RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

ONERA
THE FRENCH AEROSPACE LAB

ACETONE : Un outil open source de génération de code C prédictible à partir de modèles de réseaux de neurones

Presenters: Claire Pagetti, Yanis Ait Aissa, Benjamin Lesage (ONERA)
Contributors: Iryna de Albuquerque Silva (CEA), Thomas Carle (IRIT), Adrien Gauffriau (Airbus), Nicolas Valot (Airbus)

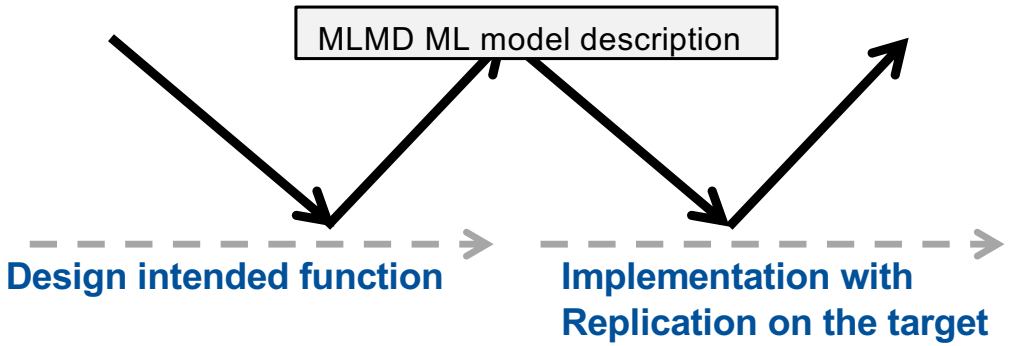
Focus Technique ANITI 2025

Outline

- **ML certification**
- ACETONE
- Integration ACETONE in AIDGE

Capture applicable regulatory requirements

- Documents
  - EASA concept paper
  - EUROCAE/SAE ED 324-ARP 6983
- Focus: (supervised) off-line trained ML models
- Outcome
  - “W Shape” Development Lifecycle
  - Semantic preservation of the trained ML model in the implementation
  - SW Implementation compliant with DO 178



ANITI – CertifAI chair

Iryna de Albuquerque Silva (ANITI PhD 2021 – 2024) "Certifiable and efficient implementation of neural networks on embedded safety-critical real-time systems."

- Developed an approach that enables the implementation of the inference model in compliance with avionics requirements
  - precise description of the inference model's operations (math level)
  - semantics preservation and traceability between the inference model description and the final executable
  - timing predictability of the implementation (WCET estimation with OTAWA)
  - efficient usage of the available resources
- **ACETONE "Avionics C code generator for Neural Networks"**

https://hal.science/tel-05018032v1

Outline

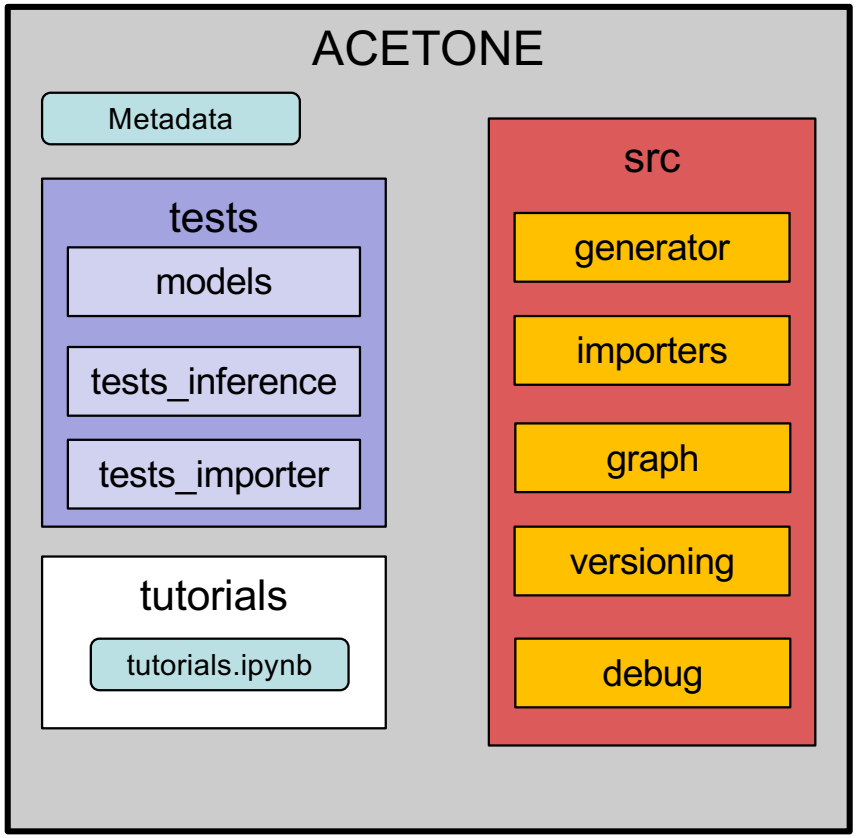
- ML certification
- **ACETONE**
- Integration ACETONE in AIDGE

ACETONE repository

https://github.com/onera/acetone

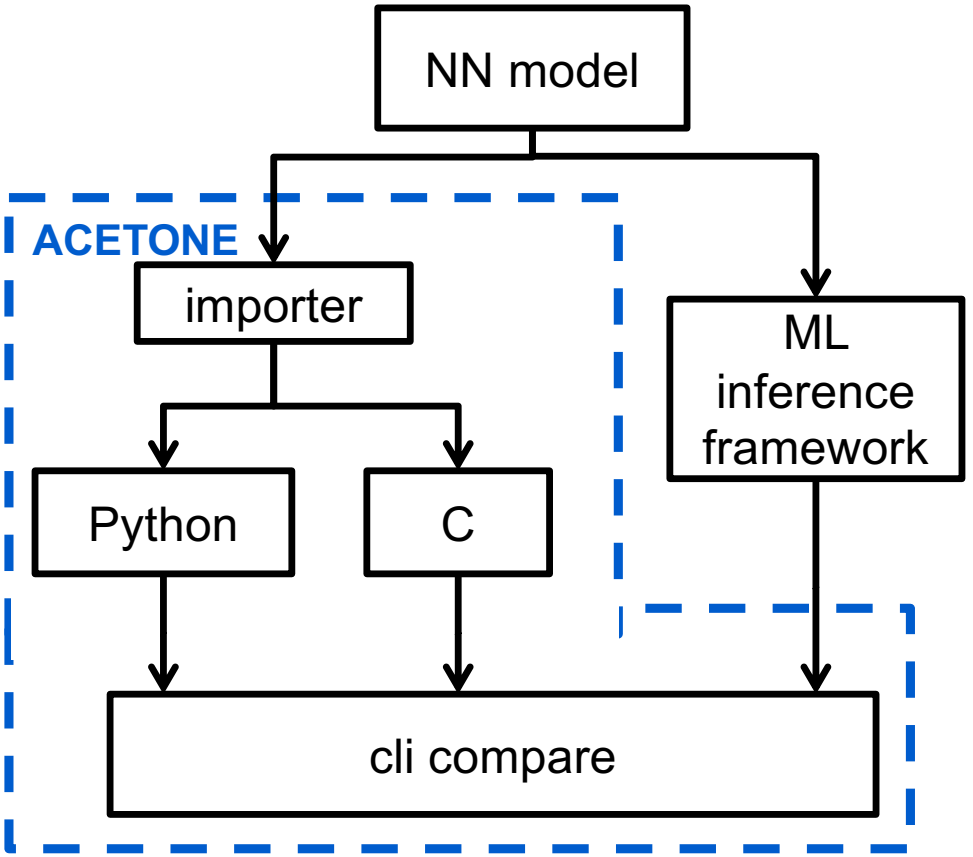
Python package

- Source code:
  - Layers description and code generator
  - Formats importers
  - Graph interpreter
  - Version handler
  - Debug tools
- Unit tests:
  - Model bank
  - Unit tests
- Tutorials



Tutorial 1

- Importers
  - *NNet*: initial format for fully connected NN, used by Reluplex git
  - *ONNX*: Widely used in industry
  - *H5*: Keras format
  - *JSON*: home made format (inspired from Keras)
- Test with ACETONE
  - ACETONE Python's reference execution
  - Training framework (e.g. ORT, Keras …)
- **Objective:**
  - **Check semantic preservation**

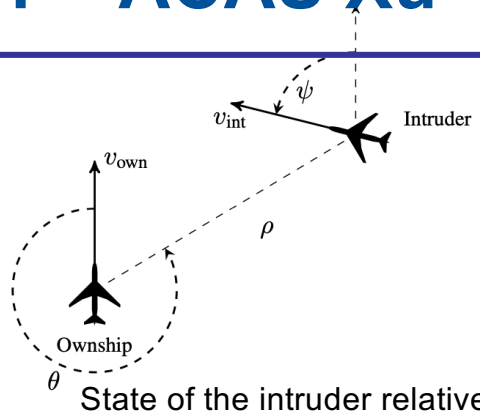


Use case for tutorial 1 – ACAS Xu

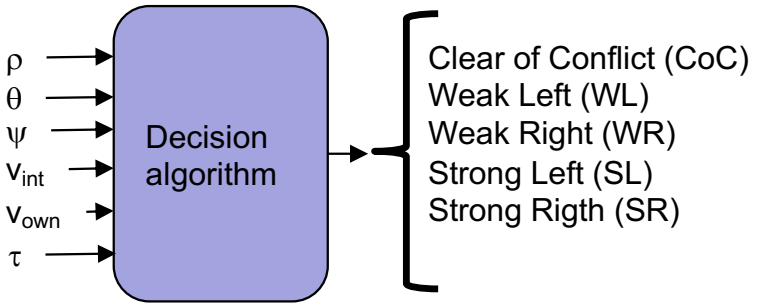
Avoidance Collision System for vertical and horizontal cooperative and non-cooperative avoidance (Multi-Intruders)

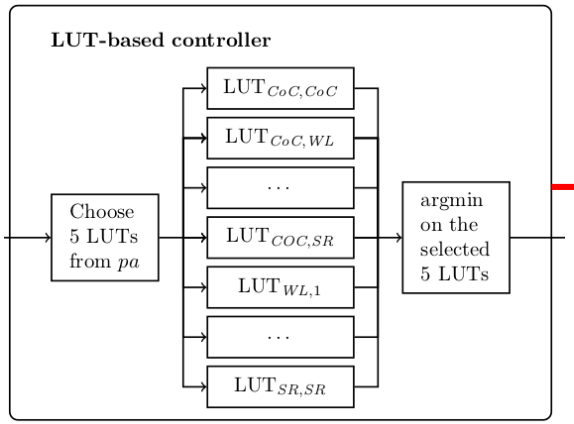
Intended function "any intruder should not enter in the ownship envelope"

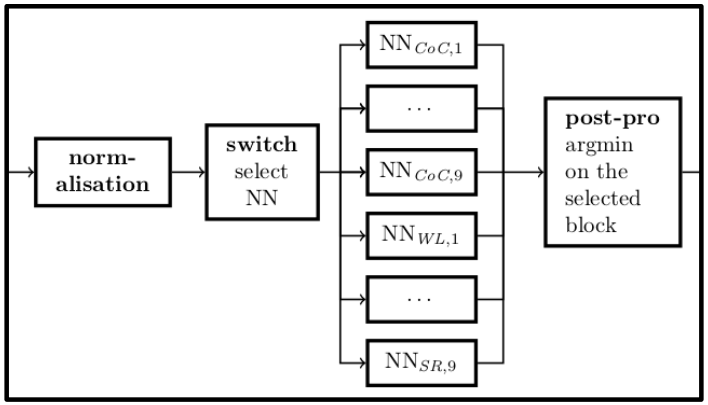
- Interest: Gain in memory footprint (from 4Go to 3Mo)



State of the intruder relative to ownship



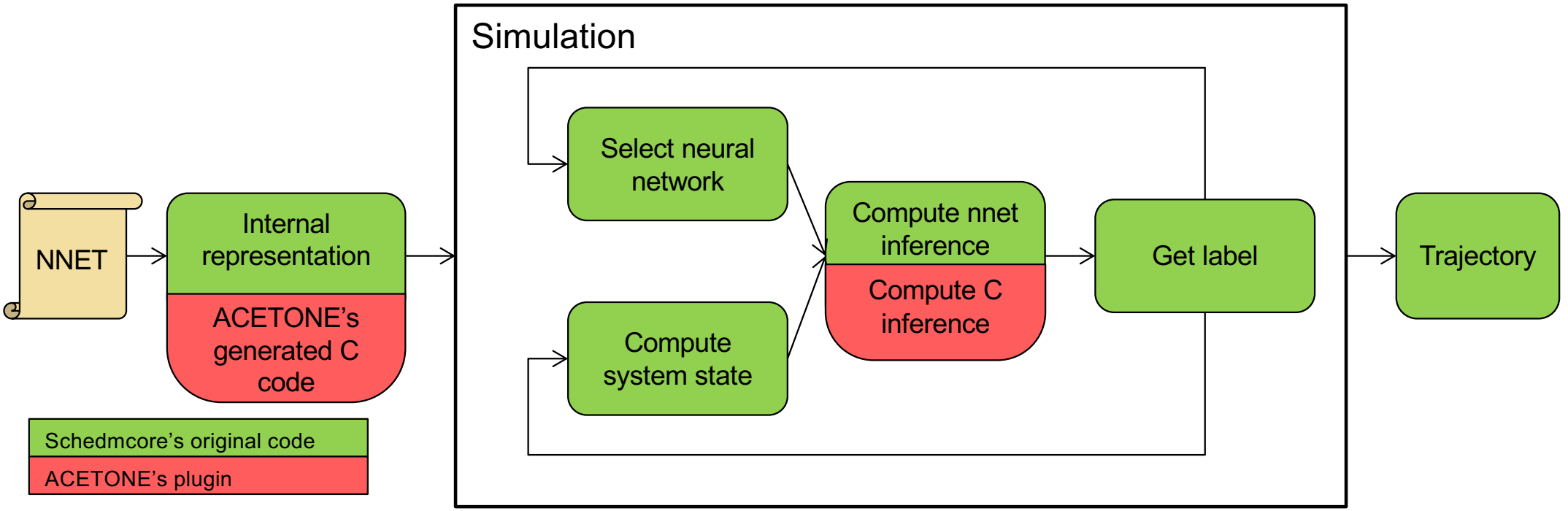




45 NN (fully connected NN with 6 layers of 50 neurons each)

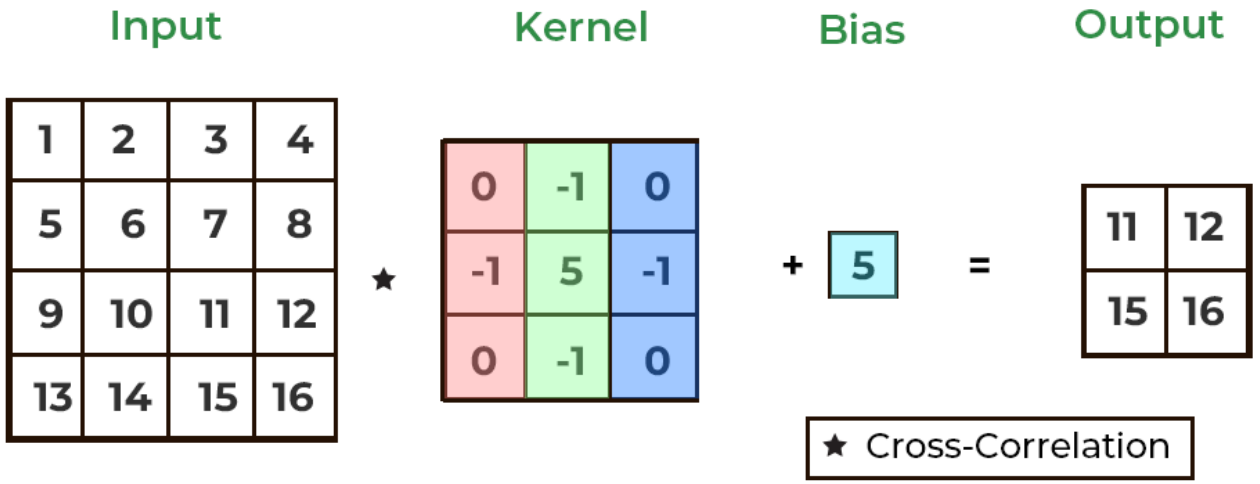
Tutorial 1: simulation

- Python Simulator developed by Arthur Clavière (PhD Collins)
  https://svn.onera.fr/schedmcore/branches/ACAS_CaseStudy/
- Extended to link with C code generated by ACETONE



Tutorial 2: convolution implementation(s)

- Reminder on convolution



- Naïve implementation 6 loops
- Often more efficient: matrix based multiplication (GEMM)
- **Objective: optimisation (better use of resources)**

Figure excerpt from https://www.geeksforgeeks.org/apply-a-2d-convolution-operation-in-pytorch/

Tutorials 3&4

ACETONE

- Tutorial 3: debug mode
  - Automatic adding of outputs within the neural network graph
- Tutorial 4: advanced use of ACETONE
  - Adding a new variant of convolution
  - Use of debug mode to check whether the new implementation preserves the semantic

Outline

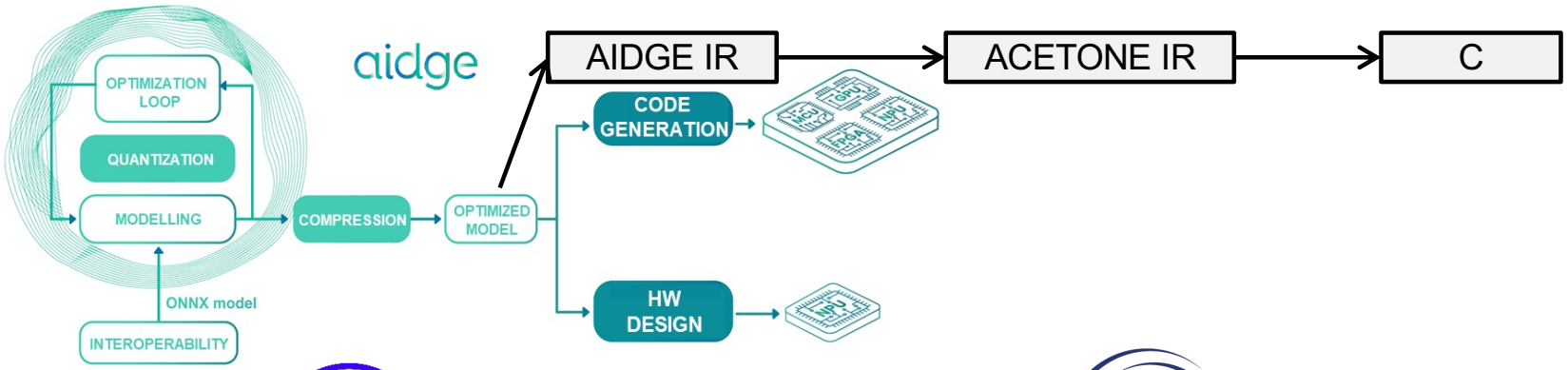
- ML certification
- ACETONE
- **Integration ACETONE in AIDGE**

AIDGE – ACETONE as a C back-end

AIDGE Embedded AI Deep Learning Platform

- Developed by CEA
- Code generation back-end: OpenMP, OpenCL, Cuda, cuDNN and TensorRT

https://eclipse.dev/aidge/



ONNX

PyTorch

DEEPGREEN

FRANCE 2030

anr agence nationale de la recherche

cea

Question?

Thank you for your attention