

Shepherding Hordes of Markov Chains

Benchmarks with new compiler flags.

Milan Češka¹, Nils Jansen², Sebastian Junges³, and Joost-Pieter Katoen³

¹ Brno University of Technology, Brno, Czech Republic

² Radboud University Nijmegen, Nijmegen, Netherlands

³ RWTH Aachen University, Aachen, Germany

The results provided in the paper were executed with an (small) extension to Storm as provided in commit

<https://github.com/moves-rwth/storm/commit/425eb4b1a9ede69edcd00eaa5c420b713dba1911>.

Storm contained for a range of commits a bug regarding the selected compiler flags. This bug **does not affect the correctness of any obtained results**, but does affect the performance. The bug was encountered and fixed with commit

<https://github.com/moves-rwth/storm/commit/1457c60e4b408a8ba0c561f25bf2fb096e1ec600>.

The fix was done very briefly before the deadline.

We realise that, among several configuration options, the change of compiler flags may have a major influence on the performance of Storm. Therefore, we have rerun the same experiments. We have not adapted any other code. For convenience, we provide the Tables 1 and 2 as presented in the paper. The new results are provided in Tables 3 and 4. We highlight the numbers that changed.

(Notice that we took the opportunity to replace estimates by precise numbers, as the performance allowed us to let the required algorithm to termination.)

Table 1. Benchmarks and timings for Approach 1–3.

Bench.	Range	K	D	Member size		Quotient size			Run time		Sched. Enum.
				Avg. S	Avg. T	S	A	T	1-by-1	All-in-1	
Pole	[3.35,3.82]	17	1327104	5623*	16727	6793	7897	22416	195h*	MO	27h*
Maze	[9.8,9800]	20	1048576	131*	204*	203	277	409	19h*	>15h	3h*
Herman	[1.86,2.44]	9	576	5287	6948	21313	102657	184096	213	306	305
DPM	[68,210]	9	32768	4463*	15056*	35154	66096	160146	4h*	>3h	3h*
BSN	[0,0.988]	10	1024	116	196	382	457	762	59	6	5

Table 2. Results for threshold synthesis via abstraction-refinement.

Inst	λ	#Below	# Subf below	#Above	# Subf above	Singles	#Iter	Time	Build	Check	Anal.	Speedup
Pole	3.37	697	176	1326407	2186	920	4723	951	183	333	280	738
	3.73	1307077	7854	20027	3279	1294	22265	5300	745	1514	2858	132
	3.76	1322181	3140	4923	1025	1022	8329	1437	247	459	461	488
	3.79	1326502	572	602	123	74	1389	137	33	66	34	5124
Maze	10	4	3	1048572	92	4	189	18	0	17	0	3000
	20	4247	2297	1044329	4637	3400	13867	1818	435	852	24	30
	30	18188	9934	1030388	18004	14010	55875	5575	498	1581	2258	10
	8000	1046285	846	2291	1125	969	3941	763	8	734	12	70
Herman	1.9	6	6	570	368	320	747	454	328	101	23	0.5
	1.71	0	0	576	258	184	515	327	233	74	19	0.7
DPM	80	160	141	32608	1292	356	2865	3436	777	2549	105	4
	70	6	6	32762	443	40	897	1229	213	979	35	12
	60	0	0	32768	104	6	207	475	70	394	11	32
BSN	.965	544	81	480	81	25	321	3	1	1	<1	2
	.985	994	41	30	8	5	97	1	<1	<1	<1	6

Table 3. Benchmarks and timings for Approach 1–3, new compiler flags.

Bench.	Range	K	D	Member size		Quotient size			Run time		Sched. Enum.
				Avg. S	Avg. T	S	A	T	1-by-1	All-in-1	
Pole	[3.35,3.82]	17	1327104	5689	16896	6793	7897	22416	36h*	MO	26360
Maze	[9.8,9800]	20	1048576	134	211	203	277	409	28000	>10h	2700
Herman	[1.86,2.44]	9	576	5287	6948	21313	102657	184096	55	72	246
DPM	[68,210]	9	32768	5572	18147	35154	66096	160146	2875	MO	2h*
BSN	[0,0.988]	10	1024	116	196	382	457	762	31	2	2

Table 4. Results for threshold synthesis via abstraction-refinement, new compiler flags.

Inst	λ	#Below	# Subf below	#Above	# Subf above	Singles	#Iter	Time	Build	Check	Anal.	Speedup
Pole	3.37	697	176	1326407	2186	920	4723	308	117	60	118	421
	3.73	1307077	7854	20027	3279	1294	22265	1668	576	317	396	77
	3.76	1322181	3140	4923	1025	1022	8329	584	187	114	197	222
	3.79	1326502	572	602	123	74	1389	58	23	10	23	2234
Maze	10	4	3	1048572	92	4	189	5	<1	3	<1	25920
	20	4247	2297	1044329	4637	3400	13867	114	21	43	29	246
	30	18188	9934	1030388	18004	14010	55875	608	80	127	270	46
	8000	1046285	846	2291	1125	969	3941	136	9	106	13	953
Herman	1.9	6	6	570	368	320	747	333	303	11	18	0.2
	1.71	0	0	576	258	184	515	232	206	8	17	0.3
DPM	80	160	141	32608	1292	356	2865	998	602	322	64	3
	70	6	6	32762	443	40	897	380	190	156	32	8
	60	0	0	32768	104	6	207	99	42	48	8	29
BSN	.965	544	81	480	81	25	321	2	<1	<1	<1	1
	.985	994	41	30	8	5	97	<1	<1	<1	<1	3