

UC Projeto
3^o ano Licenciatura em Ciências da Computação
Construção de um ferramenta genérica de verificação SAT para
propriedades de segurança e animação de sistemas de transição de 1^o
ordem (FOTS)

Alef Keuffer
(A91683)

Alexandre Baldé
(A70373)

Bruno Machado
(A91680)

Pedro Pereira
(A88062)

Supervisor: Professor José Manuel Esgalhado Valença

19 de junho de 2022

Resumo

Neste relatório apresenta-se o trabalho realizado para a UC de Projeto, que consistiu no desenvolvimento, em Python¹, de uma ferramenta genérica de verificação de propriedades de " *First-Order Transition Systems*" através de uma biblioteca-interface para SMT-LIB.

Para a verificação de correção de propriedades de segurança e animação de FOTS, foram analisados os seguintes métodos: " *Bounded Model-Checking*", " *k-induction*", " *Property Directed Reachability*", " *Interpolant-Based Model-Checking*".

¹<https://www.python.org/>

Conteúdo

1	Introdução	3
1.1	Estrutura do Relatório	3
1.2	Problema em análise	3
1.3	Resolução e Estratégias adotadas	3
1.4	Agradecimentos	3
2	Estado de arte	4
3	Análise do trabalho	5
3.1	“ <i>k-induction</i> ” e “ <i>Bounded Model-Checking</i> ”	6
3.2	“ <i>Interpolant-Based Model-Checking</i> ”	7
3.3	“ <i>Property Directed Reachability</i> ”	8
4	Caso de Estudo	9
5	Conclusão	10
5.1	Comentários	10
5.2	Trabalho Futuro	10
A	Excertos de Código Utilizado no Projeto	11
B	Repositório <i>GitHub</i> com código fonte e documentação	12

Lista de Figuras

Capítulo 1

Introdução

Este relatório contém a descrição do projeto realizado pelos autores para a UC de Projeto da Licenciatura em Ciências da Computação, para o ano letivo de 2021/2022.

1.1 Estrutura do Relatório

A estrutura do relatório é a seguinte:

- No capítulo 2 faz-se uma análise do trabalho já existente na área, e das referências usadas para o projeto.
- No capítulo 3 explicam-se alguns aspetos mais técnicos e concretos da implementação, assim como decisões tomadas e alternativas consideradas.
- No capítulo 4 apresenta-se um case de estudo com um FOTS que servirá para apresentação das funcionalidades desenvolvidas.
- No capítulo 5 termina-se o relatório com as conclusões e o trabalho futuro.
- Nos anexos A e B, encontra-se informação relativa ao código Python desenvolvido, assim como o repositório GitHub que o contém.

1.2 Problema em análise

Para implementar este projeto, utilizou-se a biblioteca **PySMT**¹.

1.3 Resolução e Estratégias adotadas

1.4 Agradecimentos

¹Documentação disponível em <https://pysmt.readthedocs.io/en/latest/>

Capítulo 2

Estado de arte

“*k-induction*” e “*Bounded Model-Checking*”

Explorar: [BCCZ99] [SSS00].

“*Interpolant-Based Model-Checking*”

Explorar: [BLW21] [FR14] [BBW14].

“*Property Directed Reachability*”

Explorar: [BD19] [Bra11] [EMB11].

Capítulo 3

Análise do trabalho

3.1 “*k*-induction” e “*Bounded Model-Checking*”

3.2 ” *Interpolant-Based Model-Checking*

3.3 ”*Property Directed Reachability*”

Capítulo 4

Caso de Estudo

Capítulo 5

Conclusão

Conclui-se desta forma a apresentação do trabalho desenvolvido pelos autores para a UC *Projeto* no ano letivo 2021/2022.

5.1 Comentários

5.2 Trabalho Futuro

Apêndice A

Excertos de Código Utilizado no Projeto

Apêndice B

Repositório *GitHub* com código fonte e documentação

Link para código fonte: <https://github.com/Alef-Keuffer/FOTS-Prover>.

Link para documentação: <https://alef-keuffer.github.io/FOTS-Prover.docs/backend.html>.

Bibliografia

- [BBW14] Johannes Birgmeier, Aaron R. Bradley, and Georg Weissenbacher. Counterexample to induction-guided abstraction-refinement (ctigar). In Armin Biere and Roderick Bloem, editors, *Computer Aided Verification*, pages 831–848, Cham, 2014. Springer International Publishing.
- [BCCZ99] Armin Biere, Alessandro Cimatti, Edmund Clarke, and Yunshan Zhu. Symbolic model checking without bdds. In W. Rance Cleaveland, editor, *Tools and Algorithms for the Construction and Analysis of Systems*, pages 193–207, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.
- [BD19] Dirk Beyer and Matthias Dangl. Software verification with pdr: Implementation and empirical evaluation of the state of the art. 2019.
- [BLW21] Dirk Beyer, Nian-Ze Lee, and Philipp Wendler. Interpolation and sat-based model checking revisited: Adoption to software verification. 2021.
- [Bra11] Aaron R. Bradley. Sat-based model checking without unrolling. In Ranjit Jhala and David Schmidt, editors, *Verification, Model Checking, and Abstract Interpretation*, pages 70–87, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.
- [EMB11] Niklas Een, Alan Mishchenko, and Robert Brayton. Efficient implementation of property directed reachability. In *Proceedings of the International Conference on Formal Methods in Computer-Aided Design*, FMCAD ’11, page 125–134, Austin, Texas, 2011. FMCAD Inc.
- [FR14] Simone Fulvio Rollini. *Craig Interpolation and Proof Manipulation - Theory and Applications to Model Checking*. PhD thesis, 2014.
- [SSS00] Mary Sheeran, Satnam Singh, and Gunnar Stålmarck. Checking safety properties using induction and a sat-solver. In Warren A. Hunt and Steven D. Johnson, editors, *Formal Methods in Computer-Aided Design*, pages 127–144, Berlin, Heidelberg, 2000. Springer Berlin Heidelberg.