

Stopping Criteria for Value Iteration for Concurrent Stochastic Reachability and Safety Games

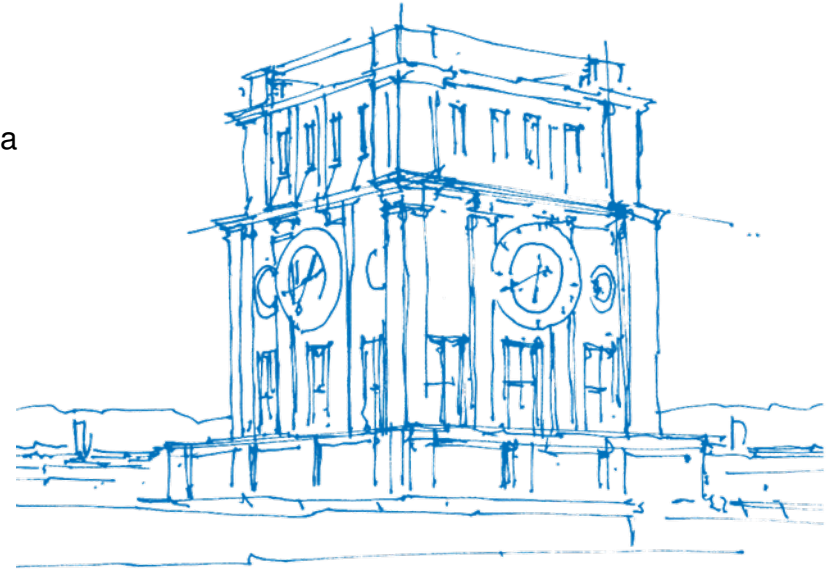
Marta Grobelna¹, Jan Křetínský², Maximilian Weininger³

¹Technical University of Munich

²Masaryk University Brno

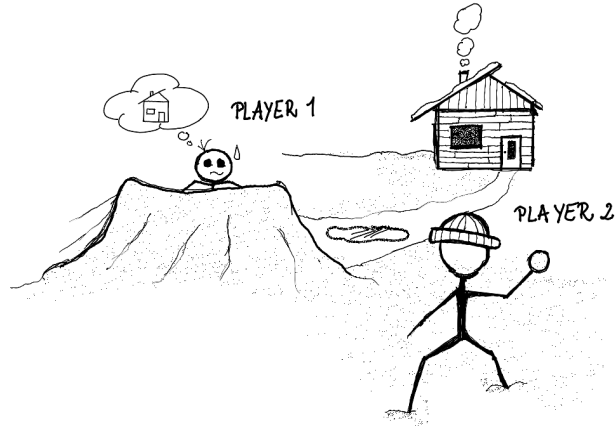
³Ruhr University Bochum \Institute of Science and Technology Austria

Eindhoven, 24. Oktober 2026

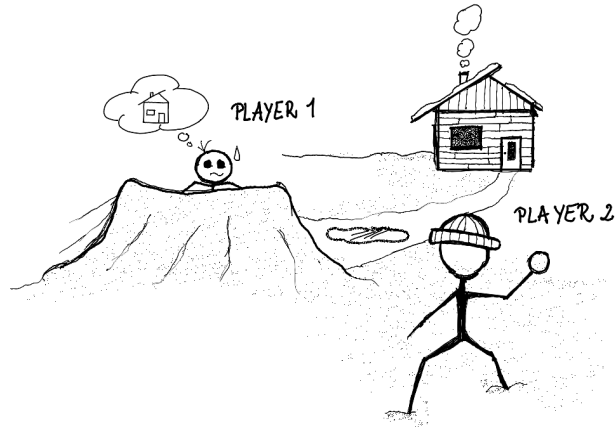


TUM Uhrenturm

Concurrent Stochastic Reachability Games (CSGs)

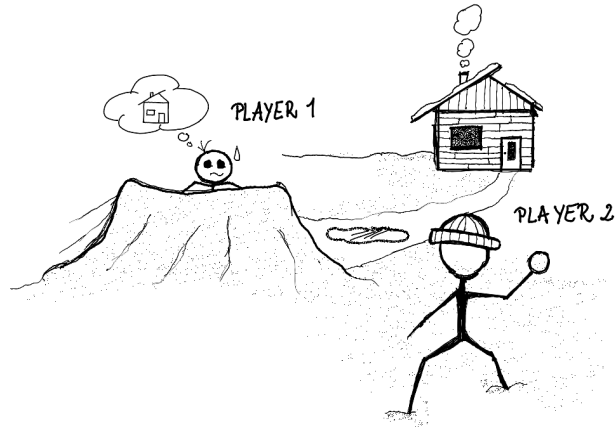


Concurrent Stochastic Reachability Games (CSGs)



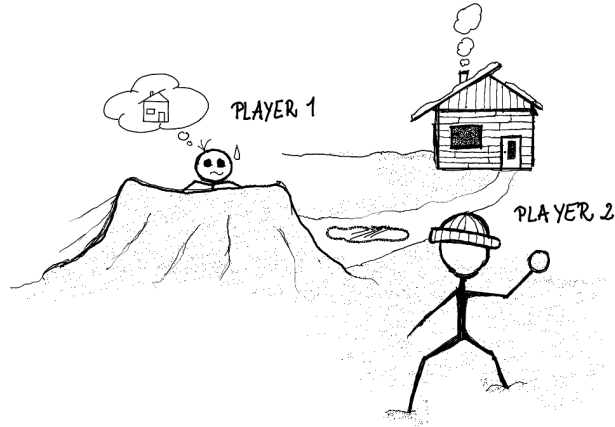
- Player 1: **Maximize** probability of reaching home, i.e. **reachability**

Concurrent Stochastic Reachability Games (CSGs)

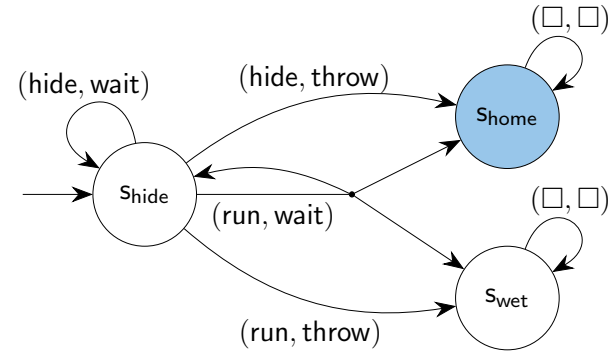


- Player 1: **Maximize** probability of reaching home, i.e. **reachability**
- Player 2: **Minimize** probability of reaching home

Concurrent Stochastic Reachability Games (CSGs)

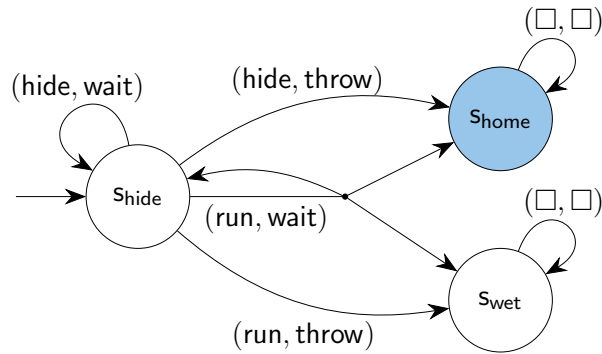


- Player 1: **Maximize** probability of reaching home, i.e. **reachability**
- Player 2: **Minimize** probability of reaching home



- $\mathcal{S} := \{S_{\text{hide}}, S_{\text{home}}, S_{\text{wet}}\}$
- $A := \{\text{hide}, \text{run}\}$
- $B := \{\text{wait}, \text{throw}\}$
- $S_0 := S_{\text{hide}}$
- $T := \{S_{\text{home}}\}$

CSG: Sequence of Two-player Zero-sum Matrix Games



$$V(S_{\text{hide}}) = V \left(\begin{array}{cc} \text{throw} & \text{wait} \\ \begin{pmatrix} 0 & \frac{1}{3} \cdot V(S_{\text{hide}}) + \frac{1}{3} \\ 1 & V(S_{\text{hide}}) \end{pmatrix} & \begin{pmatrix} \text{run} \\ \text{hide} \end{pmatrix} \end{array} \right)$$

Solving CSGs using Value Iteration

- **Value iteration:** Initialize with lowest value and iteratively update (Bellman update)

Solving CSGs using Value Iteration

- **Value iteration:** Initialize with lowest value and iteratively update (Bellman update)

$$\mathbf{L}^0(s) = \begin{cases} 1, & \text{if } s \in \mathbb{T}; \\ 0, & \text{else.} \end{cases}$$

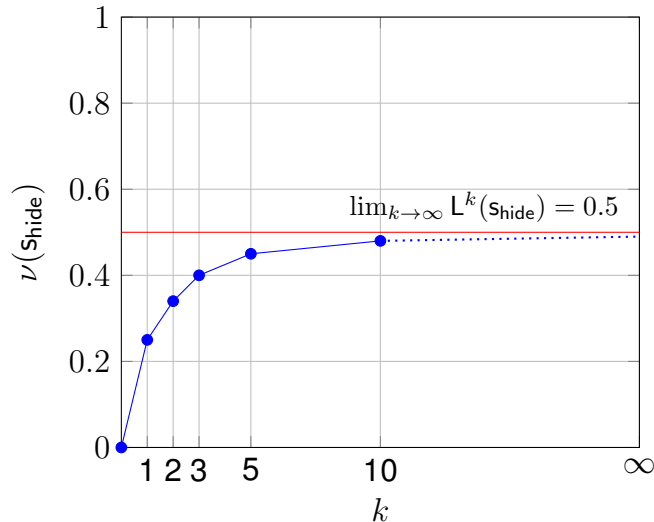
$$\mathbf{L}^{k+1}(s) = \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \sum_{s' \in \mathcal{S}} \sum_{(a,b) \in \mathcal{A}} \delta(s, a, b)(s') \cdot \mathbf{L}^k(s') \cdot \rho(a) \cdot \sigma(b) =: \mathfrak{B}(\mathbf{L}^k)(s)$$

Solving CSGs using Value Iteration

- **Value iteration:** Initialize with lowest value and iteratively update (Bellman update)

$$\mathbf{L}^0(s) = \begin{cases} 1, & \text{if } s \in \mathcal{T}; \\ 0, & \text{else.} \end{cases}$$

$$\mathbf{L}^{k+1}(s) = \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \sum_{s' \in \mathcal{S}} \sum_{(a,b) \in \mathcal{A}} \delta(s, a, b)(s') \cdot \mathbf{L}^k(s') \cdot \rho(a) \cdot \sigma(b) =: \mathfrak{B}(\mathbf{L}^k)(s)$$

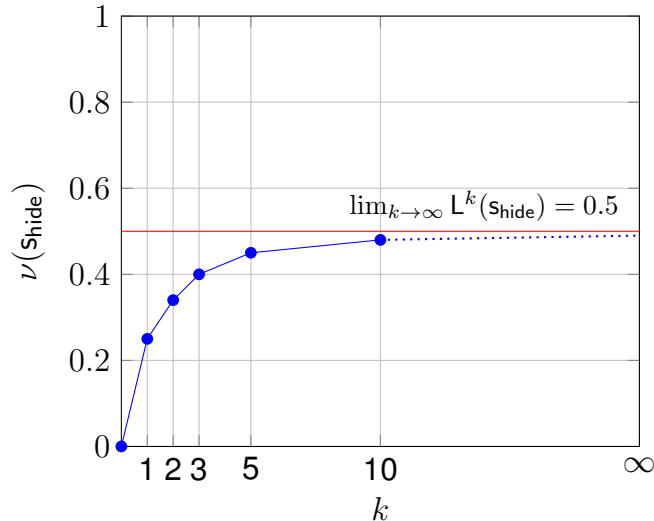


Solving CSGs using Value Iteration

- **Value iteration:** Initialize with lowest value and iteratively update (Bellman update)

$$\mathbf{L}^0(s) = \begin{cases} 1, & \text{if } s \in T; \\ 0, & \text{else.} \end{cases}$$

$$\mathbf{L}^{k+1}(s) = \sup_{\rho \in \mathcal{R}(s)} \inf_{\sigma \in \mathcal{S}(s)} \sum_{s' \in \mathcal{S}} \sum_{(a,b) \in \mathcal{A}} \delta(s, a, b)(s') \cdot \mathbf{L}^k(s') \cdot \rho(a) \cdot \sigma(b) =: \mathfrak{B}(\mathbf{L}^k)(s)$$



- **Stopping criterion:** Two consecutive approximations are ε -close
- **No precision guarantees**

Bounded Value Iteration for Precision Guarantees

- BVI: calculate lower **and** upper bound

$$L^0(s) = \begin{cases} 1, & \text{if } s \in T; \\ 0, & \text{else.} \end{cases} \quad L^{k+1}(s) = \mathfrak{B}(L^k)(s)$$

$$U^0(s) = \begin{cases} 0, & \text{if } s \in W_2; \\ 1, & \text{else.} \end{cases} \quad U^{k+1}(s) = \mathfrak{B}(U^k)(s)$$

- W_2 : winning region of Player 2 (e.g. state s_{wet})

Bounded Value Iteration for Precision Guarantees

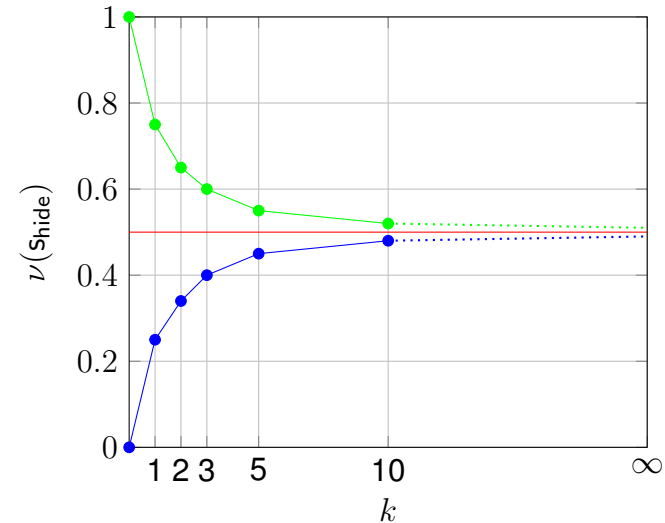
- BVI: calculate lower **and** upper bound

$$L^0(s) = \begin{cases} 1, & \text{if } s \in T; \\ 0, & \text{else.} \end{cases} \quad L^{k+1}(s) = \mathfrak{B}(L^k)(s)$$

$$U^0(s) = \begin{cases} 0, & \text{if } s \in W_2; \\ 1, & \text{else.} \end{cases} \quad U^{k+1}(s) = \mathfrak{B}(U^k)(s)$$

- W_2 : winning region of Player 2 (e.g. state s_{wet})

$$\lim_{k \rightarrow \infty} L^k(s_{hide}) = \lim_{k \rightarrow \infty} U^k(s_{hide}) = 0.5$$



Bounded Value Iteration for Precision Guarantees

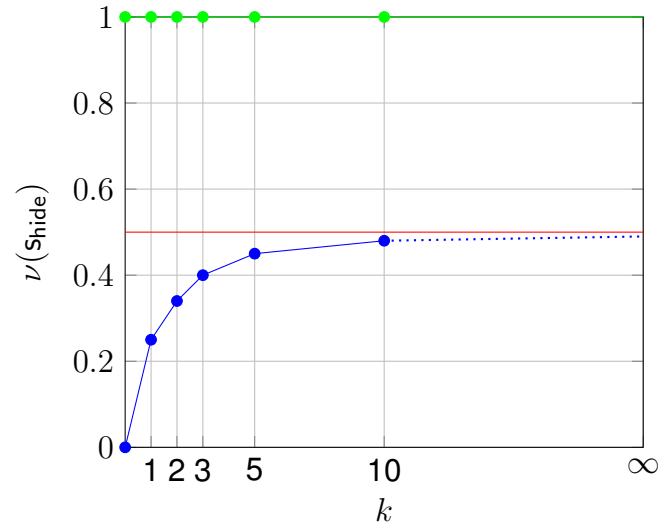
- BVI: calculate lower **and** upper bound

$$L^0(s) = \begin{cases} 1, & \text{if } s \in T; \\ 0, & \text{else.} \end{cases} \quad L^{k+1}(s) = \mathfrak{B}(L^k)(s)$$

$$U^0(s) = \begin{cases} 0, & \text{if } s \in W_2; \\ 1, & \text{else.} \end{cases} \quad U^{k+1}(s) = \mathfrak{B}(U^k)(s)$$

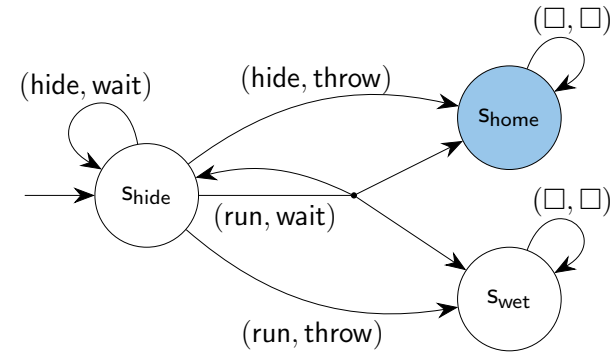
- W_2 : winning region of Player 2 (e.g. state s_{wet})

$$\lim_{k \rightarrow \infty} L^k(s_{hide}) = \underline{0.5} \neq 1 = \lim_{k \rightarrow \infty} U^k(s_{hide})$$



Hazardous and Trapping Strategies Hinder Convergence

$$Z_{U^k}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & & \text{wait} \\ 0 & \frac{1}{3} \cdot U^k(s_{\text{hide}}) + \frac{1}{3} & \text{run} \\ 1 & U^k(s_{\text{hide}}) & \text{hide} \end{pmatrix}$$

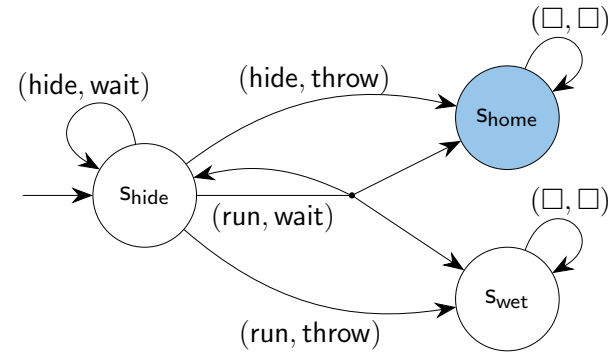


Hazardous and Trapping Strategies Hinder Convergence

$$Z_{U^k}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} & \text{run} \\ 0 & \frac{1}{3} \cdot U^k(s_{\text{hide}}) + \frac{1}{3} & \text{hide} \\ 1 & U^k(s_{\text{hide}}) & \end{pmatrix}$$

- **Player 1:**

$$Z_{U^0}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} & \text{run} \\ 0 & \frac{2}{3} & \text{hide} \\ 1 & 1 & \end{pmatrix}$$



Hazardous and Trapping Strategies Hinder Convergence

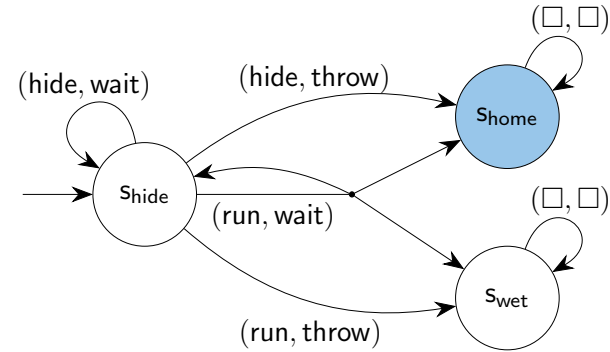
$$Z_{U^k}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} & \text{run} \\ 0 & \frac{1}{3} \cdot U^k(s_{\text{hide}}) + \frac{1}{3} & \text{hide} \\ 1 & U^k(s_{\text{hide}}) & \end{pmatrix}$$

- **Player 1:**

$$Z_{U^0}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} & \text{run} \\ 0 & \frac{2}{3} & \text{hide} \\ 1 & 1 & \end{pmatrix}$$

- Hazardous strategy: $\rho := \{\mathbf{hide} \mapsto 1\}$ is **optimal** and

...



Hazardous and Trapping Strategies Hinder Convergence

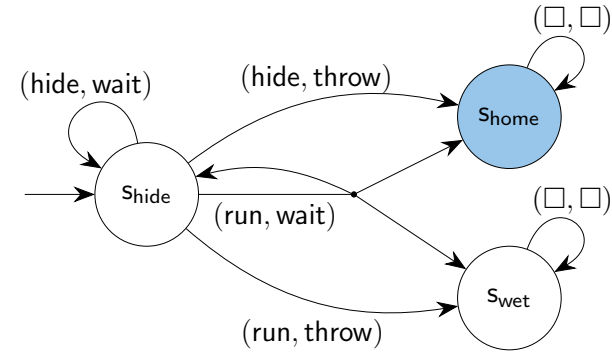
$$Z_{U^k}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} \\ 0 & \frac{1}{3} \cdot U^k(s_{\text{hide}}) + \frac{1}{3} \\ 1 & U^k(s_{\text{hide}}) \end{pmatrix} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

- **Player 1:**

$$Z_{U^0}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} \\ 0 & \frac{2}{3} \\ 1 & 1 \end{pmatrix} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

- Hazardous strategy: $\rho := \{\mathbf{hide} \mapsto 1\}$ is **optimal** and

...



- **Player 2:**

$$Z_{U^0}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} \\ 0 & \frac{2}{3} \\ 1 & 1 \end{pmatrix} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

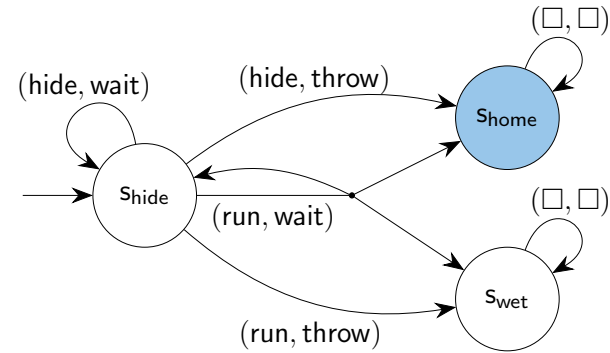
Hazardous and Trapping Strategies Hinder Convergence

$$Z_{U^k}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} \\ 0 & \frac{1}{3} \cdot U^k(s_{\text{hide}}) + \frac{1}{3} \\ 1 & U^k(s_{\text{hide}}) \end{pmatrix} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

- **Player 1:**

$$Z_{U^0}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} \\ 0 & \frac{2}{3} \\ 1 & 1 \end{pmatrix} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

- Hazardous strategy: $\rho := \{\mathbf{hide} \mapsto 1\}$ is **optimal** and ...



- **Player 2:**

$$Z_{U^0}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} \\ 0 & \frac{2}{3} \\ 1 & 1 \end{pmatrix} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

- Trapping strategy: $\sigma := \{\mathbf{wait} \mapsto 1\}$ is **optimal** wrt. ρ and ...

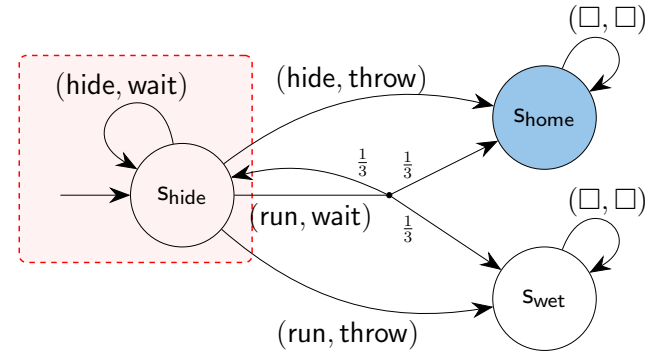
Hazardous and Trapping Strategies Hinder Convergence

$$Z_{U^k}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} \\ 0 & \frac{1}{3} \cdot U^k(s_{\text{hide}}) + \frac{1}{3} \\ 1 & U^k(s_{\text{hide}}) \end{pmatrix} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

• Player 1:

$$Z_{U^0}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} \\ 0 & \frac{2}{3} \\ 1 & 1 \end{pmatrix} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

- Hazardous strategy: $\rho := \{\text{hide} \mapsto 1\}$ is **optimal** and ...



• Player 2:

$$Z_{U^0}(s_{\text{hide}}) = \begin{pmatrix} \text{throw} & \text{wait} \\ 0 & \frac{2}{3} \\ 1 & 1 \end{pmatrix} \begin{matrix} \text{run} \\ \text{hide} \end{matrix}$$

- Trapping strategy: $\sigma := \{\text{wait} \mapsto 1\}$ is **optimal** wrt. ρ and ...

Deflating Bloated End Components

- Hazardous and trapping strategies characterize **bloated end components (BECs)**

Deflating Bloated End Components

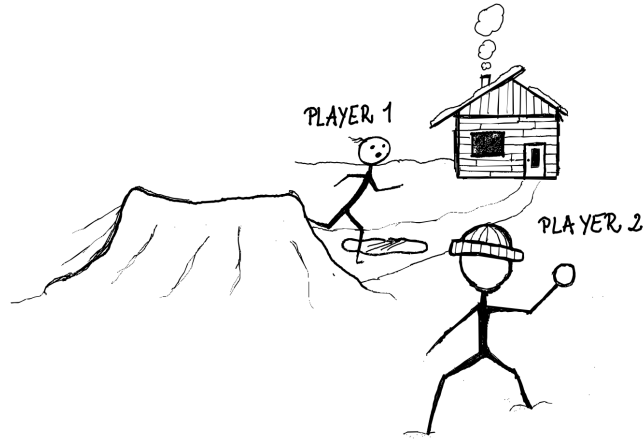
- Hazardous and trapping strategies characterize **bloated end components (BECs)**
- **Deflating**: Equalize the pressure

Deflating Bloated End Components

- Hazardous and trapping strategies characterize **bloated end components (BECs)**
- **Deflating**: Equalize the pressure
 - Estimate the **value attainable upon leaving** the BEC

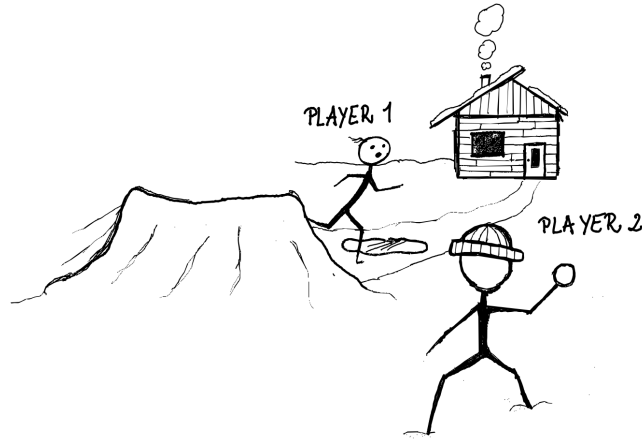
Deflating Bloated End Components

- Hazardous and trapping strategies characterize **bloated end components (BECs)**
- **Deflating**: Equalize the pressure
 - Estimate the **value attainable upon leaving** the BEC



Deflating Bloated End Components

- Hazardous and trapping strategies characterize **bloated end components (BECs)**
- **Deflating**: Equalize the pressure
 - Estimate the **value attainable upon leaving** the BEC

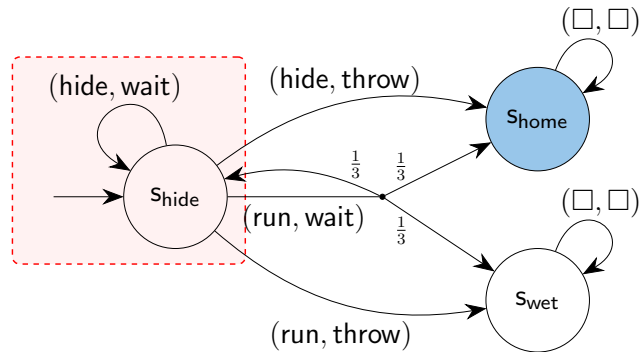


- Optimal strategy: $\rho'(\text{run}) > 0$

$$\begin{pmatrix} \text{wait} & \\ \frac{2}{3} & \text{run} \\ 1^* & \text{hide} \end{pmatrix}$$

Deflating Bloated End Components

- Hazardous and trapping strategies characterize **bloated end components (BECs)**
- **Deflating**: Equalize the pressure
 - Estimate the **value attainable upon leaving** the BEC



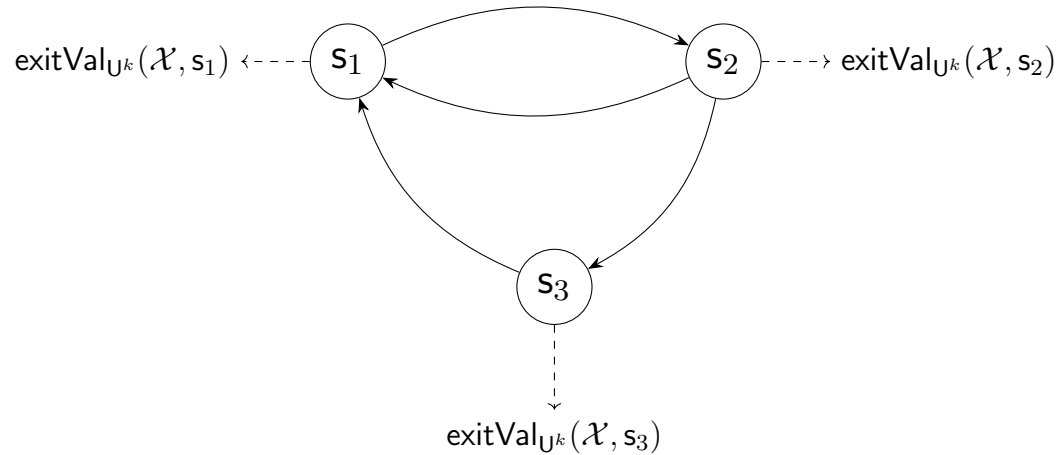
- Optimal strategy: $\rho'(\text{run}) > 0$

$$\begin{pmatrix} \text{wait} & \\ \frac{2}{3} & \text{run} \\ 1^* & \text{hide} \end{pmatrix}$$

$$\leadsto \begin{pmatrix} \text{wait} & \\ \frac{2}{3} & \text{run} \end{pmatrix}$$

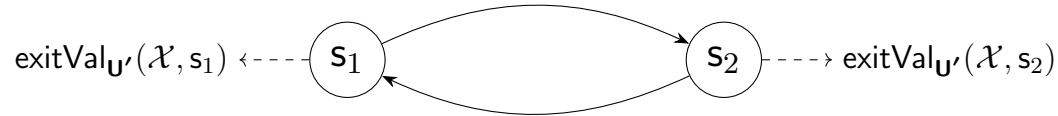
Consider the Best Exit Value in General

- Deflate with *best* exit value, i.e. $\max_{s \in \mathcal{X}} \text{exitVal}_{U^k}(\mathcal{X}, s)$ with $\mathcal{X} = \{s_1, s_2, s_3\}$



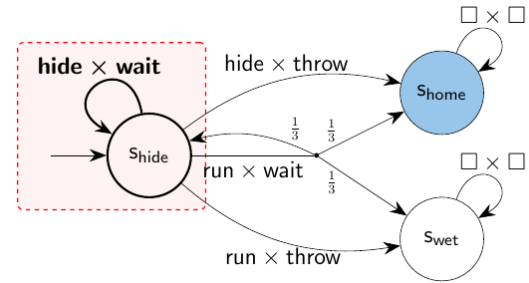
Consider the Best Exit Value in General

- Deflate with *best* exit value, i.e. $\max_{s \in \mathcal{X}} \text{exitVal}_{U^k}(\mathcal{X}, s)$ with $\mathcal{X} = \{s_1, s_2, s_3\}$
- Call recursively



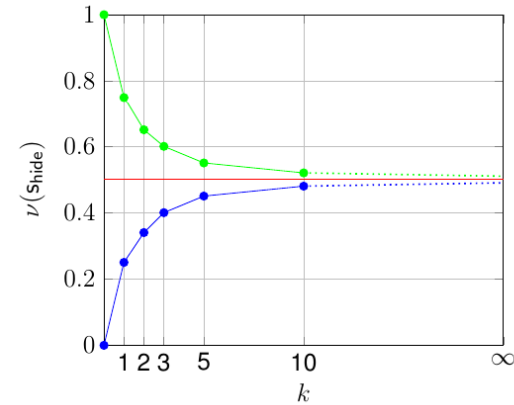
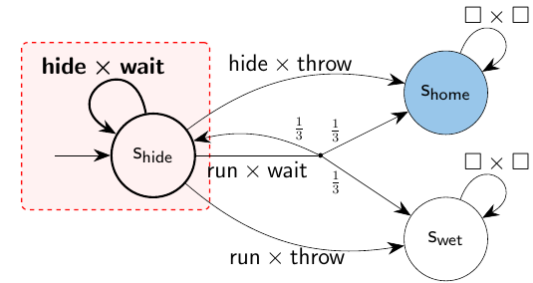
Conclusion and Outlook

- Core contribution: Definition of BECs



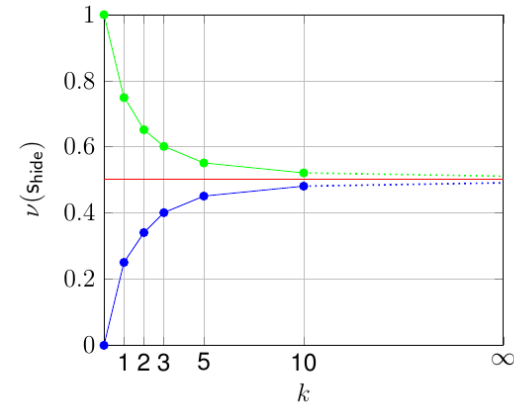
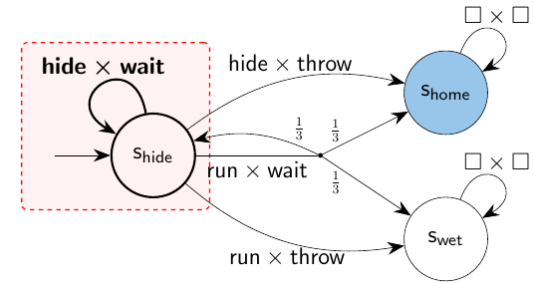
Conclusion and Outlook

- Core contribution: Definition of BECs
- Main result: Stopping criterion for value iteration for CSGs



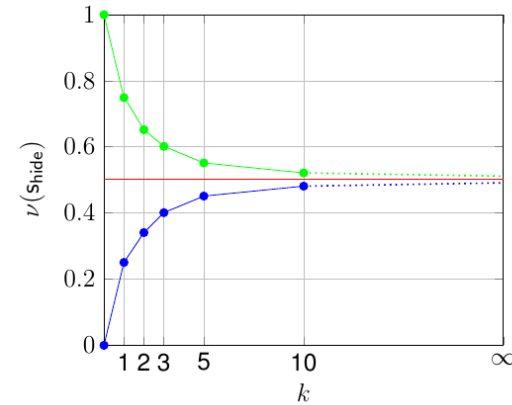
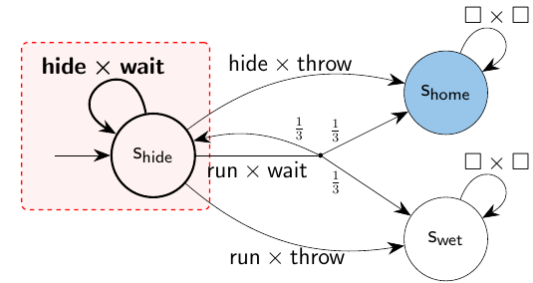
Conclusion and Outlook

- Core contribution: Definition of BECs
- Main result: Stopping criterion for value iteration for CSGs
- Implementation is pending



Conclusion and Outlook

- Core contribution: Definition of BECs
- Main result: Stopping criterion for value iteration for CSGs
- Implementation is pending
- More complex objectives & non-zero sum CSGs



Conclusion and Outlook

- Core contribution: Definition of BECs
- Main result: Stopping criterion for value iteration for CSGs
- Implementation is pending
- More complex objectives & non-zero sum CSGs

