

# Multiple provers, multiple answers: formulating and solving verification problems for diverse artificial intelligence components using CAISAR

Workshop on Safe and Robust Machine Learning, LAIV

---

**Julien Girard-Satabin (CEA LIST): [julien.girard2@cea.fr](mailto:julien.girard2@cea.fr)**

Michele Alberti (CEA LIST): [michele.alberti@cea.fr](mailto:michele.alberti@cea.fr)

François Bobot (CEA LIST): [francois.bobot@cea.fr](mailto:francois.bobot@cea.fr)

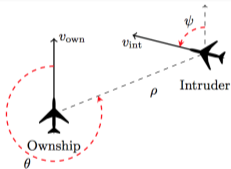
Zakaria Chihani (CEA LIST): [zakaria.chihani@cea.fr](mailto:zakaria.chihani@cea.fr)



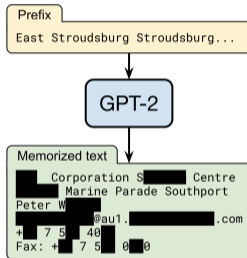
# AI are difficult to harness...



Local robustness?



Functional properties?



Privacy preservation [Car+23]?

Objects Labels Logos Web Properties Safe Search

Screenshot from 2020-04-03 09-51-57.png

Hand	77%
Gun	61%

Objects Labels Web Properties Safe Search

Screenshot from 2020-04-02 11-51-45.png

Hand	72%
Monocular	60%

Fairness?

## ... and a lot of effort is poured into their safety

- 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>)

## ... and a lot of effort is poured into their safety

- 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/PeregrinNN>)
- 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)

## Specializing

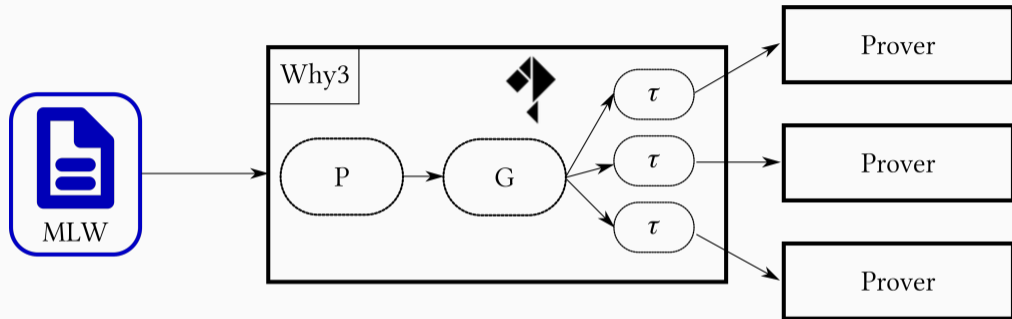
1. in properties: local robustness, functional properties, *fairness*
2. in techniques: Satisfaction Modulo Theory, Constraint Programming, Abstract Interpretation, Linear Programming...

# A dense forest

1. human-to-prover discussion is prone to error: different input languages with limited interoperability and ambiguous implementations (disjunctive formulaes or negation of property under the hood for Marabou, PyRAT negates property depending on its form...)
2. prover-to-prover discussion does not exist yet
3. just which prover will solve my query anyway?
4. modelling new problems for different provers is time-consuming



# We solved this problem before with Why3!



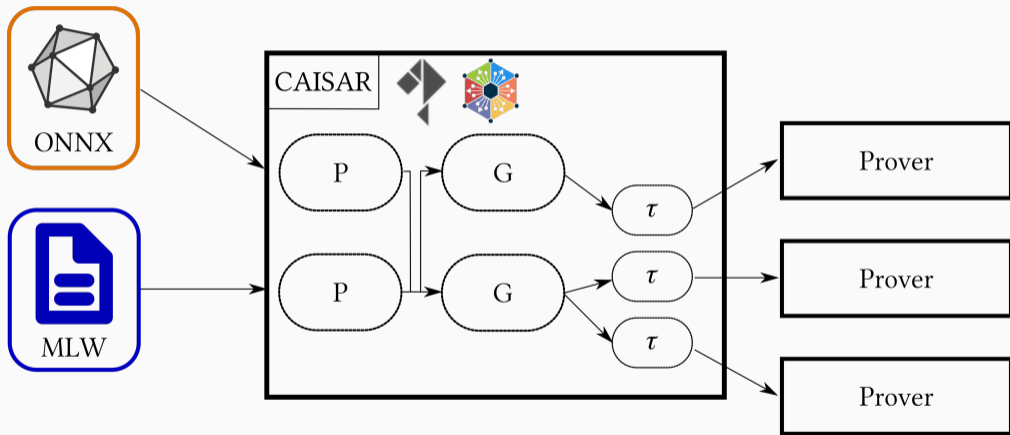
Unified modelling (WhyML), parser (P), proof obligation generator (G), transformations ( $\tau$ )

A software verification platform [FP13]

Website: <https://why3.lri.fr/>



# Building on sound foundations



Programme component in ONNX format, multiple prover semantics supported

Demo time:

1. let us read a CAISAR specification file
2. let us launch CAISAR on a simple use case with different provers and compare the results

# Our work so far

1. quantified datasets: we can check properties on a collection of discrete samples
2. eight dedicated provers are supported so far
3. translation of ONNX programs to "classical" SMTLIB2 to leverage existing "classical" provers
4. some classical predicates for AI formal verification are built-in

# Our work in progress

1. different verification  
semantic (metamorphic  
testing)
2. proof tactic granularity
3. prover combination
4. working (soon) on  
quantified networks
5. reworking the specification  
language to provide a  
certain degree of  
computation

# Our work in progress

1. different verification semantic (metamorphic testing)
2. proof tactic granularity
3. prover combination
4. working (soon) on quantified networks
5. reworking the specification language to provide a certain degree of computation

```
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
```



C A I S A R

Website: <https://caisar-platform.com/>

Free Software (LGPLv2): <https://git.frama-c.com/pub/caisar>

Papers: <https://hal.science/hal-03687211>

Job offers: <https://caisar-platform.github.io/website/positions>

Internships, post-docs, fixed-term contract available! [julien.girard2@cea.fr](mailto:julien.girard2@cea.fr)



CAISAR development was partially funded by PRISSMA and Confiance.ai projects

## 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](https://doi.org/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 Sehwal, 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).
- [FP13] Jean-Christophe Filliâtre and Andrei Paskevich. “Why3 - Where Programs Meet Provers”. In: *Programming Languages and Systems*. Ed. by Matthias Felleisen and Philippa Gardner. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2013, pp. 125–128. ISBN: 978-3-642-37036-6. DOI: 10.1007/978-3-642-37036-6\_8 (cit. on p. 8).

## 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](https://doi.org/10.1007/978-3-030-81685-8_6) (cit. on p. 4).

- [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. 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=HyGIIdiRqtm> (visited on 06/19/2019) (cit. on p. 4).
- [Urb+19] Caterina Urban, Maria Christakis, Valentin Wüstholtz, 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).