CODECHECK certificate 2025-018

https://doi.org/10.5281/zenodo.15630442

Item	Value
Title	Scalable control synthesis for stochastic systems via structural IMDP abstractions
Authors	Frederik Baymler Mathiesen (0000-0002-2243-0445) Sofie Haesaert (0000-0003-4749-4688) and Luca Laurenti (0000-0003-1190-6097)
Publication	https://doi.org/10.48550/arXiv.2411.11803
Publication repository	https://doi.org/10.4121/2c221b54-a20b-4659-99d2-af4a9a114b60.v2
Codecheckers	Niket Agrawal (0000-0002-3208-3440)
Date of check	2025-05-28
Summary	Tables 2, 3, and 4, as well as Figures 4 and 5 from the manuscript, were successfully reproduced by following the instructions provided in the README file. Instructions were only available for reproducing these specific tables and figures. To avoid the lengthy execution time required to run the full experiments, the pre-computed results provided in the repository were used to generate the tables and figures, as recommended in the README.
Codecheck repository	https://github.com/codecheckers/certificate-2025-008

Table 1: CODECHECK summary

Summary

- Tables 2, 3, and 4, as well as Figures 4 and 5 from the manuscript, were successfully reproduced by following the instructions provided in the README file. The README file is at the root of the code repository. Instructions were only available for reproducing these specific tables and figures.
- As mentioned in the README, the experiments are computationally intensive, requiring at least 100 GB of RAM and 12 CPU cores, and can take several weeks to finish executing. To avoid the lengthy execution time required to run the full experiments, the pre-computed results provided in the repository were used to generate the tables and figures, as recommended in the README.

Output	Comment
table_2_computation_performance.csv	manuscript Table 2
table_3_satisfaction_probability.csv	manuscript Table 3
table_4_car_parking.csv	manuscript Table 4
plot_4_car_parking.pdf	manuscript Figure 4
plot_5_convergence_analysis.pdf	manuscript Figure 5

Table 2: Summary of output files generated

CODECHECKER notes

The authors have made available all the necessary code and data required to reproduce figures in the manuscript. These resources can be accessed in the 4TU.ResearchData repository: https://data.4tu.nl/datasets/2c221b54-a20b-4659-99d2-af4a9a114b60/2

Setup

The results (tables and figures) were reproduced on a Windows machine using Windows Subsystem for Linux with Ubuntu 24.04.2. As specified in the README, Docker was required to run experiments. Docker version 28.2.2 was installed and used for this purpose.

Used pre-computed results to generate figures and tables

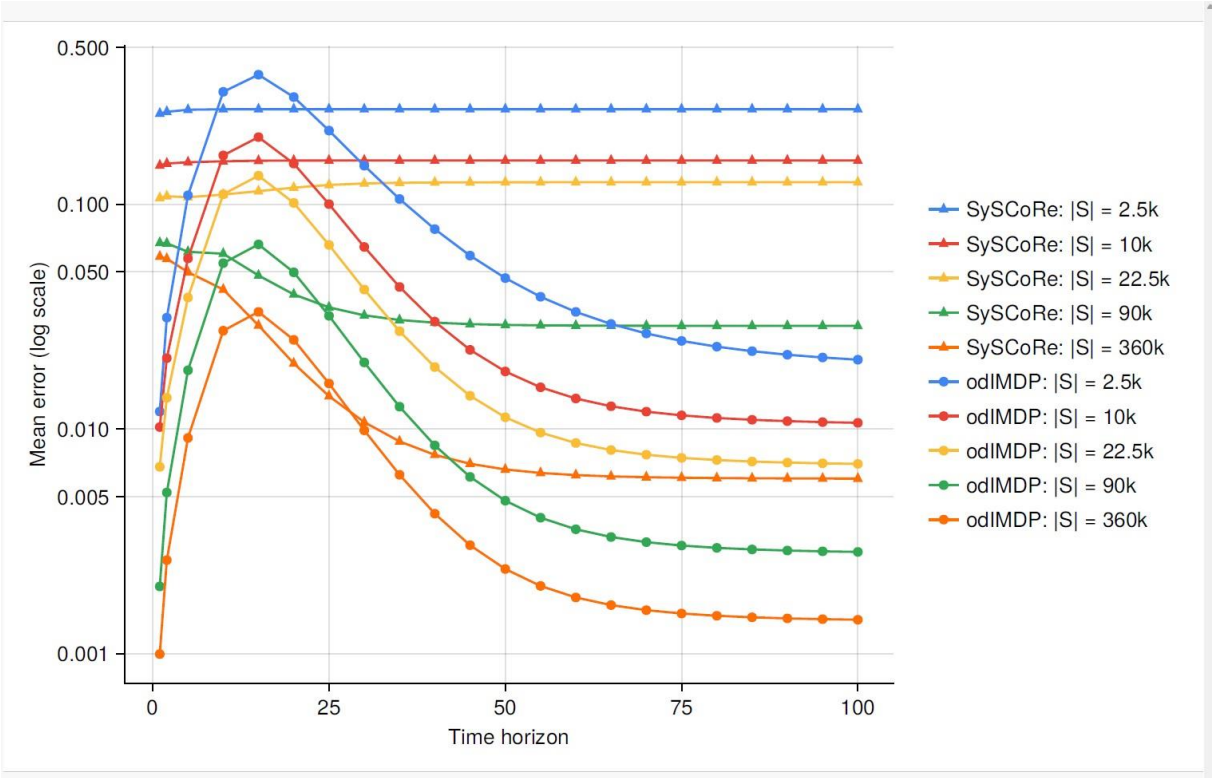
To reproduce the plots and figures presented in the manuscript, the instructions provided in the README file of the linked code repository were followed. The README outlines the steps required to reproduce Tables 2, 3, and 4, as well as Figures 4 and 5. By adhering to these instructions, the specified tables and figures could be reproduced.

As mentioned in the README, the experiments are computationally intensive, requiring at least 100 GB of RAM and 12 CPU cores, and can take several weeks to finish executing. To avoid the lengthy execution time required to run the full experiments, the pre-computed results provided in the repository were used to generate the tables and figures, as recommended in the README. The README contains instructions and commands to avoid running full experiments and instead use pre-computed results to generate the figures and the tables.

Manifest Files

plot_4_car_parking.pdf

Comment: Figure 4 in the manuscript



plot_5_convergence_analysis.pdf

Comment: Figure 5 in the manuscript

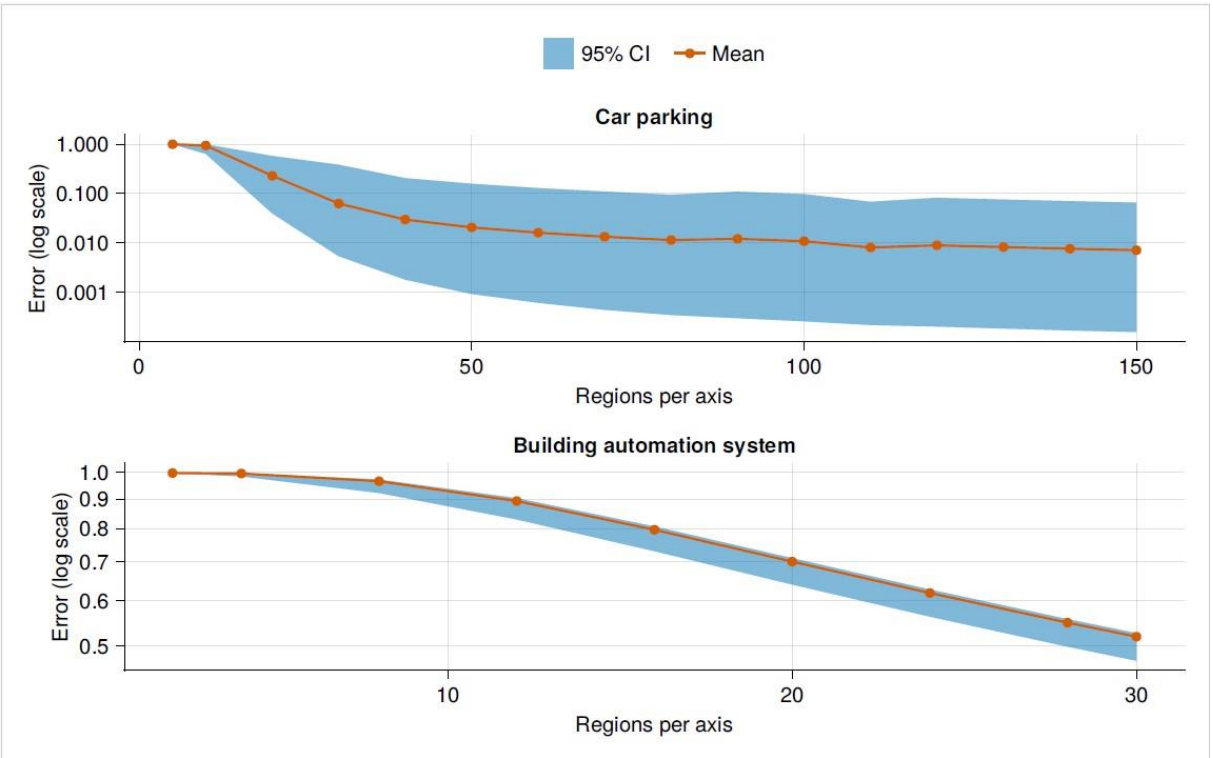


table_2_computation_performance.csv

Comment: Table 2 in the manuscript

name	odimdp_abstraction_time	odimdp_certification_time	odimdp_prob_mem	imdp_abstraction_time	imdp_certification_time	imdp_prob_mem	impact_abstraction_time	impact_certification_time	impact_prob_mem
dubins_car_gp_dkl	13.8163910 5	19.5621853 8	352.154 572	336.9399 322	31.3330167	14265. 75902	NaN	NaN	NaN
bas4d	0.0228307 48	0.2367081 23	2.89743 2	7.272964 141	0.285433 708	105.05 9676	17.564	0.318	96.04
linear_stochastically_switched	0.0328281 95	0.0447782 03	4.54865 2	3.391326 286	0.037905 204	41.030 696	NaN	NaN	NaN
robot_2d_reachability	0.6653229 97	0.1679514 65	21.6345 52	12.65847 498	0.129106 395	88.037 472	20.611	0.765	306.66 262
nndm_cartpole	236.15390 63	550.74677 54	611.442 908	42326.40 032	3.751213 305	1590.9 18472	NaN	NaN	NaN
linear6d	8.9594132 3	359.88650 01	237.271 508	NaN	NaN	NaN	NaN	NaN	NaN
linear7d	66.231402 13	13903.539 58	2329.96 8132	NaN	NaN	NaN	NaN	NaN	NaN
car_parking	0.1499414 17	0.1455921 69	19.2469 56	13.49703 406	0.255441 505	138.14 6144	19.57	0.846	304.93 82
van_der_pol	1.6577141 89	0.2570116 54	45.5149 68	27.25482 632	0.827186 841	353.56 0384	113.235	3.233	1093.4 004
robot_2d_reachavoid	14.258586 28	7.6411596 39	547.205 636	1136.720 091	6.739372 53	4143.7 89152	918.856	37.526	16387. 981

table_3_satisfaction_probability.csv

Comment: Table 3 in the manuscript

name	odimdp_mean_prob	odimdp_mean_error	imdp_mean_prob	imdp_mean_error	imdp_min_prob_diff	imdp_max_prob_diff	imdp_avg_prob_diff	impact_mean_prob	impact_mean_error	impact_min_prob_diff	impact_max_prob_diff	impact_avg_prob_diff
dubins_car_gp_dkl	0.361848828	0.34607029	0.216083367	0.504612218	0	0.83827025	0.14576546	NaN	NaN	NaN	NaN	NaN
bas4d	0.263381581	0.733617397	0.090041462	0.907552167	0.051011266	0.229691275	0.173340119	0.173642755	0.823726594	0.030380088	0.113085452	0.089738826
linear_stochastically_switched	0.411442647	0.282847092	0.365812433	0.360485831	1.90E-06	0.097890467	0.045630214	NaN	NaN	NaN	NaN	NaN
robot_2d_reachability	0.889201788	0.110761883	0.881356428	0.118628474	0.005992187	0.011864023	0.007861452	0.890211169	0.109775737	-0.005794382	0.002221171	-0.000993289
nndm_cartpole	0.003712224	0.763359494	5.24E-09	0.718378116	-1.22E-15	0.410079455	0.003712219	NaN	NaN	NaN	NaN	NaN
linear6d	0.958127812	0.041862759	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
linear7d	0.951709252	0.048279721	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
car_parking	0.268703695	0.388528444	0.212738476	0.531516591	0.004028623	0.142800555	0.055965219	0.213061863	0.518268229	0.004020587	0.142155255	0.055641832
van_der_pol	0.069156797	0.336660231	0.051437907	0.417839056	3.82E-110	0.052875211	0.017718889	NaN	NaN	NaN	NaN	NaN
robot_2d_reachavoid	0.979966909	0.019911571	0.979133952	0.020758723	0.000452377	0.002682901	0.000832957	0.979702864	0.020192979	-0.000105514	0.001837305	0.000264044

table_4_car_parking.csv

Comment: Table 4 in the manuscript

num_regions	syscore_mem_mb	syscore_eps	odimdp_mem_mb	odimdp_eps
22500	7.203978	0.125402384	937.85606	0.006994719
10000	3.203178	0.156759118	279.50737	0.010655293
360000	115.211178	0.006014842	59499.08377	0.001416572
90000	28.806378	0.028783168	7459.019402	0.002838414
2500	0.802378	0.264996946	35.54817963	0.020350222

Acknowledgements

This CODECHECK was done as part of the Reproducibility Check initiative led by TU Delft's Digital Competence Centre and 4TU.ResearchData.

Citing this document

Agrawal, Niket. (2025). CODECHECK Certificate 2025-018. CODECHECK. https://doi.org/10.5281/zenodo.15630442

About CODECHECK

This certificate confirms that the codecheckers could independently reproduce the results of a computational analysis given the data and code from a third party. A CODECHECK does not check whether the original computation analysis is correct. However, as all materials required for the reproduction are freely available by following the links in this document, the reader can then study for themselves the code and data.