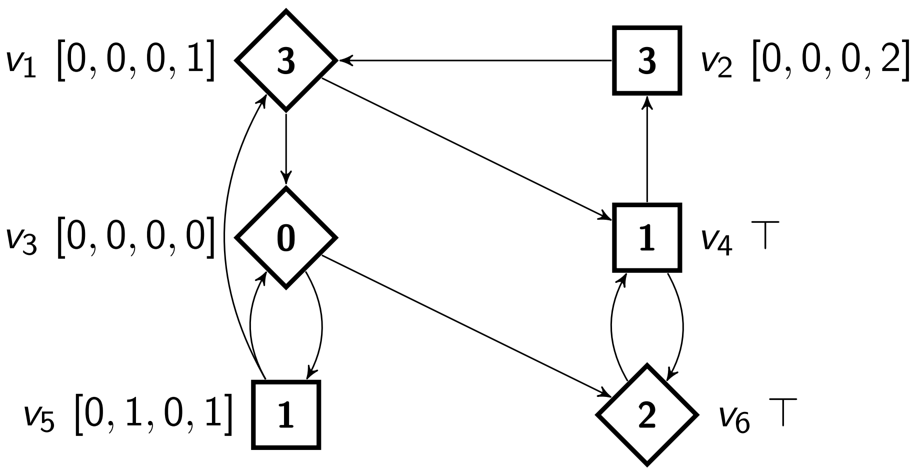
Property

$\psi_\Diamond$ is winning for player $\Diamond$ from all $v \in W$ if and only if all cycles in $G_{\psi_\Diamond} \cap W$ are even

Small progress measures (intuition)

- Characterise cycles reachable from each vertex. Cycles can be used to decide the winner.
- **Assign a certain measure to each vertex that decreases along the play with each "bad" priority encountered, and can only increase if a "good" value is reached.**
- Measure computed using fixed point iteration

A progress measure



d-tuples

Let $\alpha \in \mathbb{N}^d$ be a d-tuple of natural numbers

- we number its components from 0 to $d-1$, i.e. $\alpha = (\alpha_0, \alpha_1, \ldots, \alpha_{d-1})$,
- $<, \leqslant, =, \neq, \geqslant, >$ on tuples denote lexicographic ordering,
- $(n_0, n_1, \ldots, n_k) \equiv_i (m_0, m_1, \ldots, m_l)$ iff $(n_0, n_1, \ldots, n_i) \equiv (m_0, m_1, \ldots, m_i)$, for $\equiv \in \{<, \leqslant, =, \neq, \geqslant, >\}$
- Note that if $i > k$ or $i > l$, the tuples will be suffixed with 0s

d-tuples

Let $\alpha \in \mathbb{N}^d$ be a d-tuple of natural numbers

- we number its components from 0 to $d-1$, i.e. $\alpha = (\alpha_0, \alpha_1, \ldots, \alpha_{d-1})$,
- $<, \leqslant, =, \neq, \geqslant, >$ on tuples denote lexicographic ordering,
- $(n_0, n_1, \ldots, n_k) \equiv_i (m_0, m_1, \ldots, m_l)$ iff $(n_0, n_1, \ldots, n_i) \equiv (m_0, m_1, \ldots, m_i)$, for $\equiv \in \{<, \leqslant, =, \neq, \geqslant, >\}$
- Note that if $i > k$ or $i > l$, the tuples will be suffixed with 0s

Intuition: when encountering priority i, we are interested only in information concerning i or lower (more significant) priorities. A given priority i "cancels" the impact of all less significant priorities.

d-tuples (example)

- $(0,1,0,1) =_0 (0,2,0,1) \equiv (0) = (0) \equiv$ true
- $(0,1,0,1) <_1 (0,2,0,1) \equiv (0,1) < (0,2) \equiv$ true
- $(0,1,0,1) \geqslant_3 (0,2,0,1) \equiv (0,1,0,1) \geqslant (0,2,0,1) \equiv$ false

Restricted *d*-tuples

Let $G = (V, E, p, (V_\Diamond, V_\Box))$ be a parity game, and let $d = \max\{p(v) \mid v \in V\} + 1$.

- For $i \in \mathbb{N}$, let $V_i = \{v \in V \mid p(v) = i\}$,
- Denote $n_i = \mid V_i \mid$, the number of vertices with priority i,

Define $\mathbb{M}^\Diamond \subseteq \mathbb{N}^d$, such that it is the finite set of *d*-tuples, with:

- 0 on even positions
- Natural numbers $\leqslant n_i$ on odd positions i

$\mathbb{M}^\Box$ is defined similarly (swap even and odd in the definition)

Restricted d-tuples
!

Let $G = (V, E, p, (V_\diamond, V_\square))$ be a parity game, and let $d = \max\{p(v) \mid v \in V\} + 1$.

- For $i \in \mathbb{N}$, let $V_i = \{v \in V \mid p(v) = i\}$,
- Denote $n_i = \mid V_i \mid$, the number of vertices with priority i,

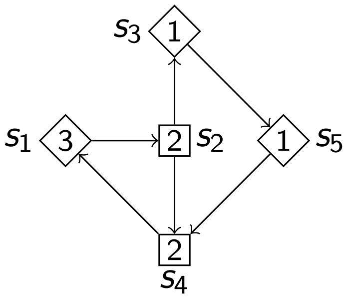
Define $\mathbb{M}^\diamond \subseteq \mathbb{N}^d$, such that it is the finite set of d-tuples, with:

- 0 on even positions
- Natural numbers $\leqslant n_i$ on odd positions i

$\mathbb{M}^\square$ is defined similarly (swap even and odd in the definition)

$\mathbb{M}^\diamond$ (example)

Determine maximum value of $\mathbb{M}^\diamond$ for the following parity game:



- Maximum value of $\mathbb{M}^\diamond$ is $(0, 2, 0, 1)$
- $\mathbb{M}^\diamond = \{0\} \times \{0, 1, 2\} \times \{0\} \times \{0, 1\}$

Parity progress measure

On solitaire games

Recall: $\psi_\diamond$ is winning for player $\diamond$ from W if and only if all cycles in $G_{\psi_\diamond} \cap W$ are even

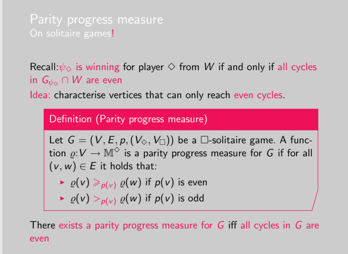
Idea: characterise vertices that can only reach even cycles.

Definition (Parity progress measure)

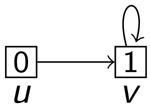
Let $G = (V, E, p, (V_\diamond, V_\square))$ be a $\square$-solitaire game. A function $\varrho: V \to \mathbb{M}^\diamond$ is a parity progress measure for G if for all $(v, w) \in E$ it holds that:

- $\varrho(v) \geqslant_{p(v)} \varrho(w)$ if $p(v)$ is even
- $\varrho(v) >_{p(v)} \varrho(w)$ if $p(v)$ is odd

There exists a parity progress measure for G iff all cycles in G are even



Parity progress measure (problem)



Problem: no parity progress measure can be assigned to these vertices, as parity progress measure only exists for even cycles. (Second clause requires $\varrho(v) >_1 \varrho(v)$)

Extended parity progress measures

Allowing odd cycles

Define $\mathbb{M}^{\bigcirc,\top} = \mathbb{M}^{\bigcirc} \cup \{\top\}$, such that:

- $m\{<, <_i\}\top$ for all $m \in \mathbb{M}^{\bigcirc}$, and $m\{\neq, \neq_i\}\top$
- $\top =_i \top$ for all i.

Extend ϱ such that $\top$ is used for infinite values.

Let $G = (V, E, p, (V_\diamond, V_\Box))$ be a solitaire game. The winning sets are determined as:

- $W_\diamond = \{v \in V \mid \varrho(v) \neq \top\}$
- $W_\Box = V \setminus W_\diamond$.

Definition (Prog)

If $\varrho{:}V \to \mathbb{M}^{\bigcirc,\top}$ and $(v,w) \in E$, then $Prog(\varrho, v, w)$ is the least $m \in \mathbb{M}^{\bigcirc,\top}$, such that

- $m \geqslant_{p(v)} \varrho(w)$ if $p(v)$ is even,
- $m >_{p(v)} \varrho(w)$, or $m = \varrho(w) = \top$ if $p(v)$ is odd.

Game parity progress measures
Cope with $\top$ element

Definition (Prog)

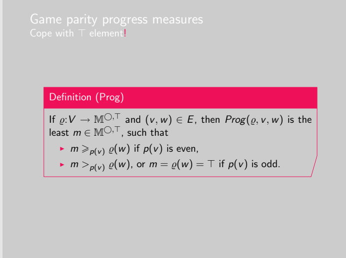
If $\varrho{:}V \to \mathbb{M}^{\bigcirc,\top}$ and $(v,w) \in E$, then $Prog(\varrho, v, w)$ is the least $m \in \mathbb{M}^{\bigcirc,\top}$, such that

- $m \geqslant_{p(v)} \varrho(w)$ if $p(v)$ is even,
- $m >_{p(v)} \varrho(w)$, or $m = \varrho(w) = \top$ if $p(v)$ is odd.

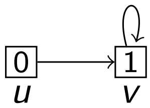
Prog (examples)

Let $\mathbb{M}^{\diamond} = \{0\} \times \{0,1,2\} \times \{0\} \times \{0,1\}$

- Suppose $p(v) = 0$, $\varrho(w) = (0,2,0,0)$.
  Then $Prog(\varrho, v, w) = (0,0,0,0)$
- Suppose $p(v) = 1$, $\varrho(w) = (0,2,0,0)$.
  Then $Prog(\varrho, v, w) = \top$
- Suppose $p(v) = 3$, $\varrho(w) = (0,2,0,0)$.
  Then $Prog(\varrho, v, w) = (0,2,0,1)$



Game parity progress measure (example)



- Observe: $\varrho(u) = \varrho(v) = \top$
- Measure can identify both even and odd reachable cycles.

Game parity progress measure

From solitaire to parity games

For each vertex in which player $\Diamond$ moves, there is at least one neighbour making progress.

Definition (Game parity progress measure)

Let $G = (V, E, p, (V_\Diamond, V_\Box))$ be a parity game. A function $\varrho{:}V \to \mathbb{M}^{\bigcirc,\top}$ is a game parity progress measure if for all $v \in V$, it holds that:

- if $v \in V_\Diamond$, then $\exists_{(v,w)\in E}\, \varrho(v) \geqslant_{p(v)} Prog(\varrho, v, w)$
- if $v \in V_\Box$, then $\forall_{(v,w)\in E}\, \varrho(v) \geqslant_{p(v)} Prog(\varrho, v, w)$

Small progress measure

If ϱ is least game parity progress measure, then the following are equivalent:

- $\varrho(v) \neq \top$
- there is a strategy of player $\Diamond$ such that in the induced $\Box$-solitaire game all cycles reachable from vertex v are even
- $v \in W_{\Diamond}$

Small progress measures (intuition)

- Characterise cycles reachable from each vertex. Cycles can be used to decide the winner.
- Assign a certain measure to each vertex that decreases along the play with each "bad" priority encountered, and can only increase if a "good" value is reached.
- **Measure computed using fixed point iteration.**

Fixed points

Characterise game parity progress measure as fixed point of monotone operators in a finite complete lattice:

- a least game parity progress measure φ exists (Knaster-Tarski),
- computable by fixed point iteration (similar to Lecture 2, slide 8),

Let $G = (V, E, p, (V_\Diamond, V_\Box))$, and $\varphi, \varrho: V \to \mathbb{M}^{\bigcirc,\top}$.

- $\varphi \sqsubseteq \varrho$ if $\varphi(v) \leqslant \varrho(v)$ for all $v \in V$
- write $\varphi \sqsubset \varrho$ if $\varphi \sqsubseteq \varrho$ and $\varphi \neq \varrho$.

$\sqsubseteq$ gives a complete lattice structure on the set of functions $V \to \mathbb{M}^{\bigcirc,\top}$.

Lifting progress measures

Define $Lift_v(\varrho)$ for $v \in V$ as follows:

$$Lift_v(\varrho) = \begin{cases} \varrho[v := \min\{Prog(\varrho, v, w) \mid (v, w) \in E\}] & \text{if } v \in V_\Diamond \\ \varrho[v := \max\{Prog(\varrho, v, w) \mid (v, w) \in E\}] & \text{if } v \in V_\Box \end{cases}$$

Observe:

- For every $v \in V$, $Lift_v$ is $\sqsubseteq$-monotone.
- A function $\varrho: V \to \mathbb{M}^{\bigcirc,\top}$ is a game parity progress measure if and only if $Lift_v(\varrho) \sqsubseteq \varrho$ for all $v \in V$.

The algorithm

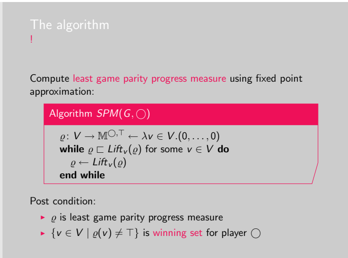
Compute least game parity progress measure using fixed point approximation:

Algorithm $SPM(G, \bigcirc)$

$\varrho \colon V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\ldots,0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
  $\varrho \leftarrow Lift_v(\varrho)$
end while

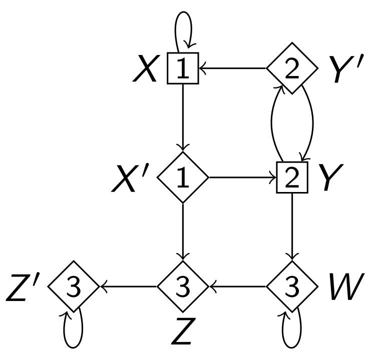
Post condition:

- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$


The algorithm
Compute least game parity progress measure using fixed point approximation:
Algorithm $SPM(G, \bigcirc)$
$\varrho \colon V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\ldots,0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ do
$\varrho \leftarrow Lift_v(\varrho)$
end while
Post condition:
ϱ is least game parity progress measure
$\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example)

Consider parity game G:



Maximum value of $\mathbb{M}^{\Diamond}$ is $(0, 2, 0, 3)$

The algorithm

!

Compute least game parity progress measure using fixed point approximation:

Algorithm $SPM(G, \bigcirc)$

$\varrho\colon V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\ldots,0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
 $\varrho \leftarrow Lift_v(\varrho)$
end while

Post condition:

- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (1)

Initially: $\varrho \leftarrow \lambda v \in V.(0,0,0,0)$, so

v	$\varrho(v)$
X	$(0,0,0,0)$
X'	$(0,0,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$(0,0,0,0)$
W	$(0,0,0,0)$

2013-10-03

Algorithms for Model Checking (2IW55) Lecture 7:Solving parity games using smallprogress measures
└─Small progress measures
 └─Example
  └─The algorithm

The algorithm

!

Compute least game parity progress measure using fixed point approximation:

Algorithm $SPM(G, \bigcirc)$

$\varrho\colon V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\ldots,0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
 $\varrho \leftarrow Lift_v(\varrho)$
end while

Post condition:

- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (2)

Step 2: $\varrho \leftarrow Lift_X(\varrho) = \varrho[X := \max\{Prog(\varrho, X, X'), Prog(\varrho, X, X)\}] = \varrho[X := \max\{(0,1,0,0),(0,1,0,0)\}] = \varrho[X := (0,1,0,0)]$

v	$\varrho(v)$
X	$(0,1,0,0)$
X'	$(0,0,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$(0,0,0,0)$
W	$(0,0,0,0)$

Small progress measures (example) (3)

Step 3: $\varrho \leftarrow Lift_X(\varrho) = \varrho[X := \max\{Prog(\varrho, X, X'), Prog(\varrho, X, X)\}] = \varrho[X := \max\{(0,1,0,0),(0,2,0,0)\}] = \varrho[X := (0,2,0,0)]$

v	$\varrho(v)$
X	$(0,2,0,0)$
X'	$(0,0,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$(0,0,0,0)$
W	$(0,0,0,0)$

2013-10-03
Algorithms for Model Checking (2IW55) Lecture 7:Solving parity games using smallprogress measures
└─Small progress measures
  └─Example
    └─The algorithm

The algorithm
!

Compute least game parity progress measure using fixed point approximation:

Algorithm $SPM(G, \bigcirc)$

$\varrho\colon V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\ldots,0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
$\varrho \leftarrow Lift_v(\varrho)$
end while

Post condition:
- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (4)

Step 4: $\varrho \leftarrow Lift_X(\varrho) = \varrho[X := \max\{Prog(\varrho, X, X'), Prog(\varrho, X, X)\}] = \varrho[X := \max\{(0,1,0,0),\top\}] = \varrho[X := \top]$

v	$\varrho(v)$
X	$\top$
X'	$(0,0,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$(0,0,0,0)$
W	$(0,0,0,0)$

Small progress measures (example) (5)

Step 5:$Lift_{Y'}(\varrho) = \varrho[Y' := \min\{Prog(\varrho, Y', X), Prog(\varrho, Y', Y)\}] = \varrho[Y' := \min\{\top, (0,0,0,0)\}] = \varrho[Y' := (0,0,0,0)]$
$Lift_{Y}(\varrho) = \varrho[Y := \max\{Prog(\varrho, Y, W), Prog(\varrho, Y, Y')\}] = \varrho[Y := \max\{(0,0,0,0), (0,0,0,0)\}] = \varrho[Y := (0,0,0,0)]$
$\varrho \leftarrow Lift_{X'}(\varrho) = \varrho[X' := \min\{Prog(\varrho, X', Y), Prog(\varrho, X', Z)\}] = \varrho[X' := \min\{(0,1,0,0), (0,1,0,0)\}] = \varrho[X' := (0,1,0,0)]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$(0,0,0,0)$
W	$(0,0,0,0)$

2013-10-03
Algorithms for Model Checking (2IW55) Lecture 7:Solving parity games using smallprogress measures
└─Small progress measures
└─Example
└─The algorithm

The algorithm
!
Compute least game parity progress measure using fixed point approximation:
Algorithm $SPM(G, \bigcirc)$
$\varrho : V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\ldots,0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
$\varrho \leftarrow Lift_v(\varrho)$
end while
Post condition:
- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (6)

Step 6: $\varrho \leftarrow Lift_{Z'}(\varrho) = \varrho[Z' := \min\{Prog(\varrho, Z', Z')\}] = \varrho[Z' := \min\{(0,0,0,1)\}] = \varrho[Z' := (0,0,0,1)]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$(0,0,0,1)$
W	$(0,0,0,0)$

The algorithm

!

Compute least game parity progress measure using fixed point approximation:

Algorithm *SPM*$(G, \bigcirc)$

$\varrho : V \to \mathbb{M}^{\bigcirc, \top} \leftarrow \lambda v \in V.(0, \ldots, 0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
 $\varrho \leftarrow Lift_v(\varrho)$
end while

Post condition:

- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (4)

Step 7: $\varrho \leftarrow Lift_{Z'}(\varrho) = \varrho[Z' := \min\{Prog(\varrho, Z', Z')\}] = \varrho[Z' := \min\{(0,0,0,2)\}] = \varrho[Z' := (0,0,0,2)]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$(0,0,0,2)$
W	$(0,0,0,0)$

2013-10-03
Algorithms for Model Checking (2IW55) Lecture 7:Solving parity games using smallprogress measures
└─Small progress measures
 └─Example
  └─The algorithm

The algorithm

!

Compute least game parity progress measure using fixed point approximation:

Algorithm *SPM*$(G, \bigcirc)$

$\varrho : V \to \mathbb{M}^{\bigcirc, \top} \leftarrow \lambda v \in V.(0, \ldots, 0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
 $\varrho \leftarrow Lift_v(\varrho)$
end while

Post condition:

- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (8)

Step 8: $\varrho \leftarrow Lift_{Z'}(\varrho) = \varrho[Z' := \min\{Prog(\varrho, Z', Z')\}] = \varrho[Z' := \min\{(0,0,0,3)\}] = \varrho[Z' := (0,0,0,3)]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$(0,0,0,3)$
W	$(0,0,0,0)$

The algorithm
!

Compute least game parity progress measure using fixed point approximation:

Algorithm $SPM(G, \bigcirc)$
$\varrho : V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\ldots,0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
$\varrho \leftarrow Lift_v(\varrho)$
end while

Post condition:
- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (9)

Step 9: $\varrho \leftarrow Lift(\varrho, Z') = \varrho[Z' := \min\{Prog(\varrho, Z', Z')\}] = \varrho[Z' := \min\{(0,1,0,0)\}] = \varrho[Z' := (0,1,0,0)]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$(0,1,0,0)$
W	$(0,0,0,0)$

2013-10-03
Algorithms for Model Checking (2IW55) Lecture 7:Solving parity games using smallprogress measures
└─Small progress measures
└─Example
└─The algorithm

The algorithm
!

Compute least game parity progress measure using fixed point approximation:

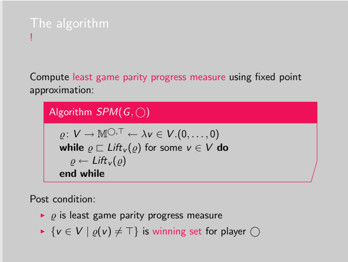
Algorithm $SPM(G, \bigcirc)$
$\varrho : V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\ldots,0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
$\varrho \leftarrow Lift_v(\varrho)$
end while

Post condition:
- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (10)

Step 10: $\varrho \leftarrow Lift_{Z'}(\varrho) = \varrho[Z' := \min\{Prog(\varrho, Z', Z')\}] = \varrho[Z' := \min\{(0,1,0,1)\}] = \varrho[Z' := (0,1,0,1)]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$(0,1,0,1)$
W	$(0,0,0,0)$



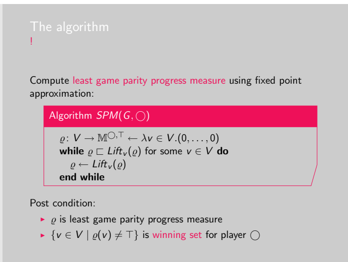
Small progress measures (example) (11)

Step 11*: Repeat lifting Z' even more often

$\varrho \leftarrow Lift_{Z'}(\varrho) = \varrho[Z' := \min\{Prog(\varrho, Z', Z')\}] = \varrho[Z' := \min\{\top\}] = \varrho[Z' := \top]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$(0,0,0,0)$
Z'	$\top$
W	$(0,0,0,0)$

2013-10-03
Algorithms for Model Checking (2IW55) Lecture 7:Solving parity games using smallprogress measures
Small progress measures
Example
The algorithm



Small progress measures (example) (12)

Step 12:

$\varrho \leftarrow Lift_{Z}(\varrho) = \varrho[Z := \min\{Prog(\varrho, Z, Z')\}] = \varrho[Z := \min\{\top\}] = \varrho[Z := \top]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$\top$
Z'	$\top$
W	$(0,0,0,0)$

The algorithm
!

Compute least game parity progress measure using fixed point approximation:

Algorithm $SPM(G, \bigcirc)$

$\varrho\colon V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\dots,0)$
while $\varrho \sqsubset \mathit{Lift}_v(\varrho)$ for some $v \in V$ **do**
$\varrho \leftarrow \mathit{Lift}_v(\varrho)$
end while

Post condition:
- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (13)

Step 13: $\varrho \leftarrow \mathit{Lift}_W(\varrho) = \varrho[W := \min\{\mathit{Prog}(\varrho, W, Z), \mathit{Prog}(\varrho, W, W')\}] = \varrho[W := \min\{\top, (0,0,0,1)\}] = \varrho[W := (0,0,0,1)]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$\top$
Z'	$\top$
W	$(0,0,0,1)$

2013-10-03
Algorithms for Model Checking (2IW55) Lecture 7:Solving parity games using smallprogress measures
└─Small progress measures
└─Example
└─The algorithm

The algorithm
!

Compute least game parity progress measure using fixed point approximation:

Algorithm $SPM(G, \bigcirc)$

$\varrho\colon V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\dots,0)$
while $\varrho \sqsubset \mathit{Lift}_v(\varrho)$ for some $v \in V$ **do**
$\varrho \leftarrow \mathit{Lift}_v(\varrho)$
end while

Post condition:
- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (14)

Step 14*: Repeat lifting of W often

$\varrho \leftarrow \mathit{Lift}_W(\varrho) = \varrho[W := \min\{\mathit{Prog}(\varrho, W, Z), \mathit{Prog}(\varrho, W, W')\}] = \varrho[W := \min\{\top, \top\}] = \varrho[W := \top]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$(0,0,0,0)$
Y'	$(0,0,0,0)$
Z	$\top$
Z'	$\top$
W	$\top$

Small progress measures (example) (15)

Step 15: $\varrho \leftarrow Lift_Y(\varrho, Y) = \varrho[Y := \max\{Prog(\varrho, Y, W), Prog(\varrho, Y, Y')\}] = \varrho[Y := \max\{\top, (0,0,0,0)\}] = \varrho[Y := \top]$

v	$\varrho(v)$
X	$\top$
X'	$(0,1,0,0)$
Y	$\top$
Y'	$(0,0,0,0)$
Z	$\top$
Z'	$\top$
W	$\top$

2013-10-03
Algorithms for Model Checking (2IW55) Lecture 7:Solving parity games using smallprogress measures
└─Small progress measures
  └─Example
    └─The algorithm

The algorithm
!
Compute least game parity progress measure using fixed point approximation:
Algorithm $SPM(G, \bigcirc)$
$\varrho : V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\dots,0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
$\varrho \leftarrow Lift_v(\varrho)$
end while
Post condition:
- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Small progress measures (example) (16)

Step 16: $\varrho \leftarrow Lift_{X'}(\varrho) = \varrho[X' := \min\{Prog(\varrho, X', Z), Prog(\varrho, X', Y)\}] = \varrho[X' := \min\{\top, \top\}] = \varrho[X' := \top]$

v	$\varrho(v)$
X	$\top$
X'	$\top$
Y	$\top$
Y'	$(0,0,0,0)$
Z	$\top$
Z'	$\top$
W	$\top$

Small progress measures (example) (17)

Step 17: $\varrho \leftarrow Lift_{Y'}(\varrho) = \varrho[Y' := \min\{Prog(\varrho, Y', X), Prog(\varrho, Y', Y)\}] = \varrho[Y' := \min\{\top, \top\}] = \varrho[Y' := \top]$

v	$\varrho(v)$
X	$\top$
X'	$\top$
Y	$\top$
Y'	$\top$
Z	$\top$
Z'	$\top$
W	$\top$

ϱ is least game parity progress measure, and $\{v \in V \mid \varrho(v) \neq \top\} = \emptyset$ is winning set for player $\Diamond$. Hence player $\Box$ wins from all vertices

2013-10-03
Algorithms for Model Checking (2IW55) Lecture 7:Solving parity games using smallprogress measures
└Small progress measures
└Example
└The algorithm

The algorithm

Compute least game parity progress measure using fixed point approximation:

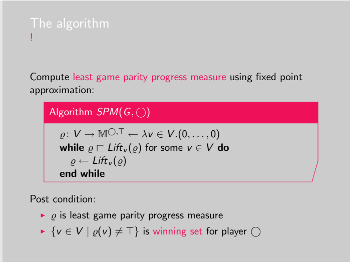
Algorithm $SPM(G, \bigcirc)$
$\varrho: V \to \mathbb{M}^{\bigcirc,\top} \leftarrow \lambda v \in V.(0,\ldots,0)$
while $\varrho \sqsubset Lift_v(\varrho)$ for some $v \in V$ **do**
$\varrho \leftarrow Lift_v(\varrho)$
end while

Post condition:
- ϱ is least game parity progress measure
- $\{v \in V \mid \varrho(v) \neq \top\}$ is winning set for player $\bigcirc$

Strategies from progress measures

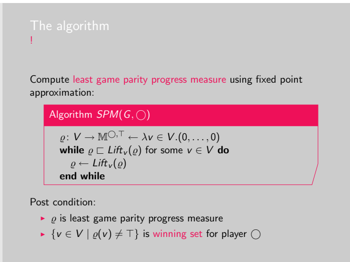
Let $G = (V, E, p, (V_\Diamond, V_\Box))$ be a parity game, and $\varrho: V \to \mathbb{M}^{\bigcirc,\top}$ be least game parity progress measure.

- Define strategy $\overline{\varrho}: V_\Diamond \to V$ for player $\Diamond$, by setting $\overline{\varrho}(v)$ to be a successor w of $v \in V_\Diamond$ that minimises $\varrho(w)$.
- $\overline{\varrho}$ is a winning strategy for player $\Diamond$ from $\{v \in V \mid \varrho(v) \neq \top\}$.



Strategy (example)

- As the winning set for player ◇ is empty, the strategy for player ◇ can be chosen arbitrarily
- Stategy for player □ cannot be inferred directly (winning set *can* be determined), some tricks have to be applied...



Complexity

Let $G = (V, E, p, (V_\diamond, V_\Box))$ be a parity game;
$n = |V|, e = |E|, d = \max\{p(v) \mid v \in V\}$.

Worst-case running time complexity:

$$\mathcal{O}(de \cdot (\frac{n}{\lfloor d/2 \rfloor})^{\lfloor d/2 \rfloor})$$

Lowerbound on worst-case:

$$\Omega((\lceil n/d \rceil)^{\lceil d/2 \rceil})$$

Summary

- Parity games
- Relation to Boolean Equation Systems
- Link to model checking
- Simplification techniques (self-loop elim. priority compaction/propagation)
- Solving:
  - Recursive $\mathcal{O}(en^d)$
  - Small progress measures $\mathcal{O}(de \cdot (\frac{n}{\lfloor d/2 \rfloor})^{\lfloor d/2 \rfloor})$
  - bigstep (combination of the two above): $\mathcal{O}(n^{d/3})$