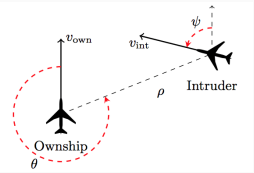
Caractériser des propriétés de confiance d'intelligence artificielle avec Why3

JFLA 2023

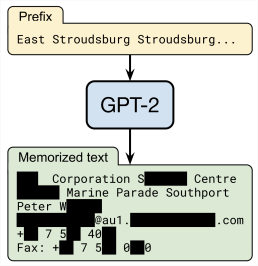
Julien Girard-Satabin (CEA LIST): julien.girard2@cea.fr
Michele Alberti (CEA LIST): michele.alberti@cea.fr
François Bobot (CEA LIST): francois.bobot@cea.fr
Zakaria Chihani (CEA LIST): zakaria.chihani@cea.fr

Des programmes difficiles à maîtriser...

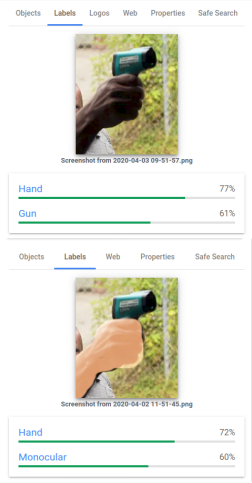
Robustesse locale?



Propriétés fonctionnelles?



Respect de la vie privée (en bonus avec les images [Car+23])?



Fairness?

... et beaucoup d'effort pour les analyser

- Marabou [Kat+19]
- Neurify
- ERAN [Sin+19; Mül+21]
- $\alpha - \beta$-Crown [Wan+21]
- Nnenum [Bak21]
- NNV (https://github.com/verivital/nnv)
- FaceLattice (https://arxiv.org/abs/2003.01226, https://github.com/verivital/FaceLattice)
- Facet-Vertex incidence (https://github.com/Shaddadi/Facet-Vertex-FFNN)
- Veritex (https://github.com/Shaddadi/veritex)
- Verinet and Venus (https://github.com/vas-group-imperial/VeriNet)

... et beaucoup d'effort pour les analyser

- ReluDiff (https://arxiv.org/abs/2001.03662, https://github.com/pauls658/ReluDiff-ICSE2020-Artifact)
- Peregrinn (https://arxiv.org/abs/2006.10864, https://github.com/rcpsl/PeregriNN)
- Oval (https://github.com/oval-group/oval-bab)
- Libra [Urb+19]
- MIPVerify [TXT19]
- Planet [Ehl17]
- Sherlock [Dut+17]
- ZoPE (https://arxiv.org/abs/2106.05325,https://github.com/sisl/NeuralPriorityOptimizer.jl)
- DNNV [SED21]

(crédit: Bill Wurtz)

Des niches écologique

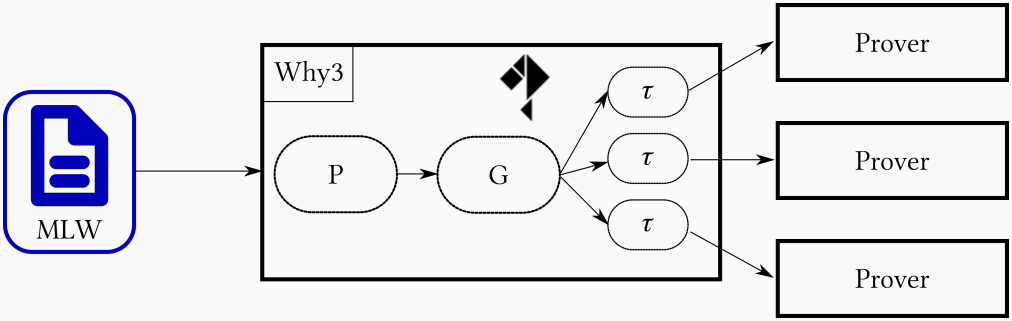
Spécialisations

1. par propriétés: robustesse locale, propriété fonctionnelles, *fairness*;
2. par techniques: Satisfaction Modulo Théorie (SMT), Programmation par Contrainte (CP), interprétation abstraite;

Ne pas se perdre dans la forêt

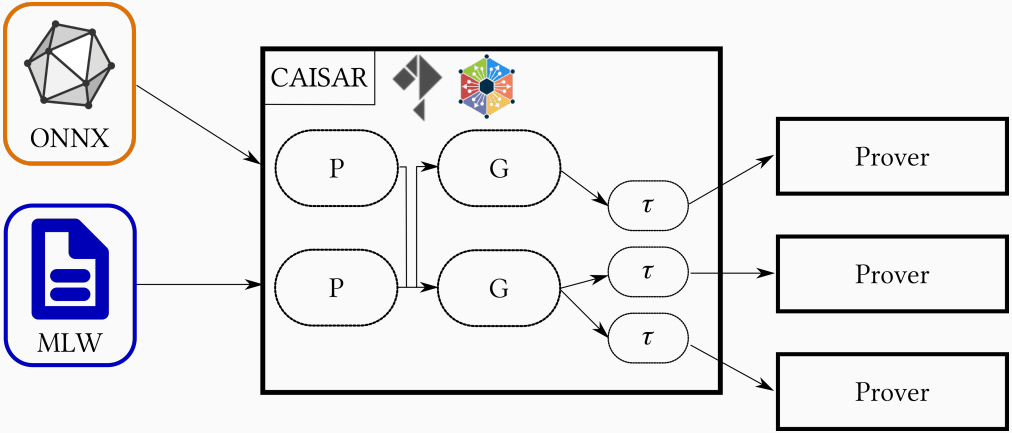
1. dialogue prouveur-humain et prouveur-prouveur complexe: différents langages, pas d'interopérabilité;
2. difficile de savoir quel prouveur utiliser au départ ?
3. modélisation de nouveaux problèmes difficile;

Why3 à la rescousse



Modélisation unifiée (WhyML), parser (P), générateur d'obligations de preuves (G), transformations (τ)

Construire sur du solide



Programme au format ONNX, gestion de sémantiques de prouveurs différentes

C’est l’heure de la démo:

1. décortiquage d’un fichier de spécification;
2. lancement sur cas d’usage ACAS de deux prouveurs;

Ce qu'on a ajouté

1. gestion des ensembles d'apprentissage, propriétés non "réellement" universelles;
2. support de six prouveurs dédiés: SMT; interprétation abstraite, test métamorphique;
3. traducteur du flot de contrôle du programme (réseaux de neurones, SVM, arbres) vers une représentation intermédiaire et du SMTLIB "classique";
4. génération automatique des signatures de fonction;
5. prédicats classiques de vérification d'IA;

Travaux en cours

1. sémantiques de preuves différentes (tests métamorphiques);
2. granularité dans les tactiques;
3. propagation des résultats de vérification d'un programme vers un autre;
4. spécifier (et prouver) les données?

Chantier: extension de l'interpréteur Why3 (*reduction engine*)

Travaux en cours

1. sémantiques de preuves différentes (tests métamorphiques);
2. granularité dans les tactiques;
3. propagation des résultats de vérification d'un programme vers un autre;
4. spécifier (et prouver) les données?

Chantier: extension de l'interpréteur Why3 (*reduction engine*)

```
theory MyImportantAIVerif
  use ieee_float.Float64
  use caisar.DatasetClassificationProps

  constant dataset = open_dataset "path/to/dataset"
  constant net = open_program "path/to/onnx"
  constant svm = open_program "path/to/svm"
  constant y = apply net dataset

  goal robustness_svm:
    let eps = (0.5:t) in
    robust svm y eps

  goal functional_prop:
    forall _x in dataset.
      _x[3] .≥ (0.5:t) ∧ _x[2] .≤ (0.0:t) →
      y[3] .≥ 0.5

  (...)

end
```

CAISAR

Site web: `https://caisar-platform.github.io/website/`
Logiciel libre (LGPLv2): `https://git.frama-c.com/pub/caisar`
Rapport technique: `https://hal.science/hal-03687211`
Offres: `https://caisar-platform.github.io/website/positions`

Stages, post-docs, CDD à pourvoir! julien.girard2@cea.fr

La maturation de CAISAR est partiellement financée par les projets PRISSMA et Confiance.ai, du Grand Défi IA de Confiance

References

[Bak21] Stanley Bak. “Nnenum: Verification of ReLU Neural Networks with Optimized Abstraction Refinement”. In: *NASA Formal Methods*. Ed. by Aaron Dutle, Mariano M. Moscato, Laura Titolo, César A. Muñoz, and Ivan Perez. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2021, pp. 19–36. ISBN: 978-3-030-76384-8. DOI: 10.1007/978-3-030-76384-8_2 (cit. on p. 3).

Bibliography ii

[Car+23] Nicholas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramèr, Borja Balle, Daphne Ippolito, and Eric Wallace. *Extracting Training Data from Diffusion Models*. 2023. DOI: 10.48550/ARXIV.2301.13188. URL: https://arxiv.org/abs/2301.13188 (cit. on p. 2).

[Dut+17] Souradeep Dutta, Susmit Jha, Sriram Sanakaranarayanan, and Ashish Tiwari. "Output Range Analysis for Deep Neural Networks". In: *arXiv:1709.09130 [cs, stat]* (Sept. 2017). arXiv: 1709.09130 [cs, stat] (cit. on p. 4).

Bibliography iii

[Ehl17] Ruediger Ehlers. “Formal Verification of Piece-Wise Linear Feed-Forward Neural Networks”. In: *arXiv:1705.01320 [cs]* (May 2017). arXiv: 1705.01320 [cs] (cit. on p. 4).

Bibliography iv

[Kat+19] Guy Katz, Derek A. Huang, Duligur Ibeling, Kyle Julian, Christopher Lazarus, Rachel Lim, Parth Shah, Shantanu Thakoor, Haoze Wu, Aleksandar Zeljić, David L. Dill, Mykel J. Kochenderfer, and Clark Barrett. "The Marabou Framework for Verification and Analysis of Deep Neural Networks". en. In: *Computer Aided Verification*. Ed. by Isil Dillig and Serdar Tasiran. Vol. 11561. Cham: Springer International Publishing, 2019, pp. 443–452. ISBN: 978-3-030-25539-8 978-3-030-25540-4. (Visited on 07/18/2019) (cit. on p. 3).

Bibliography v

[Mül+21] Christoph Müller, François Serre, Gagandeep Singh, Markus Püschel, and Martin Vechev. “Scaling Polyhedral Neural Network Verification on GPUs”. In: *Proceedings of Machine Learning and Systems* 3 (2021) (cit. on p. 3).

[SED21] David Shriver, Sebastian Elbaum, and Matthew B. Dwyer. “DNNV: A Framework for Deep Neural Network Verification”. In: *Computer Aided Verification*. Ed. by Alexandra Silva and K. Rustan M. Leino. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2021, pp. 137–150. ISBN: 978-3-030-81685-8. DOI: 10.1007/978-3-030-81685-8_6 (cit. on p. 4).

Bibliography vi

[Sin+19] Gagandeep Singh, Rupanshu Ganvir, Markus Püschel, and Martin Vechev. “Beyond the Single Neuron Convex Barrier for Neural Network Certification”. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d\textquotesingle Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 15098–15109. URL: http://papers.nips.cc/paper/9646-beyond-the-single-neuron-convex-barrier-for-neural-network-certification.pdf (visited on 07/27/2020) (cit. on p. 3).

Bibliography vii

[TXT19] Vincent Tjeng, Kai Xiao, and Russ Tedrake. “Evaluating Robustness of Neural Networks with Mixed Integer Programming”. In: International Conference on Learning Representations (ICLR). 2019. URL: https://openreview.net/pdf?id=HyGIdiRqtm (visited on 06/19/2019) (cit. on p. 4).

[Urb+19] Caterina Urban, Maria Christakis, Valentin Wüstholz, and Fuyuan Zhang. “Perfectly Parallel Fairness Certification of Neural Networks”. In: *arXiv:1912.02499 [cs]* (Dec. 2019). arXiv: 1912.02499 [cs] (cit. on p. 4).

Bibliography viii

[Wan+21] Shiqi Wang, Huan Zhang, Kaidi Xu, Xue Lin, Suman Jana, Cho-Jui Hsieh, and J. Zico Kolter. *Beta-CROWN: Efficient Bound Propagation with Per-neuron Split Constraints for Complete and Incomplete Neural Network Robustness Verification*. Oct. 31, 2021. arXiv: 2103.06624 [cs, stat]. URL: http://arxiv.org/abs/2103.06624 (visited on 03/04/2022) (cit. on p. 3).