Yet another Bisimulation Application

Roland Glück

Universität Augsburg

18.07.2011
Sion

Overview

- stochastic games
- bisimulations
- synthesis
- conclusion and outlook

Definition

Definition

A *stochastic game* is a graph $G = (V, E)$ with

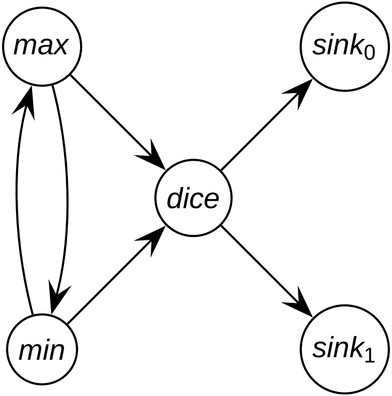
- $V = V_{max} \dot{\cup} V_{min} \dot{\cup} V_{dice} \dot{\cup} \{sink_0\} \dot{\cup} \{sink_1\}$
- $\forall v \in V, v \notin \{sink_0, sink_1\} : deg_{out}(v) \in \{1, 2\}$
- $deg_{out}(sink_0) = deg_{out}(sink_1) = 0$

w.l.o.g. $V = \{1, \ldots, n\}$, $sink_0 = n - 1$, $sink_1 = n$

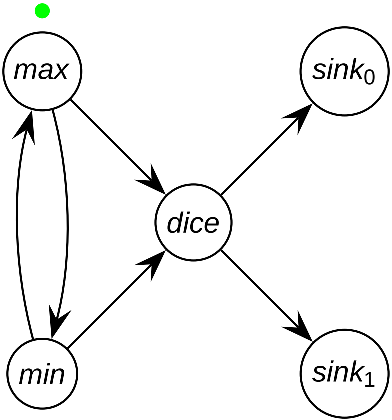
Rules

- token placed on a start node and moved along edges
- on min(max)-node, min(max)-player chooses one edge
- on dice-nodes one edge is chosen arbitrarily
- games ends if tokens reaches sink-node
- at $sink_0$ ($sink_1$) min(max)-player wins

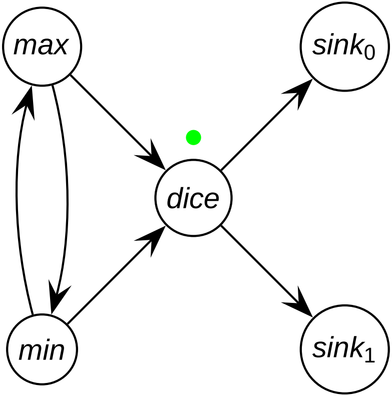
Example Game



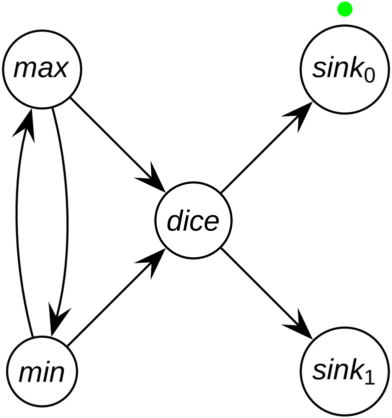
Example Game



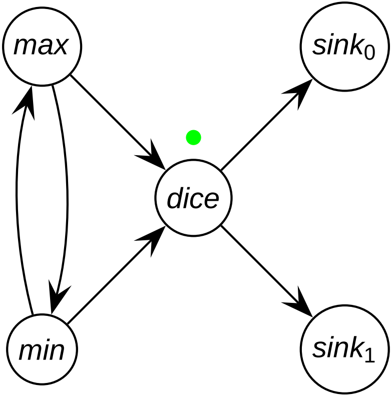
Example Game



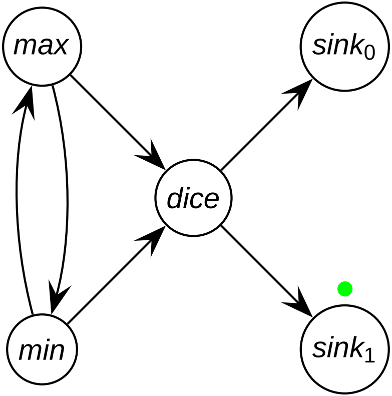
Example Game



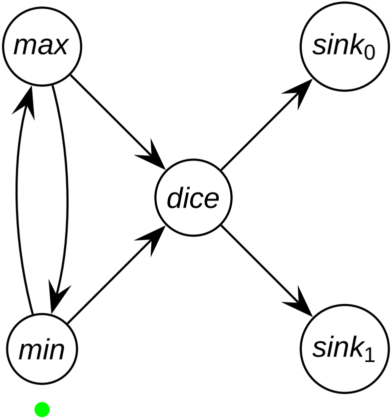
Example Game



Example Game



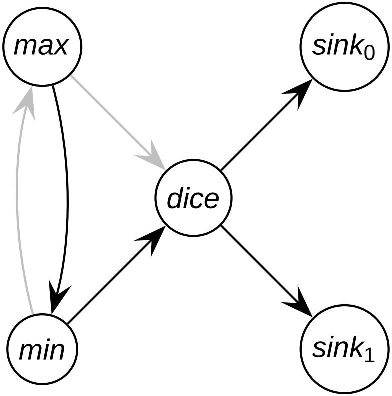
Example Game



Strategy

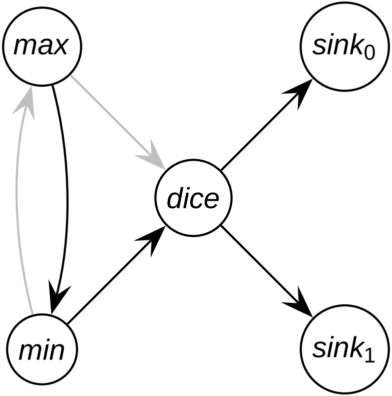
- *min-strategy* τ: $\tau \subseteq V_{min} \times V : \forall v \in V_{min} : deg^{\tau}_{out}(v) = 1$
- *max-strategy* σ: $\sigma \subseteq V_{max} \times V : \forall v \in V_{max} : deg^{\sigma}_{out}(v) = 1$
- corresponds to Markov process
- $val_{\sigma,\tau}(v) = P($max-player wins under strategies σ and τ if game starts in node $v)$
- model refinement

Example Game



$\sigma = \{(max, min)\}, \tau = \{(min, dice)\}$

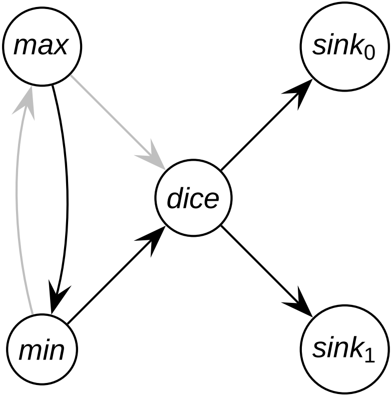
Example Game



$val_{\sigma,\tau}(sink_0) = 0$

$\sigma = \{(max, min)\}$, $\tau = \{(min, dice)\}$

Example Game

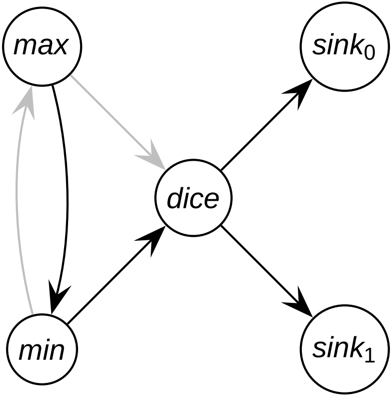


$val_{\sigma,\tau}(sink_0) = 0$

$val_{\sigma,\tau}(sink_1) = 1$

$\sigma = \{(max, min)\}, \tau = \{(min, dice)\}$

Example Game



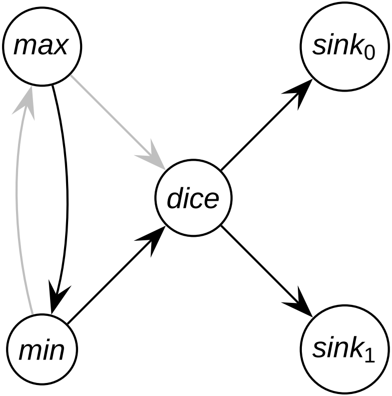
$val_{\sigma,\tau}(sink_0) = 0$

$val_{\sigma,\tau}(sink_1) = 1$

$val_{\sigma,\tau}(dice) = 0.5$

$\sigma = \{(max, min)\}, \tau = \{(min, dice)\}$

Example Game



$$val_{\sigma,\tau}(sink_0) = 0$$

$$val_{\sigma,\tau}(sink_1) = 1$$

$$val_{\sigma,\tau}(dice) = 0.5$$

$$val_{\sigma,\tau}(max) = 0.5$$

$$val_{\sigma,\tau}(min) = 0.5$$

$\sigma = \{(max, min)\}$, $\tau = \{(min, dice)\}$

optimal value

- $val_{opt}(v) = max_\sigma min_\tau val_{\sigma,\tau}(v)$
- $max_\sigma min_\tau val_{\sigma,\tau}(v) = min_\tau max_\sigma val_{\sigma,\tau}(v)$
- given σ and τ, *val* can be computed easily(and is unique)
- computation of optimal value is in $NP \cap coNP$
- key idea: optimality of given strategy can be checked easily
- no provably polynomial algorithm known

Definition

$B \subseteq V_1 \times V_2$ is a *bisimulation* between two graphs (V_1, E_1) and (V_2, E_2) iff

- $Dom(B) = X_1$ and $Cod(B) = X_2$
- $v_1 B v_2 \wedge v_1 E_1 w_1 \Rightarrow \exists w_2 : w_1 B w_2 \wedge v_2 E_2 w_2$
- $v_2 B^{\smile} v_1 \wedge v_2 E_2 w_2 \Rightarrow \exists w_1 : w_2 B^{\smile} w_1 \wedge v_1 E_1 w_1$

relational definition:

- $B^{\smile}; E_1 \subseteq E_2; B^{\smile} \ \wedge \ B; E_2 \subseteq E_1; B$

Coarsest Bisimulation

- bisimulations between G and itself are closed under
  - union,
  - composition, and
  - taking the converse
- identity is a bisimulation between G and itself
- existence of a *coarsest bisimulation equivalence on G*

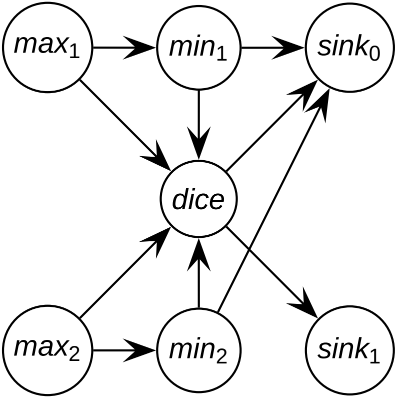
Compatible Bisimulations

- bisimulation equivalence B respects partition $V = \dot{\bigcup}_{i \in I} V_i$ if every V_i is the union of suitable equivalence classes of B
- for every partition of V exists a coarsest respecting bisimulation
- here main interest in bisimulations respecting $\{V_{max}, V_{min}, V_{dice}, \{sink_0\}, \{sink_1\}\}$

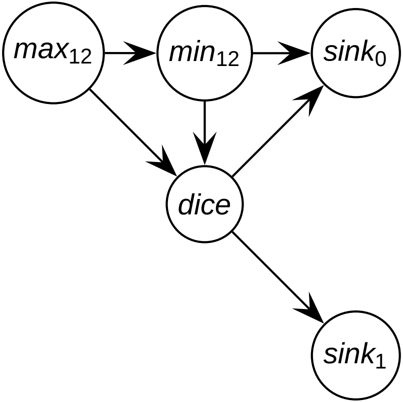
Quotient Graph

- for a bisimulation equivalence B and a graph $G = (V, E)$ the *quotient* $G/B = (V/B, E/B)$ is defined by
  - V/B is the set of equivalence classes of B
  - $(v/B, w/B) \in E/B \Leftrightarrow (v, w) \in E$
- G/B has in general a smaller node set then G
- *coarsest quotient* respecting T induced by coarsest bisimulation respecting $\{T, V - T\}$

Example Quotient

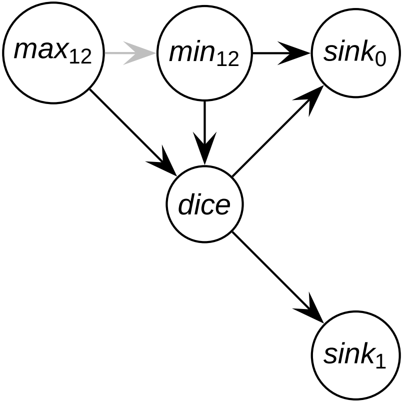


Example Quotient



- idea: compute val-function and optimal strategy on the quotient and expand solution
- missing: expansion operation
- solution: for a subgraph $(G/B)' = (V/B, (E/B)')$ of G/B define the *expansion* $(G/B)'\backslash B = (V', E')$ by
  - $V' = V$
  - $(v_1, v_2) \in E' \Leftrightarrow (v_1, v_2) \in E \wedge (v_1/B, v_2/B) \in (E/B)'$

Optimal Quotient Strategy



Example Quotient