On Parameterized Verification of Asynchronous Shared-Memory Pushdown Systems

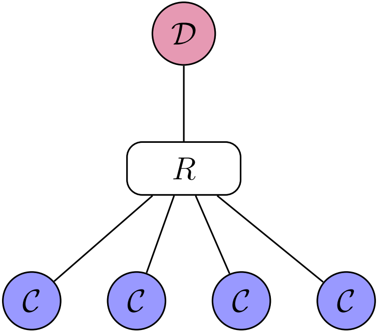
Marie Fortin[1] Anca Muscholl[2] Igor Walukiewicz[2]

[1]LSV, ENS Cachan

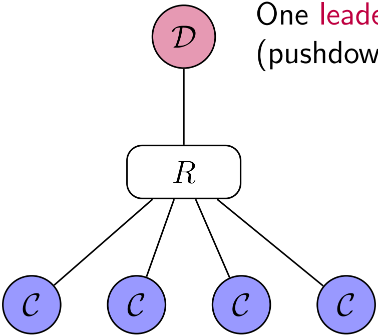
[2]LaBRI, University of Bordeaux

Highlights 2016, Brussels

$(\mathcal{C}, \mathcal{D})$-systems

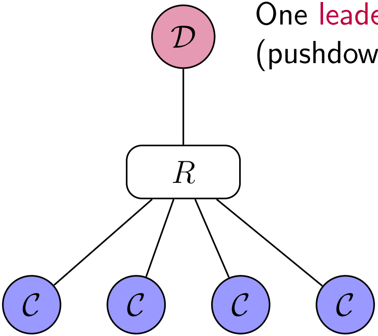


$(\mathcal{C}, \mathcal{D})$-systems



One leader process $\mathcal{D}$
(pushdown system)

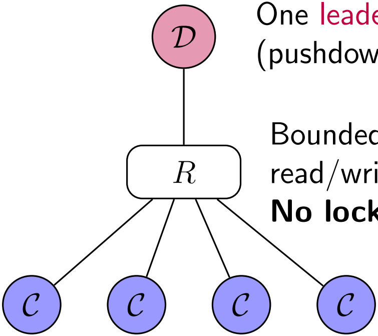
$(\mathcal{C}, \mathcal{D})$-systems



One leader process $\mathcal{D}$
(pushdown system)

Arbitrarily many identical and **anonymous** contributors $\mathcal{C}$
(pushdown systems)

$(\mathcal{C}, \mathcal{D})$-systems



One leader process $\mathcal{D}$
(pushdown system)

Bounded shared register
read/write
No lock

Arbitrarily many identical and
anonymous contributors $\mathcal{C}$
(pushdown systems)

Previous Work

Previous Work

Reachability

Is there a run of the $(\mathcal{C}, \mathcal{D})$-system where the leader performs a special action $\top$, for some number of contributors?

[Hague, 2011] EXPSPACE

[Esparza, Ganty, Majumdar, 2013] PSPACE-complete

[La Torre, Muscholl, Walukiewicz, 2015] Generalization

Previous Work

Reachability

Is there a run of the $(\mathcal{C}, \mathcal{D})$-system where the leader performs a special action $\top$, for some number of contributors?

[Hague, 2011] EXPSPACE

[Esparza, Ganty, Majumdar, 2013] PSPACE-complete

[La Torre, Muscholl, Walukiewicz, 2015] Generalization

Repeated reachability

Is there a run of the $(\mathcal{C}, \mathcal{D})$-system where the leader performs $\top$ infinitely often, for some number of contributors?

[Durand-Gasselin, Esparza, Ganty, Majumdar, 2015]
PSPACE-hard and in NEXPTIME

Repeated reachability

Repeated reachability

Theorem

The repeated reachability problem is PSPACE-complete.

Repeated reachability

Theorem

The repeated reachability problem is PSPACE-complete.

[Durand-Gasselin, Esparza, Ganty, Majumdar, 2015]:

- ▷ Reduction to the case of finite-state contributors, by bounding the stacks of the contributors
- ▷ NP in the case of finite-state contributors $\rightarrow$ NEXPTIME for pushdown contributors

Repeated reachability

Theorem

The repeated reachability problem is PSPACE-complete.

[Durand-Gasselin, Esparza, Ganty, Majumdar, 2015]:

- Reduction to the case of finite-state contributors, by bounding the stacks of the contributors
- NP in the case of finite-state contributors $\rightarrow$ NEXPTIME for pushdown contributors

We re-use the reduction to finite-state contributors, but change the decision procedure

Key steps for the PSPACE upper bound

- Look for an ultimately periodic run.

Key steps for the PSPACE upper bound

- Look for an ultimately periodic run.
- Adapt from finite runs to infinite periodic runs the techniques of [La Torre, Muscholl, Walukiewicz, 2015]:

Key steps for the PSPACE upper bound

- Look for an ultimately periodic run.
- Adapt from finite runs to infinite periodic runs the techniques of [La Torre, Muscholl, Walukiewicz, 2015]:
  - Define a transition system $\mathcal{D}^\kappa = \mathcal{D} +$ capacity: set of values written by the contributors in the register.

Idea: if a contributor can produce a value g once, by adding copies of this contributor we can produce as many g's as needed.

Key steps for the PSPACE upper bound

- Look for an ultimately periodic run.
- Adapt from finite runs to infinite periodic runs the techniques of [La Torre, Muscholl, Walukiewicz, 2015]:
    - Define a transition system $\mathcal{D}^\kappa = \mathcal{D} +$ capacity: set of values written by the contributors in the register.

Idea: if a contributor can produce a value g once, by adding copies of this contributor we can produce as many g's as needed.

- Define similarly $\mathcal{C}^\kappa$. A loop in $\mathcal{D}^\kappa$ corresponds to a loop in the $(\mathcal{C}, \mathcal{D})$-system if each addition to the capacity is supported by a loop in $\mathcal{C}^\kappa$ producing the necessary write.

Key steps for the PSPACE upper bound

- Look for an ultimately periodic run.
- Adapt from finite runs to infinite periodic runs the techniques of [La Torre, Muscholl, Walukiewicz, 2015]:
    - Define a transition system $\mathcal{D}^\kappa = \mathcal{D} +$ capacity: set of values written by the contributors in the register.

Idea: if a contributor can produce a value g once, by adding copies of this contributor we can produce as many g's as needed.

- Define similarly $\mathcal{C}^\kappa$. A loop in $\mathcal{D}^\kappa$ corresponds to a loop in the $(\mathcal{C}, \mathcal{D})$-system if each addition to the capacity is supported by a loop in $\mathcal{C}^\kappa$ producing the necessary write.
- Replace $\mathcal{D}^\kappa$ by its downard closure, and look for a supported loop in $\mathcal{D}^\kappa\downarrow$: one run of $\mathcal{D}^\kappa\downarrow + \ell$ runs of $\mathcal{C}^\kappa$.

Key steps for the PSPACE upper bound

- Look for an ultimately periodic run.
- Adapt from finite runs to infinite periodic runs the techniques of [La Torre, Muscholl, Walukiewicz, 2015]:
  - Define a transition system $\mathcal{D}^\kappa = \mathcal{D} +$ capacity: set of values written by the contributors in the register.

Idea: if a contributor can produce a value g once, by adding copies of this contributor we can produce as many g's as needed.

- Define similarly $\mathcal{C}^\kappa$. A loop in $\mathcal{D}^\kappa$ corresponds to a loop in the $(\mathcal{C}, \mathcal{D})$-system if each addition to the capacity is supported by a loop in $\mathcal{C}^\kappa$ producing the necessary write.
- Replace $\mathcal{D}^\kappa$ by its downard closure, and look for a supported loop in $\mathcal{D}^\kappa\!\downarrow$: one run of $\mathcal{D}^\kappa\!\downarrow + \ell$ runs of $\mathcal{C}^\kappa$.
  $\rightarrow$ Intersection emptiness of $\ell + 1$ finite automata computable in PSPACE.

Universal reachability

Reachability

Is there a run of the $(\mathcal{C}, \mathcal{D})$-system where the leader performs a special action $\top$, for some number of contributors?

Repeated reachability

Is there a run of the $(\mathcal{C}, \mathcal{D})$-system where the leader performs $\top$ infinitely often, for some number of contributors?

Universal reachability

Does the leader perform $\top$ in all **(finite or infinite) maximal** runs of the $(\mathcal{C}, \mathcal{D})$-system, for any number of contributors ?

Universal reachability

Motivation

Correctness problems for distributed algorithms: is the outcome correct (does the leader execute $\top$) for an arbitrary number of participants and for every run of the algorithm?

Universal reachability

Motivation

Correctness problems for distributed algorithms: is the outcome correct (does the leader execute $\top$) for an arbitrary number of participants and for every run of the algorithm?

Specificity

We consider finite maximal runs as well as infinite ones
$\rightarrow$ deadlock detection

Universal reachability

Theorem

The universal reachability problem is coNEXPTIME-complete.

Universal reachability

Theorem

The universal reachability problem is coNEXPTIME-complete.

Is there a maximal run without any occurrence of $\top$?

Universal reachability

Theorem

The universal reachability problem is co$\mathrm{N}$EXPTIME-complete.

Is there a maximal run without any occurrence of $\top$?

- For infinite runs: reduction to repeated reachability

Universal reachability

Theorem

The universal reachability problem is co$\mathrm{N}$EXPTIME-complete.

Is there a maximal run without any occurrence of $\top$?

- For infinite runs: reduction to repeated reachability
- For finite runs: use the reduction to the case of finite-state contributors, show that it is NP-complete

Universal reachability

Theorem

The universal reachability problem is co$\textsc{Nexptime}$-complete.

Is there a maximal run without any occurrence of $\top$?

- For infinite runs: reduction to repeated reachability
- For finite runs: use the reduction to the case of finite-state contributors, show that it is NP-complete
- $\textsc{Nexptime}$-hardness: tiling of the $2^n \times 2^n$ square.

Generalization

Until now: verification of properties on leader actions only.

We consider the verification of regular properties $\mathcal{P} \subseteq (\Sigma_C \cup \Sigma_D)^* \cup (\Sigma_C \cup \Sigma_D)^\omega$ on both

- finite **and** infinite traces
- leader **and** contributor actions

Generalization

Until now: verification of properties on leader actions only.

We consider the verification of regular properties $\mathcal{P} \subseteq (\Sigma_C \cup \Sigma_D)^* \cup (\Sigma_C \cup \Sigma_D)^\omega$ on both

- finite **and** infinite traces
- leader **and** contributor actions

Restriction

$\mathcal{C}$-expanding properties: if $u \in \mathcal{P}$ and u' is obtained by repeating some contributor actions in u, then $u' \in \mathcal{P}$.

Main result

Theorem

The following problem is $\textsc{Nexptime}$-complete:

Input: a $(\mathcal{C}, \mathcal{D})$-system, and a regular $\mathcal{C}$-expanding property $\mathcal{P} \subseteq (\Sigma_\mathcal{C} \cup \Sigma_\mathcal{D})^* \cup (\Sigma_\mathcal{C} \cup \Sigma_\mathcal{D})^\omega$.

Question: Is there a maximal trace of the $(\mathcal{C}, \mathcal{D})$-system that belongs to $\mathcal{P}$?

Main result

Theorem

The following problem is NEXPTIME-complete:
Input: a $(\mathcal{C}, \mathcal{D})$-system, and a regular $\mathcal{C}$-expanding property $\mathcal{P} \subseteq (\Sigma_{\mathcal{C}} \cup \Sigma_{\mathcal{D}})^* \cup (\Sigma_{\mathcal{C}} \cup \Sigma_{\mathcal{D}})^{\omega}$.
Question: Is there a maximal trace of the $(\mathcal{C}, \mathcal{D})$-system that belongs to $\mathcal{P}$?

Steps of the proof:

- Reduction to a property on leader actions, by transforming the $(\mathcal{C}, \mathcal{D})$-system into one where all contributor actions are reflected in leader writes

Main result

Theorem

The following problem is NEXPTIME-complete:
Input: a $(\mathcal{C}, \mathcal{D})$-system, and a regular $\mathcal{C}$-expanding property $\mathcal{P} \subseteq (\Sigma_{\mathcal{C}} \cup \Sigma_{\mathcal{D}})^* \cup (\Sigma_{\mathcal{C}} \cup \Sigma_{\mathcal{D}})^{\omega}$.
Question: Is there a maximal trace of the $(\mathcal{C}, \mathcal{D})$-system that belongs to $\mathcal{P}$?

Steps of the proof:

- Reduction to a property on leader actions, by transforming the $(\mathcal{C}, \mathcal{D})$-system into one where all contributor actions are reflected in leader writes
- Reduction to our previous results
  - Infinite traces: reduction to repeated reachability
  - Finite traces: results about universal reachability

Summary

Summary

- We improved the complexity upper bound for repeated reachability from NEXPTIME to PSPACE

Summary

- We improved the complexity upper bound for repeated reachability from NEXPTIME to PSPACE
- We introduced universal reachability, and showed that it is coNEXPTIME-complete

Summary

- We improved the complexity upper bound for repeated reachability from NEXPTIME to PSPACE
- We introduced universal reachability, and showed that it is coNEXPTIME-complete
- Verification of regular $\mathcal{C}$-expanding properties is also NEXPTIME-complete

Summary

- We improved the complexity upper bound for repeated reachability from NEXPTIME to PSPACE
- We introduced universal reachability, and showed that it is coNEXPTIME-complete
- Verification of regular $\mathcal{C}$-expanding properties is also NEXPTIME-complete

Thank you !