Accelerating Markov Chain Model Checking

Good-for-Games Meets Unambiguous Automata

Yong Li	Chinese Academy of Sciences
Soumyajit Paul	University of Liverpool
Sven Schewe	University of Liverpool
Qiyi Tang	University of Liverpool

CAV 2025

Probabilistic Verification

Probabilistic Verification

Guarantees for autonomous systems in uncertain environment

- Autonomous vehicles
- Communication protocols
- Secure systems
- Planning and synthesis
- ...

Markov Chain Verification

Input : Finite Markov chain $\mathcal{M}$ and ω-regular specification φ

$\mathrm{Pr}(\mathcal{M} \models \varphi)$?

State of the Art

State of the Art

Theory

Specification	*Complexity*
LTL	PSPACE-complete [Vardi'85,CY'95]
DBA	PTIME
UBA	PTIME [BKKKMW'16]
NBA	PSPACE-complete [Vardi'85,CY'95]

State of the Art

Theory

Specification	*Complexity*
LTL	PSPACE-complete [Vardi'85,CY'95]
DBA	PTIME
UBA	PTIME [BKKKMW'16]
NBA	PSPACE-complete [Vardi'85,CY'95]

Practice

Tools : PRISM, STORM, EPMC, MoChiBA, etc

Only PRISM supports MC model checking against UBA specifications as input

State of the Art

Theory

Specification	*Complexity*
LTL	PSPACE-complete [Vardi'85,CY'95]
DBA	PTIME
UBA	PTIME [BKKKMW'16]
NBA	PSPACE-complete [Vardi'85,CY'95]

Practice

Tools : PRISM, STORM, EPMC, MoChiBA, etc

Only PRISM supports MC model checking against UBA specifications as input

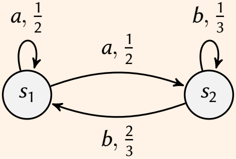
This Work

- New PTIME algorithm for MC model checking for UBA specification
- Outperforms PRISM for large inputs with provably exponential improvement in some cases

Labelled Markov Chain and Automata

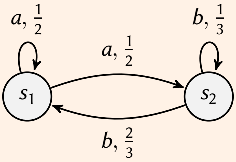
Labelled Markov Chain and Automata

Labelled Markov Chain

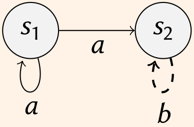


Labelled Markov Chain and Automata

Labelled Markov Chain



Non-deterministic Buchi Automata

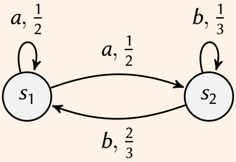


Transition based acceptance

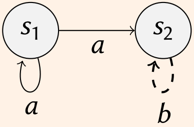
$$\mathcal{L} = aa^* b^\omega$$

Labelled Markov Chain and Automata

Labelled Markov Chain



Non-deterministic Buchi Automata



Transition based acceptance

$$\mathcal{L} = aa^* b^\omega$$

DBA : Transitions are deterministic

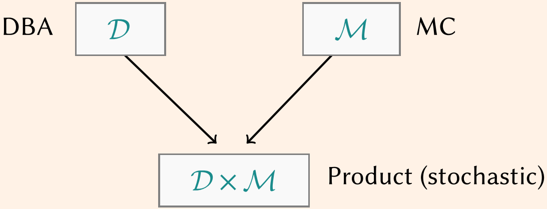
UBA : Every word has at most one accepting run

Setting

Input : Finite Markov chain $\mathcal{M}$ and UBA $\mathcal{U}$

$\Pr_{\mathcal{M}}(\mathcal{L}(\mathcal{U}))$?

Classic Algorithm for DBA



DBA $\mathcal{D}$

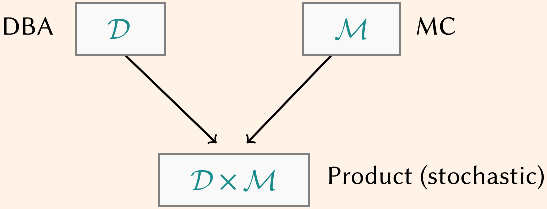
$\mathcal{M}$ MC

$\mathcal{D} \times \mathcal{M}$ Product (stochastic)

Pr (Reach bottom SCCs containing accepting transitions)

Standard steady state linear equation system + Constraints for BSCCs

Classic Algorithm for DBA



DBA $\mathcal{D}$

$\mathcal{M}$ MC

$\mathcal{D} \times \mathcal{M}$ Product (stochastic)

Pr (Reach bottom SCCs containing accepting transitions)

Standard steady state linear equation system + Constraints for BSCCs

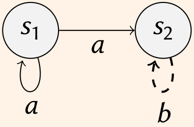
Next : GfG Automata

Good-for-Games Automata

Non-determinism resolved using history dependent strategy [Henzinger, Pitermen’06]

Good-for-Games Automata

Non-determinism resolved using history dependent strategy [Henzinger, Pitermen'06]



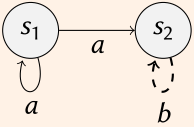
Has to go s_2 after some a^k

$a^{k+1}b^\omega$ is not accepted

Non-example

Good-for-Games Automata

Non-determinism resolved using history dependent strategy [Henzinger, Pitermen'06]



Non-example

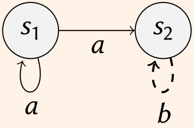
Has to go s_2 after some a^k

$a^{k+1}b^\omega$ is not accepted

Application in reactive synthesis

Good-for-Games Automata

Non-determinism resolved using history dependent strategy [Henzinger, Pitermen'06]



Has to go s_2 after some a^k

$a^{k+1}b^{\omega}$ is not accepted

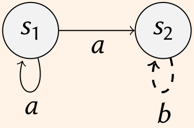
Non-example

Application in reactive synthesis

Good-for-Games co-Buchi automata can be minimized in PTIME [Abu Radi, Kupferman'21]

Good-for-Games Automata

Non-determinism resolved using history dependent strategy [Henzinger, Pitermen'06]



Non-example

Has to go s_2 after some a^k

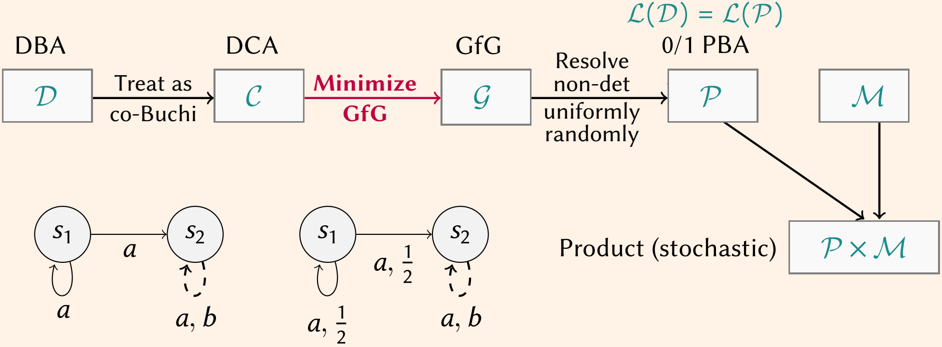
$a^{k+1}b^{\omega}$ is not accepted

Application in reactive synthesis

Good-for-Games co-Buchi automata can be minimized in PTIME [Abu Radi, Kupferman'21]

Next : Our Algorithm on DBA

Our algorithm on DBA



$\mathcal{L}(\mathcal{D}) = \mathcal{L}(\mathcal{P})$

DBA $\mathcal{D}$ → Treat as co-Buchi → DCA $\mathcal{C}$ → **Minimize GfG** → GfG $\mathcal{G}$ → Resolve non-det uniformly randomly → 0/1 PBA $\mathcal{P}$

$\mathcal{M}$

Product (stochastic) $\mathcal{P} \times \mathcal{M}$

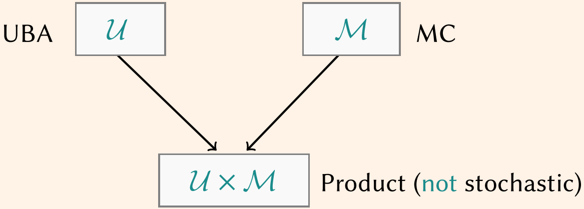
0/1 Probabilistic Buchi automata : Every word w is accepted with probability 0 or 1.

Algorithm for UBA from [BKKKMW’16]

$$\begin{tikzcd}
\text{UBA}\ \boxed{\mathcal{U}} \arrow[dr] & & \boxed{\mathcal{M}}\ \text{MC} \arrow[dl] \\
& \boxed{\mathcal{U} \times \mathcal{M}}\ \text{Product (not stochastic)} &
\end{tikzcd}$$

Standard linear equation system + Constraints for “special” SCCs

Algorithm for UBA from [BKKKMW'16]



UBA $\mathcal{U}$

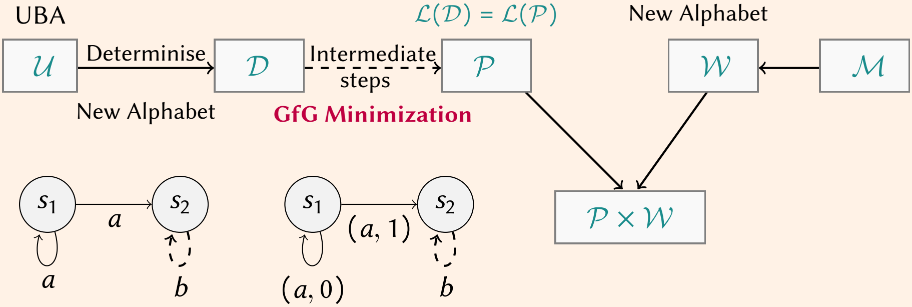
$\mathcal{M}$ MC

$\mathcal{U} \times \mathcal{M}$ Product (not stochastic)

Standard linear equation system + Constraints for "special" SCCs

Next : Our algorithm

Our Algorithm



UBA

$\mathcal{U}$ Determinise $\mathcal{D}$ Intermediate steps $\mathcal{P}$

New Alphabet

GfG Minimization

$\mathcal{L}(\mathcal{D}) = \mathcal{L}(\mathcal{P})$

New Alphabet

$\mathcal{W}$ $\mathcal{M}$

$\mathcal{P} \times \mathcal{W}$

Standard linear equation system + Constraints for special components

Exponential Gain

Theorem

There is a family of UBA $\{\mathcal{U}_n\}_n$, where the final GfG NCAs $\mathcal{G}_n$ are exponentially smaller.

Exponential Gain

Theorem

There is a family of UBA $\{\mathcal{U}_n\}_n$, where the final GfG NCAs $\mathcal{G}_n$ are exponentially smaller.

n	**PRISM UBA** [BKKKMW'16]		**Our Algorithm**		
	UBA size	Product size	GfG size	Product size	t_{tr}
3	7	100	5	100	0.052 s
4	15	200	6	120	0.021 s
5	31	400	7	140	0.077 s
6	63	800	8	160	0.059 s
7	127	1600	9	180	0.117 s
8	255	3200	10	200	0.102 s

Markov chain size : 20

More Benchmarks

Significant reduction in number of states

Complete UBAs				**Nearly-complete UBAs**			
k	**UBA**	**GfG**	t_{tr}	k	**UBA**	**GfG**	t_{tr}
5	193	96	0.121 s	5	193	94	0.122 s
6	449	192	0.126 s	6	449	190	0.126 s
7	1025	384	0.215 s	7	1025	382	0.251 s
8	2305	768	1.119 s	8	2305	766	1.070 s
9	5121	1536	8.160 s	9	5121	1534	8.179 s

Benchmarks from [BKKKMW'16]

Summary

Practice : More efficient algorithm for Markov chain model checking against UBA specifications

Conceptual : First synergy between UBA and GFG automata

Summary

Practice : More efficient algorithm for Markov chain model checking against UBA specifications

Conceptual : First synergy between UBA and GFG automata

Current and Future work

Summary

Practice : More efficient algorithm for Markov chain model checking against UBA specifications

Conceptual : First synergy between UBA and GFG automata

Current and Future work

- Faster implementation

Summary

Practice : More efficient algorithm for Markov chain model checking against UBA specifications

Conceptual : First synergy between UBA and GFG automata

Current and Future work

- Faster implementation
- For general NBA specifications, do exact model checking using stochastically resolvable NBA[1] or approximative verification using λ-PBA[2] instead of 0/1-PBA

[1] Resolving Non-determinism with Randomness [HPT] (To appear in MFCS'25)

[2] Resolving Non-determinism by Chance [PPSTTY](To appear in CONCUR'25)

Thank You