

Winning Strategy **Templates** for Stochastic Parity Games

towards Permissive and Resilient Control

Kittiphon Phalakarn (NII)
Sasinee Pruekprasert (UTokyo)
Ichiro Hasuo (NII, SOKENDAI, Imiron)

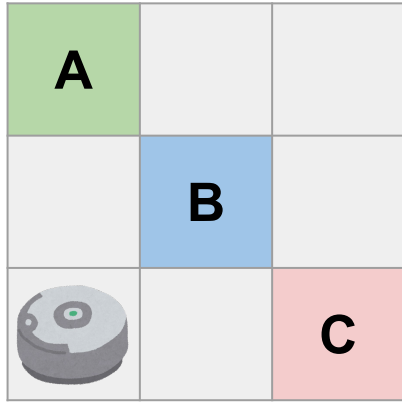
Published in
ICTAC'24, TCS'25



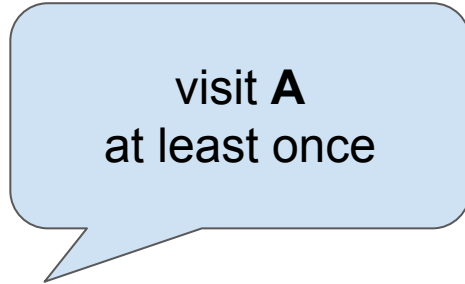
Outline

1. **Motivation**
2. Templates and Use Cases
3. Recent Results
4. Conclusion

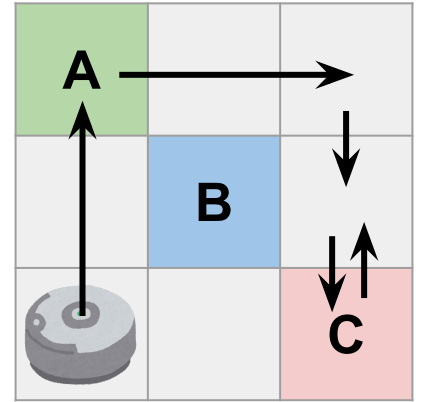
Problem Setting: **Controller Synthesis**



Input 1
Agent & Environment



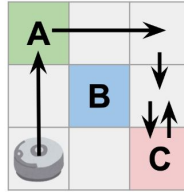
Input 2
Objective



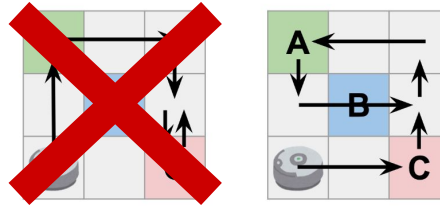
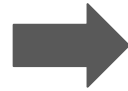
Output
"Controller"
(what to do at each state)

Research Question: Dealing with Objective Increments

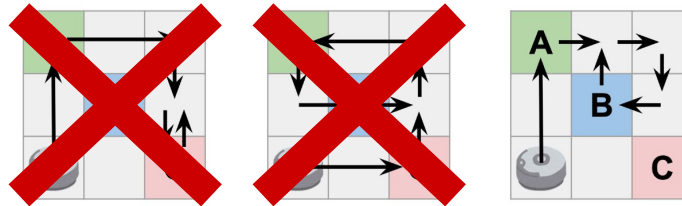
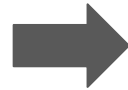
visit A
at least once



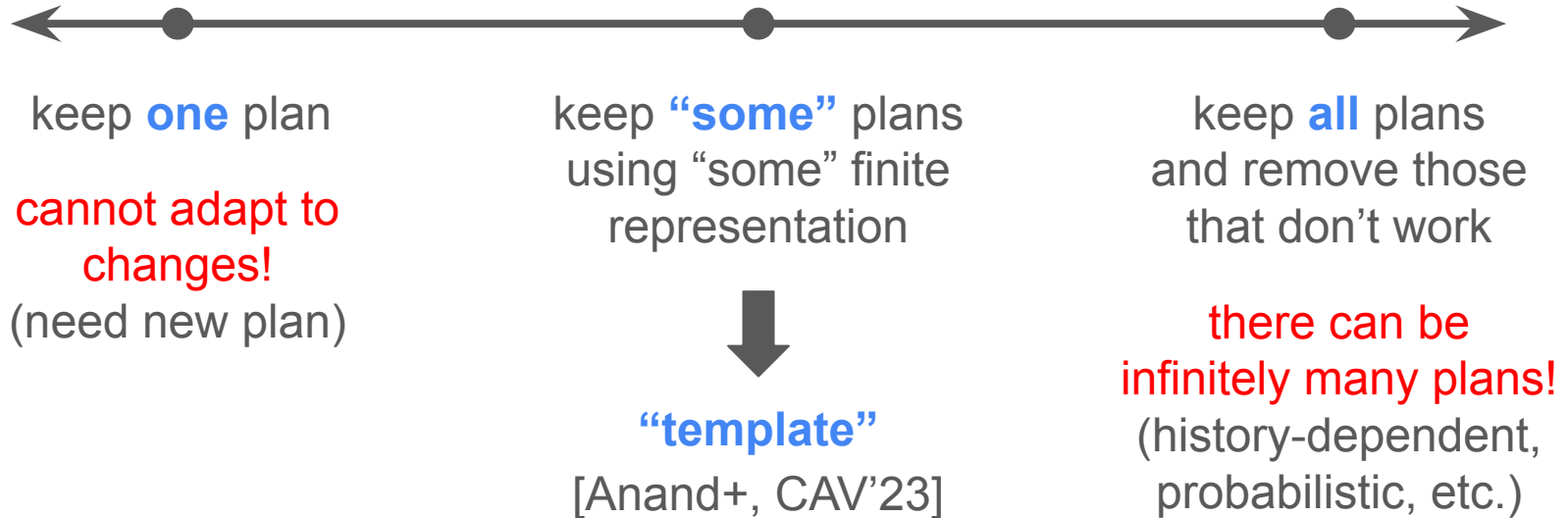
... AND
visit B
infinitely often



... AND
must NOT
visit C



Permissiveness = Adapt to Changes at Runtime!



Outline

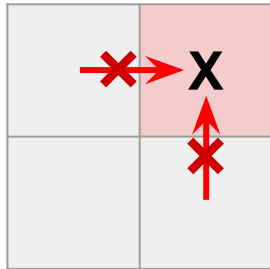
1. Motivation
- 2. Templates and Use Cases**
3. Recent Results
4. Conclusion

Template = High-Level Summary of Objectives

- Informally, a *template* consists of three sets

P for *prohibited actions*

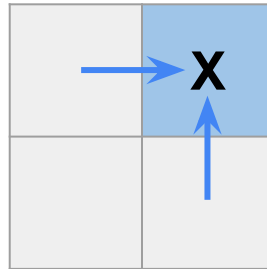
(must not take)



must not visit X

L for *live actions*

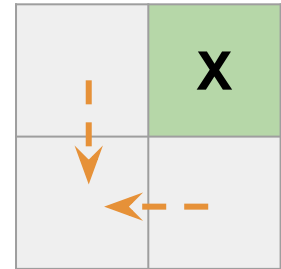
(must take infinitely often if
visiting the state infinitely often)



visit X infinitely often

C for *co-live actions*

(must take finitely often)

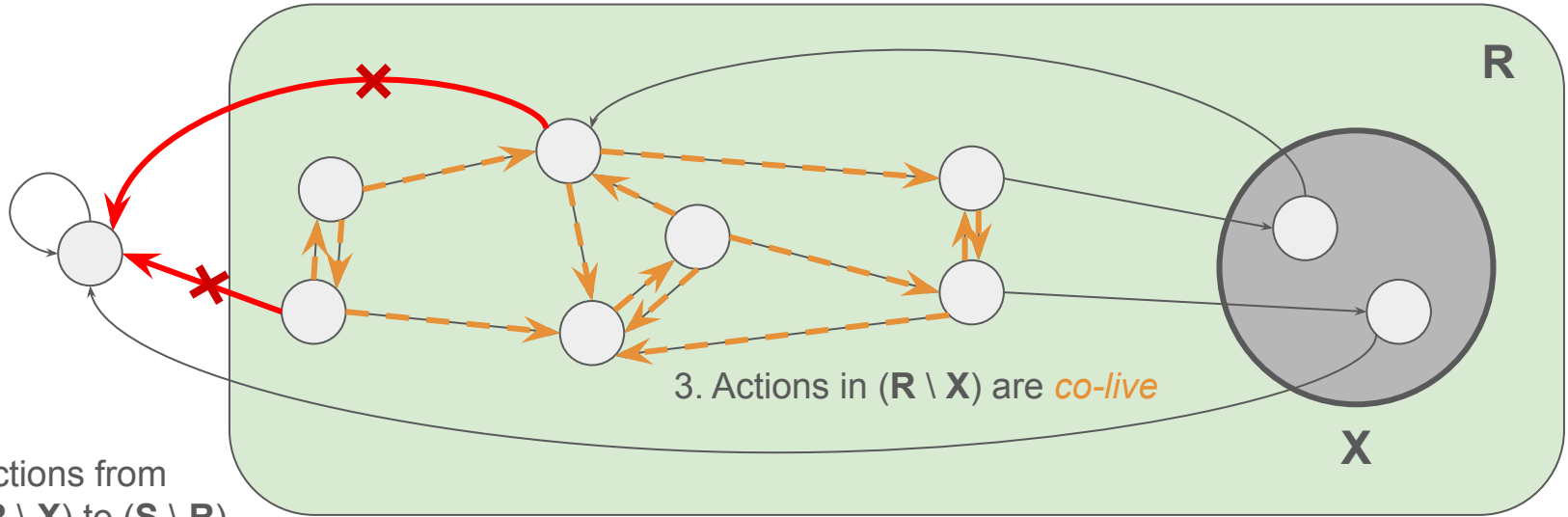


visit X at least once

Example: Template for Reachability Objective

- Reachability objective: visit X at least once

1. Compute the reachable set R



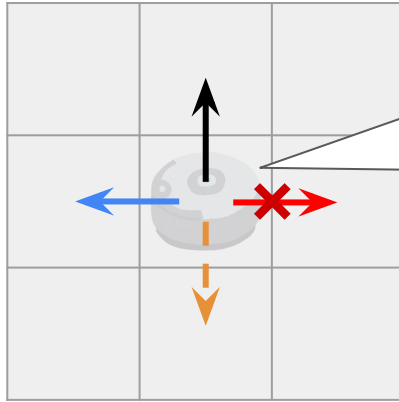
From Templates to **Controllers**

- A template suggests what to do at each state

P for *prohibited actions*
(must not take)

L for *live actions*
(must take infinitely often if
visiting the state infinitely often)

C for *co-live actions*
(must take finitely often)



Example controller

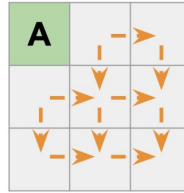
visit #1: ↓

visit #2, 4, 6, ... : ←

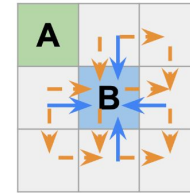
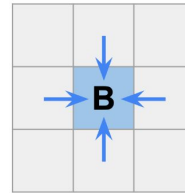
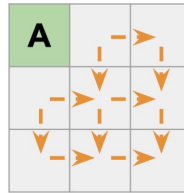
visit #3, 5, 7, ... : ↑

Use Case #1: Objective Increments

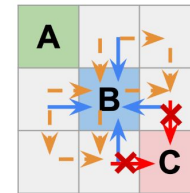
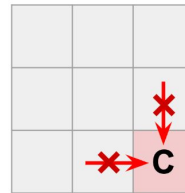
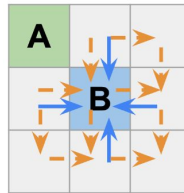
visit A
at least once



... AND
visit B
infinitely often

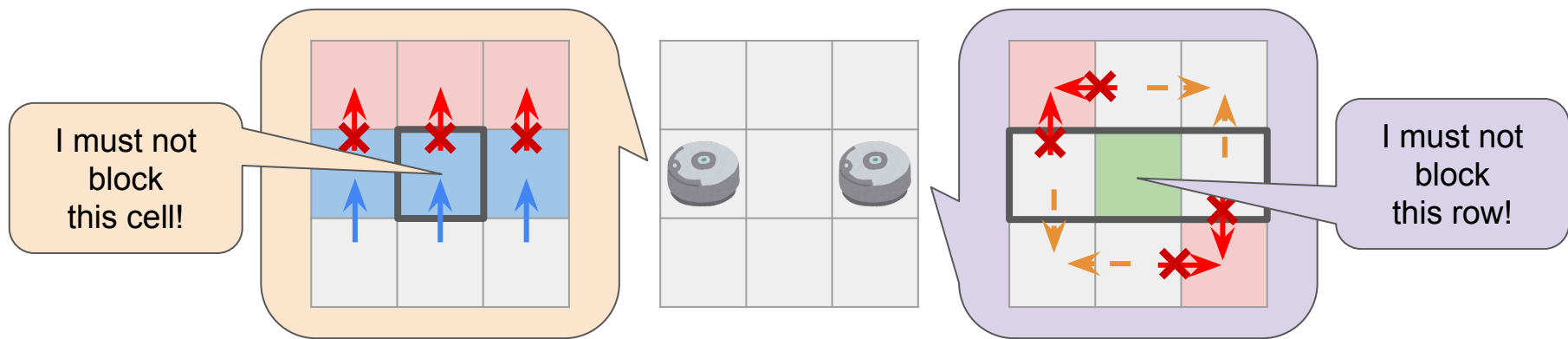


... AND
must NOT
visit C



Use Case #2: Multi-Agent Controller Synthesis

- Templates as *contracts* between collaborating agents



- See [Anand+, TACAS'23] [Anand+, HSCC'24] [Nayak–Schmuck, TACAS'24]

Outline

1. Motivation
2. Templates and Use Cases
- 3. Recent Results**
4. Conclusion

More Complex Settings: Opponent & Randomness

- So far, all examples are *1-player* games

[Anand+, CAV'23]: *2-player* game

- At some states, a *demonic opponent* controls the agent
- Algorithms were proposed to construct templates for
 - Reachability, Safety
 - Büchi, Co-Büchi
 - Parity

[P+, TCS'25]: *2.5-player* game

- At some states, a *demonic opponent* controls the agent
- At some states, *randomness* chooses an action randomly
- Our contribution is the use of [Chatterjee+, CSL'03] and [Banerjee+, TACAS'22]

Conclusion: Templates Provide Adaptability

- Templates = important actions
- Getting a controller is simple!
- Merging templates is fast!
- Various use cases!
- [Anand+, CAV'23]: 2-player game = with **opponent**
- [P+, TCS'25]: 2.5-player game = with **opponent** + **randomness**

