Probabilistic Floating-Time Transition System: A New Approach For State Space Reduction of Probabilistic Actors

Ali Jafari[1], Ehsan Khamespanah[1,2], Marjan Sirjani[1] and Holger Hermanns[3]

[1] Reykjavik University, School of Computer Science and CRESS
ali11@ru.is, ekhamespanah@yahoo.com, marjan@ru.is
[2] University of Tehran, School of ECE
[3] University of Saarland, School of Computer Science
hermanns@cs.uni-saarland.de

Probabilistic Timed Rebeca language was proposed in [1] as an extension of Timed Rebeca language. PTRebeca is an actor-based modeling language that supports modeling of timing, probabilistic and non-deterministic features of real-time systems. In [1], the semantics of PTRebeca is presented as a timed Markov decision process (TMDP) which can be regarded as a discrete-time semantics of a probabilistic timed automaton (PTA). We also developed a supporting tool for formal analysis of PTRebeca models which uses the back-end model checker PRISM to provide performance evaluation of models.

The state space generated from TMDP representation of PTRebeca models suffer from the state space explosion problem. In our semantics, the execution of statements of message servers are interleaved from various actors concurrently being executed in the real-time system. The semantics also includes a discrete global time and probabilistic information which make the state space explosion problem even more serious. In the semantics, the local time of all actors progresses in a lock step manner with the global time.

In [2], authors proposed floating time transition system (FTTS) as a solution of the state space explosion problem in model checking of Timed Rebeca models. In FTTS, actors proceed with their own rates with independent local clocks instead of synchronizing with the global time. In Timed Rebeca language, and consequently in PTRebeca language, actors can request a service from other actors by sending a message to them; each actor has a bag of messages which stores the received messages. The receiver actor takes a message from its bag and executes its corresponding message server to provide the requested service. In FTTS, by taking a transition, all statements of a message server of an actor are executed and the execution result is available in the next state. The execution of statements of the message server do not interleave with the execution of statements of other message servers from other actors. Since the message server may include timed statements, the local time of actors can have different values in a state. Relaxing the synchronization of progress of time among actors and the complete execution of a message server in a step avoid many interleaves and result in a significant state space reduction in FTTS.

In this work, we propose a probabilistic version of FTTS, called PFTTS, as a new semantics for PTRebeca language. Similar to FTTS, the proposed semantics reduces the state space significantly in comparison to TMDP semantics. Our intuitive understanding is that for a given PTRebeca model, its TMDP (aka PTTS) interpretation and its PFTTS interpretation are probabilistic trace-distribution equivalence, but we do not have a formal proof yet. Therefore, there is no LTL-without-Next formula which distinguishes two semantics in the sense that the min/max probabilities are the same for whatever formula is picked. As each action is not

Problem	Size	Using PFTTS			Using TMDP			Reduction	
		#states	#trans	time	#states	#trans	time	#states	#trans
Ticket Service	**1 customer**	13	17	< 1 sec	19	23	< 1 sec	32%	27%
	2 customers	315	417	< 1 sec	515	625	< 1 sec	39%	34%
	3 customers	3694	4949	2 sec	5962	7462	3 sec	39%	34%
	4 customers	23799	33617	11 sec	39528	51372	22 sec	40%	35%
	5 customers	92431	137041	72 sec	156919	212211	144 sec	41%	35%
Sensor Network	**1 sensor**	280	542	< 1 sec	343	605	< 1 sec	19%	19%
	2 sensor	3426	7119	1 sec	4254	7947	1 sec	20%	19%
	3 sensor	33959	79007	11 sec	40321	85369	15 sec	16%	8%
	4 sensor	211579	568603	83 sec	241011	598035	114 sec	12%	5%
TinyOS	**1 sensor**	2302	3100	< 1 sec	4507	5443	1 sec	49%	43%
	2 sensor	22630	38101	5 sec	48155	66765	13 sec	53%	43%

Table 1: Number of states and transitions, time consumption, and reduction ratio in model checking based on PFTTS and TMDP.

logged in the traces of TMDP, internal actions are not logged in the traces, LTL-without-Next properties are preserved. Therefore, model checking algorithms proposed for LTL properties can be applied to PFTTS instead of TMDP.

We also examined different case studies in different sizes to show that PFTTS generates fewer states and transitions in comparison to TMDP semantics. Apparently, the needed time for state space generation is decreased. To this goal, a toolset was developed to generate the state space of PFTTS and TMDP for a given PTRebeca model. The PTRebeca models of Ticket Service, Sensor Network, and TinyOS examples are accessible from [3]. Table 1 shows the experimental results.

Here, we briefly explain the first case study. There are a customer, a ticket server, and an agent. The customer sends a ticket request by sending a message to the agent. The agent forwards the request to the ticket server. The ticket server issues a ticket and replies to the agent request, and then the agent sends the ticket to the customer. The customer sends a new request after a specified amount of time. In order to have different sizes of the ticket service model, there are different number of customers in the model varying from one to five customers. We aim at investigating how much PFTTS can reduce the state space size. The experimental results show that the state space enlarges quickly in accordance with the number of rebecs (i.e. actors like customer, agent, and ticket server) in the model.

References

[1] Ali Jafari, Ehsan Khamespanah, Marjan Sirjani, and Holger Hermanns. Performance analysis of distributed and asynchronous systems using probabilistic timed actors. In *14th InternationalWorkshop on Automated Verification of Critical Systems (AVoCS)*, volume 70, 2014.

[2] Ehsan Khamespanah, Zeynab Sabahi Kaviani, Marjan Sirjani, Ramtin Khosravi, and Mohammad-Javad Izadi. Timed Rebeca Schedulability and Deadlock Freedom Analysis Using Bounded Floating-Time Transition System. In *Journal of Science of Computer Programming*, 2014.

[3] Rebeca. Rebeca homepage. http://www.rebeca-lang.org.