Robustness Verification: Neural Network's Surrogate

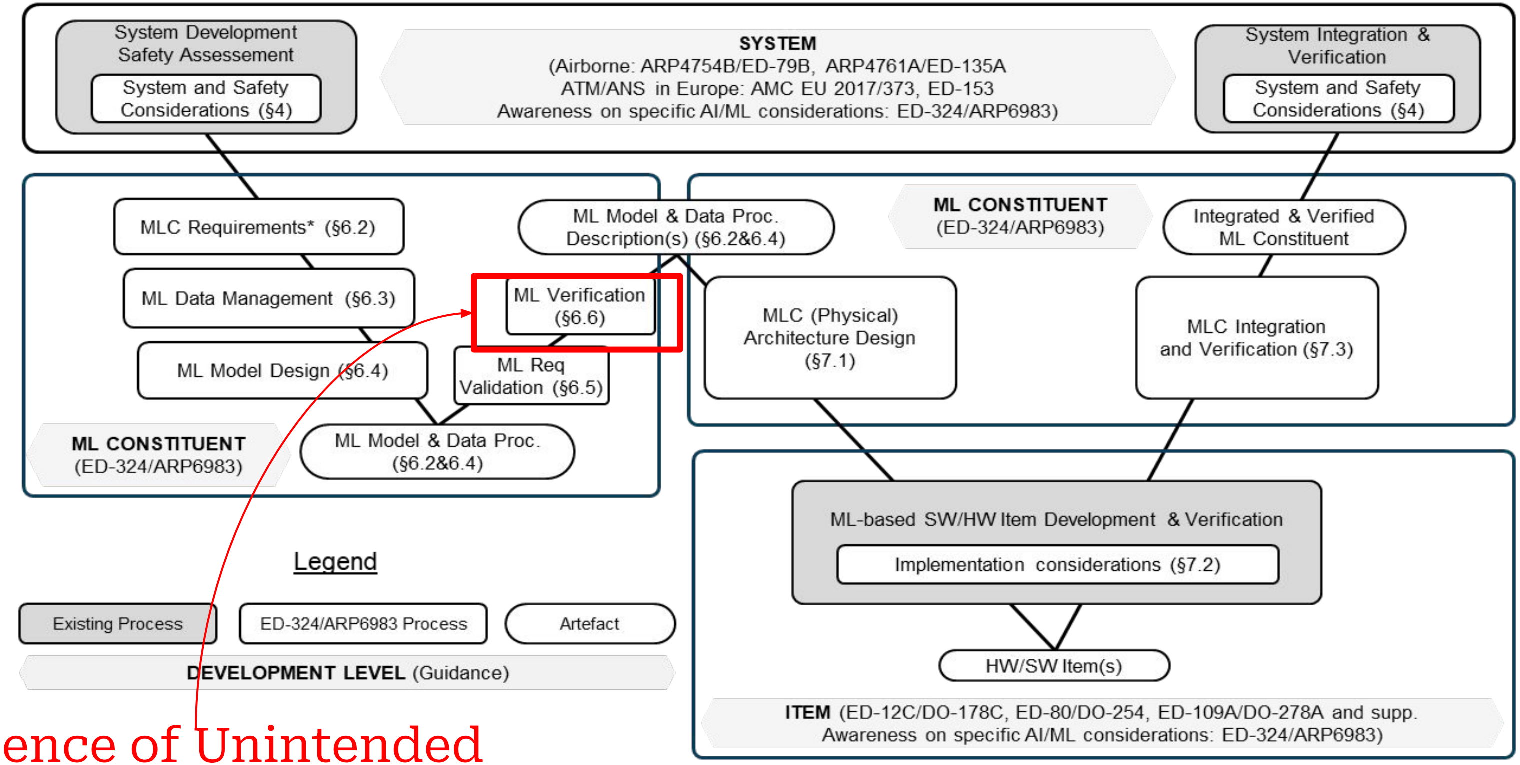
Melanie Ducoffe

ANITI DAYS
November 2024

ANITI DEEL AIR Airbus AI Research

Robustness during ML development (ARP6983)

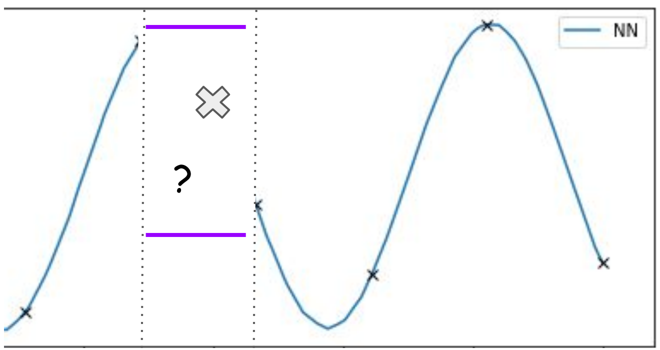
AIRBUS



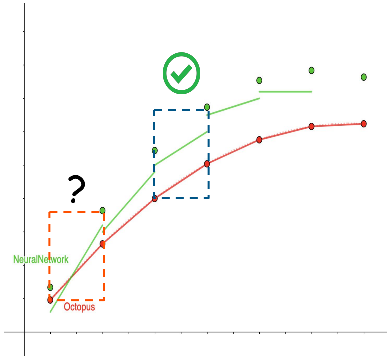
Absence of Unintended Functionality

ForMuLA: Formal Methods Use for Learning Assurance – EASA & Collins Aerospace partnership

Property Requirement for Surrogate Models



$$\max_{x\in\Omega} | f(x) - f(x_0)|$$



$$\min_{x\in\Omega} f(x) - f(x_0) \geq 0$$

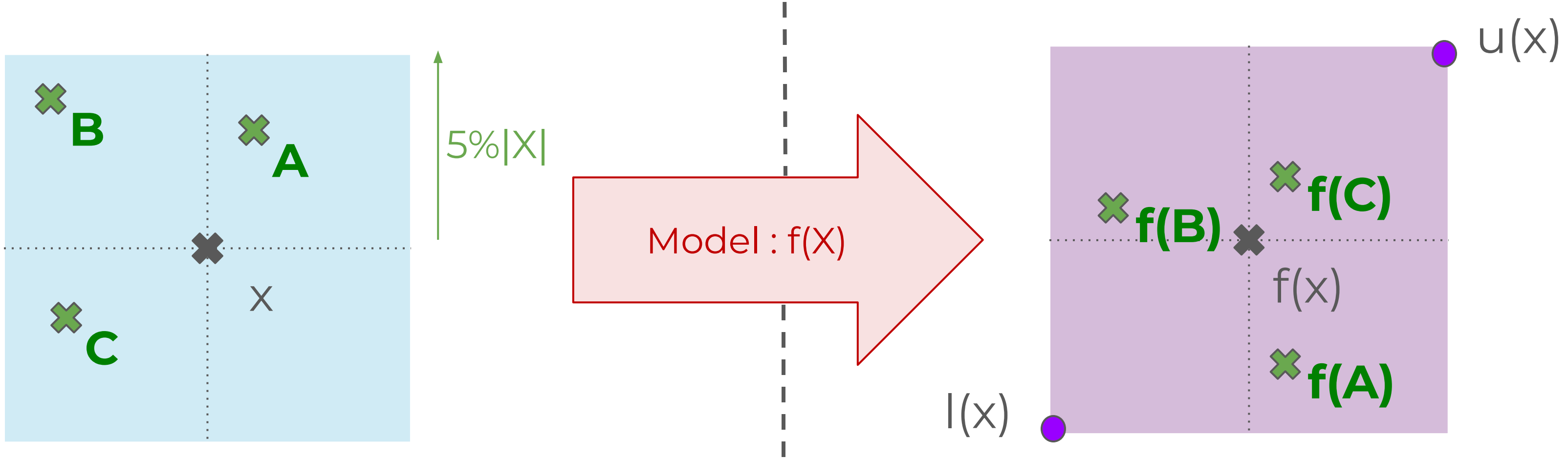
Partial Input Monotony

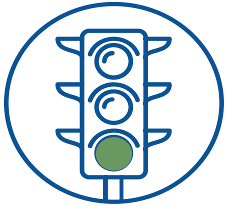
$$\begin{array}{ccccc} x_1 & & x_2 & f(x_1) & f(x_2) \\ \begin{pmatrix} \text{speed} \\ \text{weight} \\ \text{dry runway} \end{pmatrix} & \begin{matrix} = \\ = \\ < \end{matrix} & \begin{pmatrix} \text{speed} \\ \text{weight} \\ \text{wet runway} \end{pmatrix} & \Longrightarrow & \text{BDE}_1 < \text{BDE}_2 \end{array}$$

Introducing Verification (on Local Stability)

INPUT DOMAIN Ω

OUTPUT DOMAIN





The property is **verified** !

$$\max_{Z \in \Omega} g(Z;X) \leq 0$$

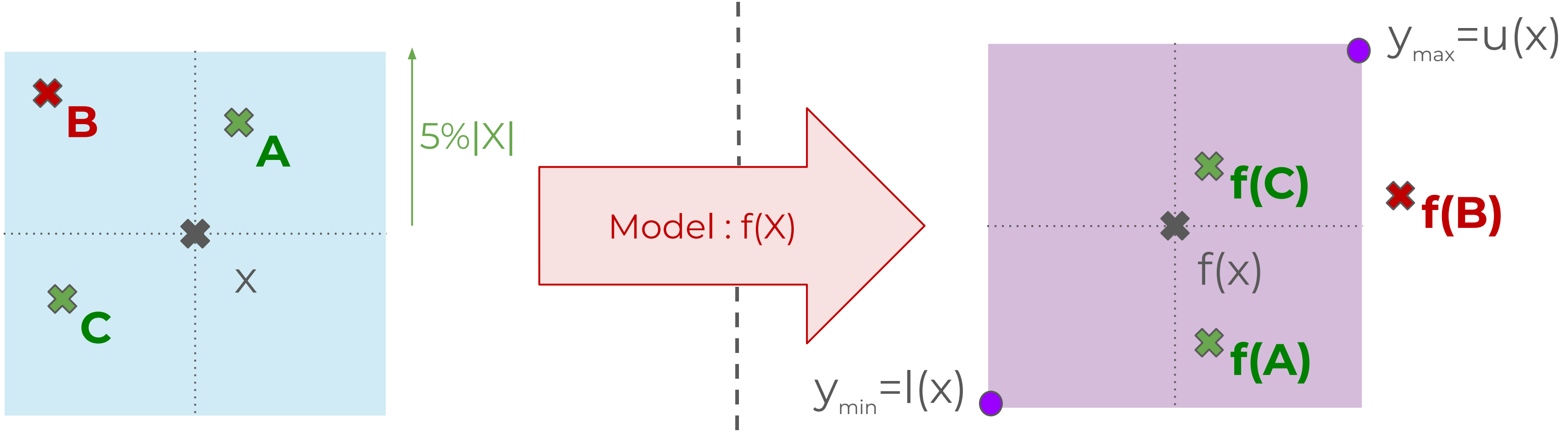
$$g(Z;X) = \max_i \max(f_i(Z) - u_i(X), l_i(X) - f_i(Z))$$

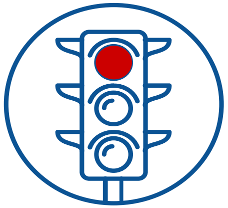
Introducing Verification (on Local Stability)

AIRBUS

INPUT DOMAIN Ω

OUTPUT DOMAIN





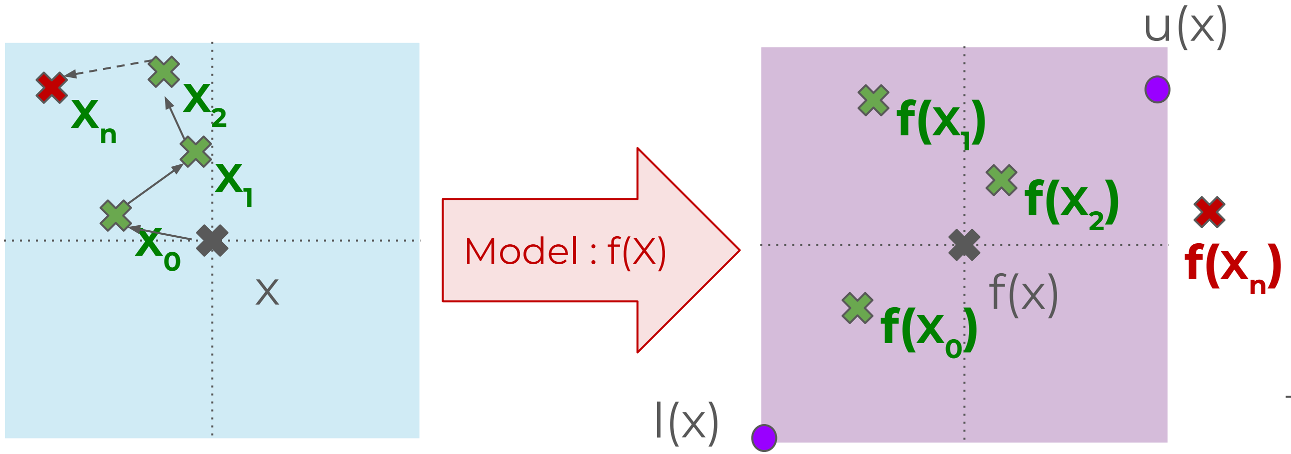
The property is **violated**

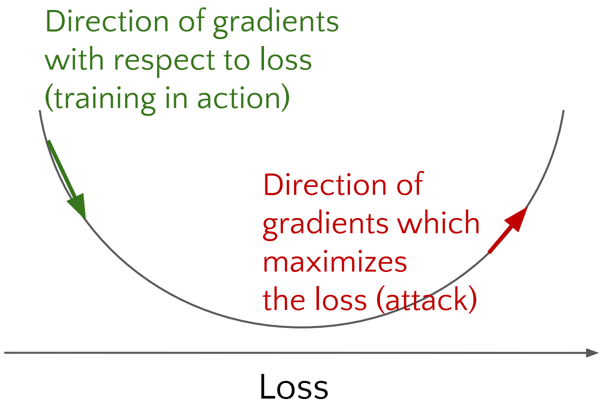
$$\max_{Z \in \Omega} g(Z;X) > 0$$

$$g(Z;X) = \max_i \max(\, f_i(Z) - u_i(X),\; l_i(X) - f_i(Z)\,)$$

Counter-example with Adversarial Attack

$$\max_{Z \in \Omega} g(Z;X) > 0 \Rightarrow \exists\ Z \in \Omega \text{ s.t } g(Z;X) > 0$$

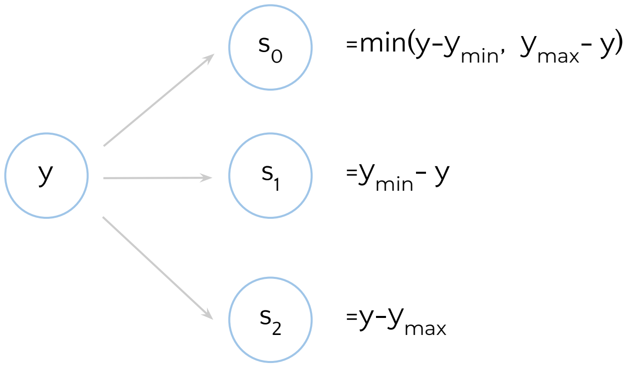




Shows model vulnerabilities but not their absence
>> Do not provide property verification guarantee.

Naïve attacks schemes can be used for regression (FGSM, PGD)

Casting Local Verification as a Classification Property



$$\max_{Z\in\Omega} g(Z;X)\leq 0$$

$$\Updownarrow$$

$$\operatorname{argmax}_{Z\in\Omega} s(Z;X)=0$$

FGSM / PGD

N_{iter} **gradient ascent** on the loss function in x.

(SUPER) DEEPFOOL (S-DF)

iterative **projection** on the closest linearized hyperplane boundary.

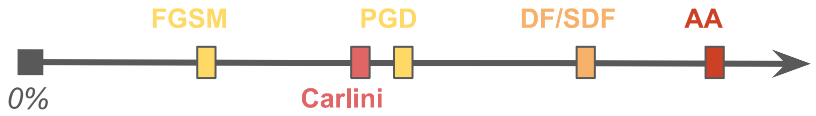
AUTO ATTACK (AA)

fixed **combination** of 3 attacks (AutoPGD, FAB, Square) with different losses

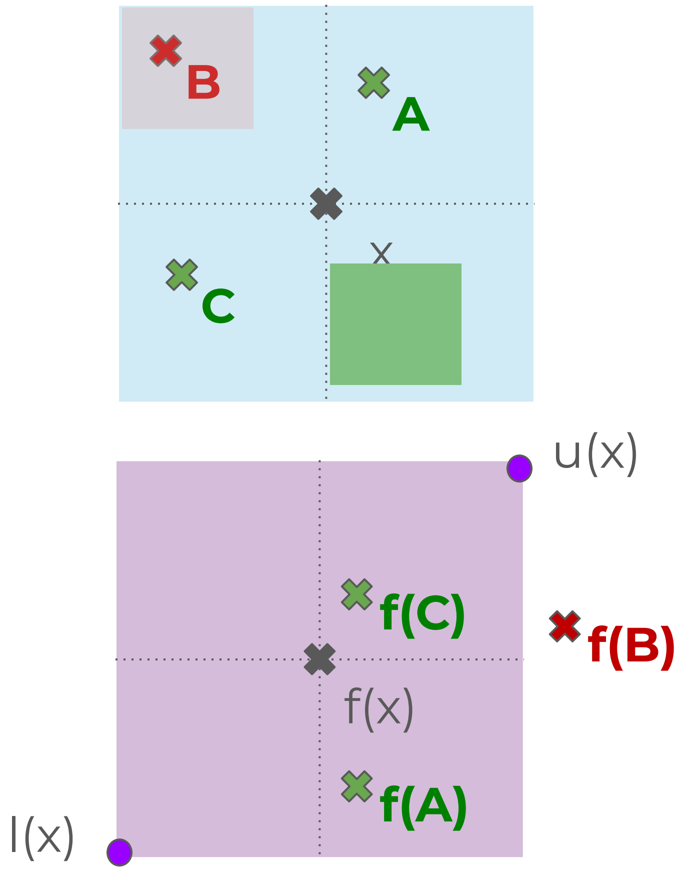
CARLINI & WAGNER (C&W)

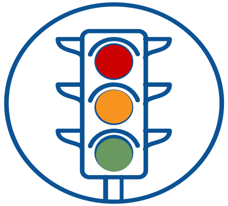
Targeted attack based on the logits.

FOOLING RATE (success rate)



Verification as an Exact Optimization problem





$$\max_{Z \in \Omega} g(Z;X)>0$$

- convergence to the true optimum that implies robustness or non robustness.
- Not scalable to larger network

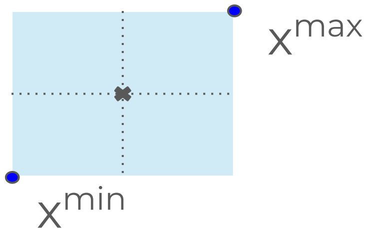
  SMT-solver [Marabou]

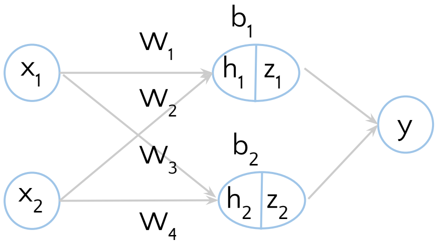
  Lipschitz optimization **(Paul Novello)**

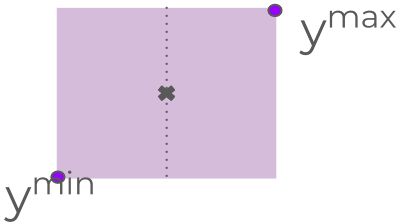
  Mixed Integer Programming (VENUS)

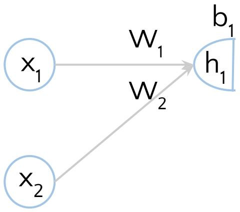
No magical trick:
white box setting

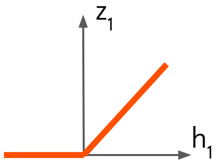
Verification as an Exact Optimization problem: MILP











Bounding the input perturbation

$$x_i^{min} \leq x_i \leq x_i^{max}$$

Encoding Neural Network

$$h_1 = w_1.x_1 + w_2.x_2 + b_1$$
$$h_2 = w_3.x_1 + w_4.x_2 + b_2$$

Big M encoding for ReLU

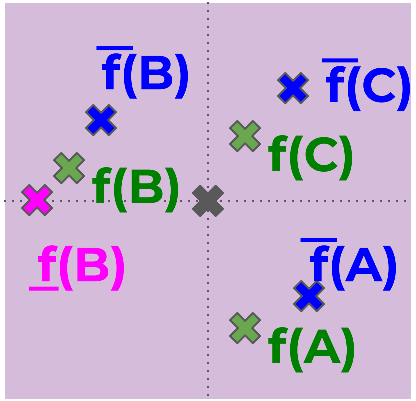
$$\begin{array}{l} z_1 = \max(0,h_1) \\ z_2 = \max(0,h_2) \end{array} \equiv \begin{array}{ll} z_i \geq 0 \ \ldots & z_i \leq u_i.\delta_i \\ z_i \geq h_i & z_i \leq h_i - (1-\delta_i) \\ \delta_i \in \{0,1\}^{|xi|} & \end{array}$$

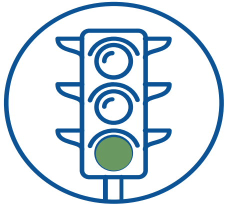
Encoding property violation (SAT)

$$y_1^{max} < y_1 \text{ or } y_1 < y_1^{min} \text{ etc.}$$

Verification as a Relaxed Optimization Problem

AIRBUS





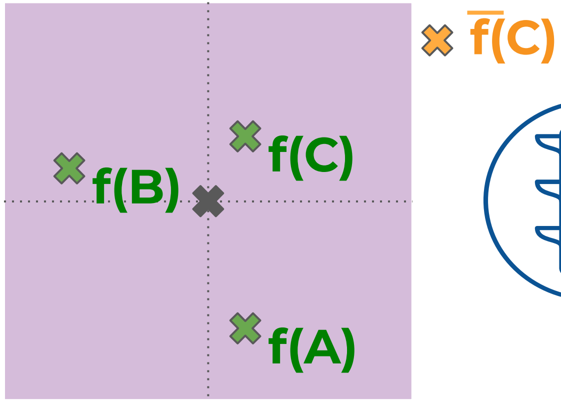
Build Under/Over approximation of f

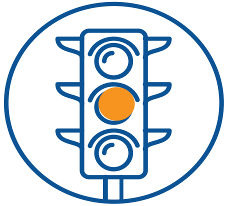
$$\forall z \in \Omega\ \underline{f}(z) \leq f(z) \leq \overline{f}(z)$$

Use it for dominating g

$$\forall z \in \Omega\ g(z) \leq \overline{g}(z)$$

$$\max_{z \in \Omega} \overline{g}(Z;X) \leq 0 \Rightarrow \max_{z \in \Omega} g(Z;X) \leq 0$$



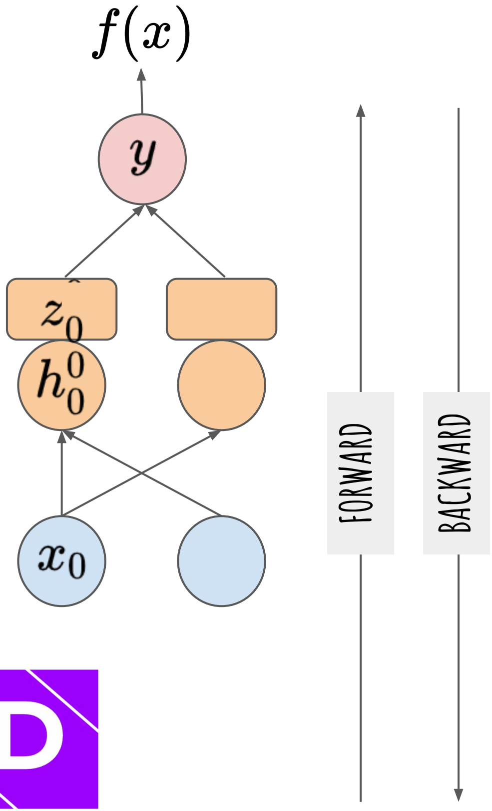


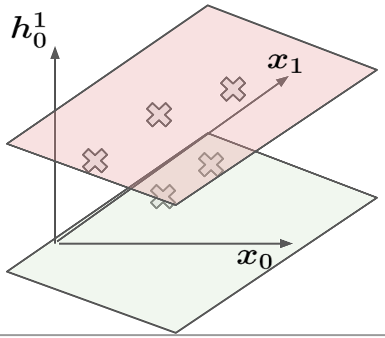
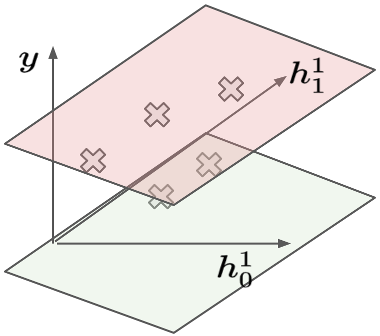
outer-approximations that only implies robustness:

Linear Relaxation [CROWN]
Convex Relaxation [SDP]

Verification as a Relaxed Optimization problem: LIRPA

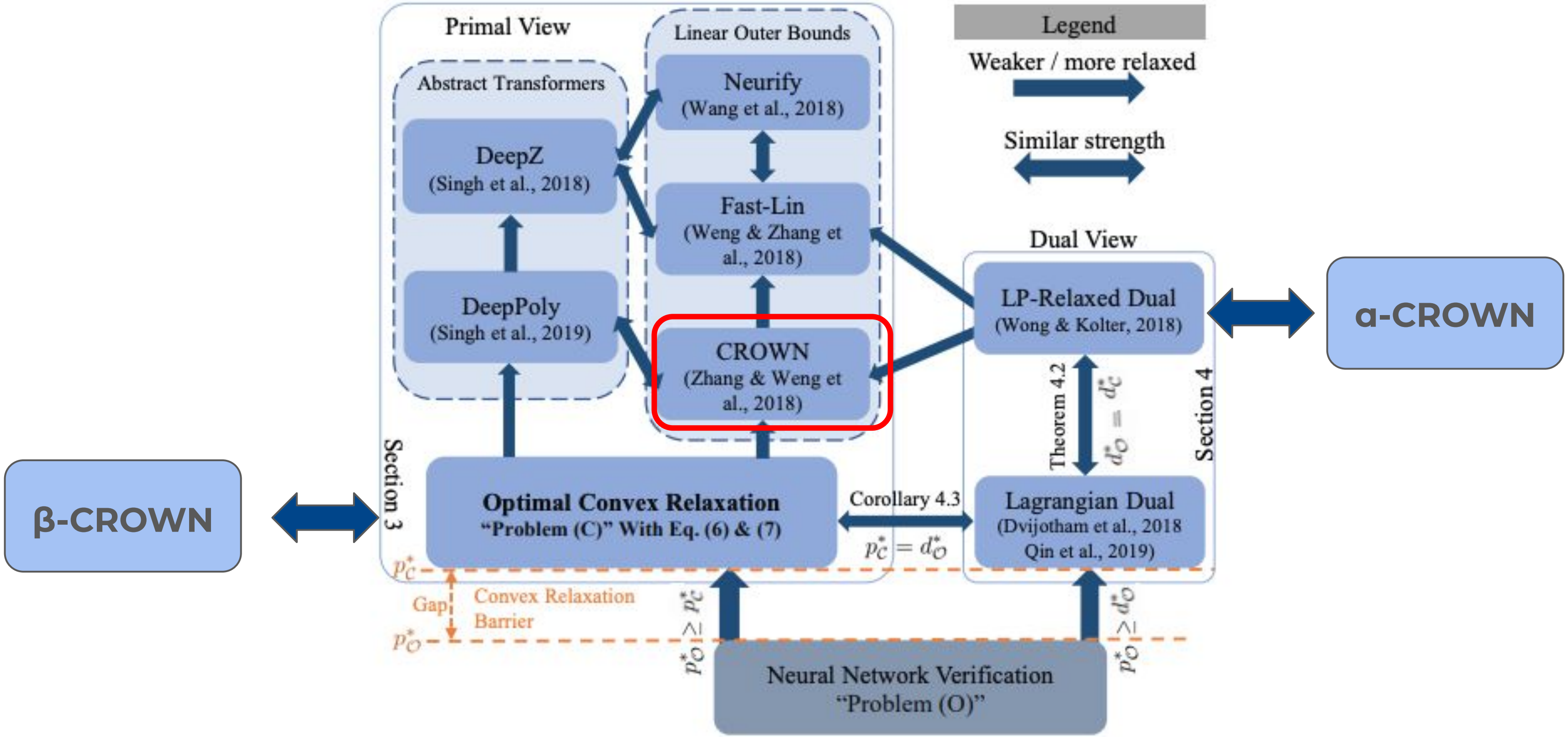
AIRBUS



VERIFICATION 'FORWARD-FEED'	VERIFICATION 'BACKWARD-FEED'
 	
+ Self-sufficient + Complexity = cost of inference - Less accurate - Not scalable on large images	+ Not self-sufficient (pre-processing) + Complexity = cost of backpropagation at best (gradient) - More accurate - Not scalable on large outputs (GAN)

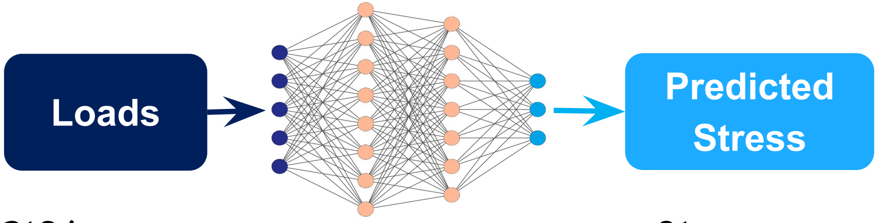
github.com/airbus/decomon

Verification and automatic differentiation



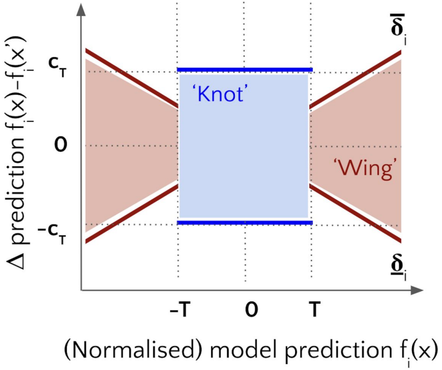
A Convex Relaxation Barrier to Tigh Robustness Verification of NNs, Salman et al.

Aircraft Loads-to-Stress Prediction



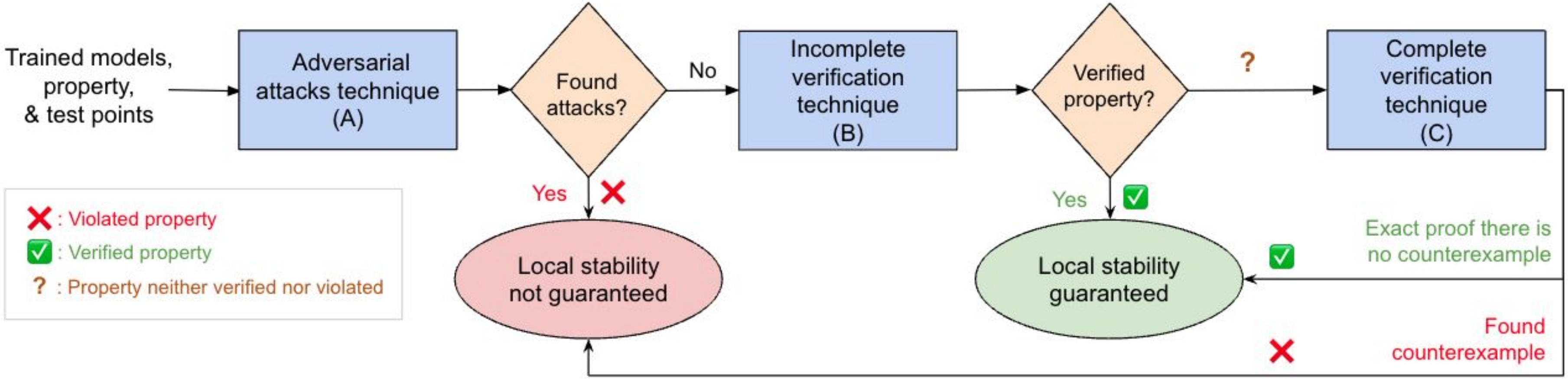
216 inputs

81 outputs

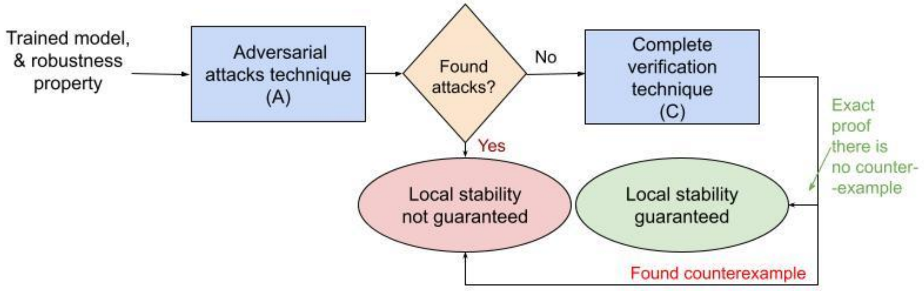


- Model- **<u>Research prototypes</u>**:
  - Two hidden layers (165 neurons)
  - ReLu activation functions
  - Dense output layer (81)
- Test data: 1000 loads/stress points

Verification approach – Combinaison



A+B+C



A+C

A: PGD (Cleverhans)
B: CROWN (Decomon)
C: MILP (Gurobi)

Results

	(1) A	(2) B	(3) C	(4) A+C	(5) B+C	(6) Pipeline A+B+C
#Tested	1000	1000	1000	1000/558	1000/446	1000/558/4
#True	-	554	558	558	558	-/554/4 = 558
#False	442	-	442	442	442	442/-/0 = 442
Runtime	**10.7**	**3.3**	**267**	**19.8**	**267**	**10.7/1.96/3.91 = 16.6**

~45% of test data are shown to be non locally stable

The "Adversarial attacks" step was able to find all non-stabilities

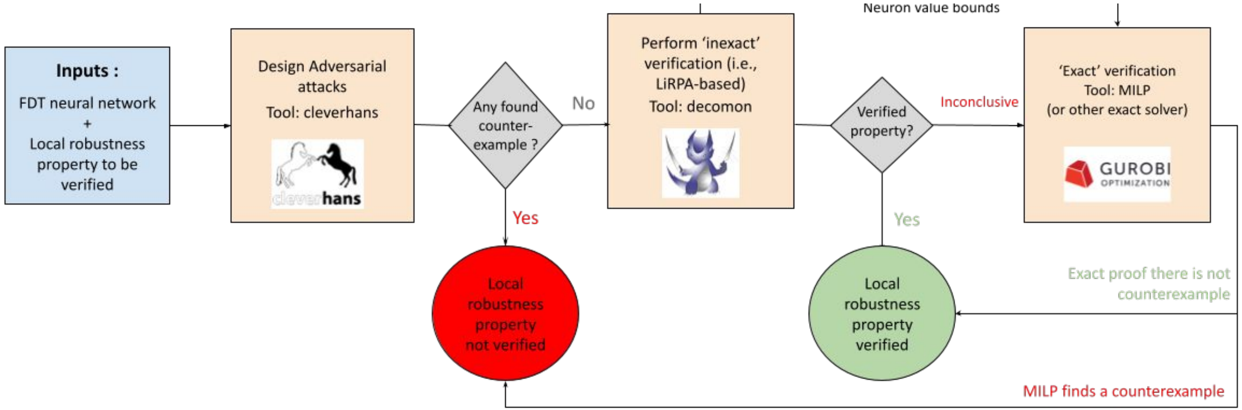
Low number of remaining test data to be evaluated by "C" after (A or A+B)

Significant decrease in computational time

Open source library: Airobas

Complete Verification Pipeline for Stability

Stability accuracy can be efficiently measured with a verification pipeline



Current models have **deceiving stability accuracy: ~40%**. What tools are at our disposal ?

1) XAI actionability: Reducing the problem complexity (input and output dimensions)
2) **Regularizing the training to balance between good <u>regression performance</u> and good <u>stability accuracy</u>**

Stability training

AIRBUS

Robust = Accurate + Stable

Data Augmentation

Artificially increase the size of the dataset by applying domain-specific transformations on the input and output data. It introduces stability invariance.

Weights Constraints

Weights constraints limit the Lipschitz constant in a neural network.It is known to increase the model's resilience against adversarial attacks or input perturbations by limiting the model's capacity to fit noise

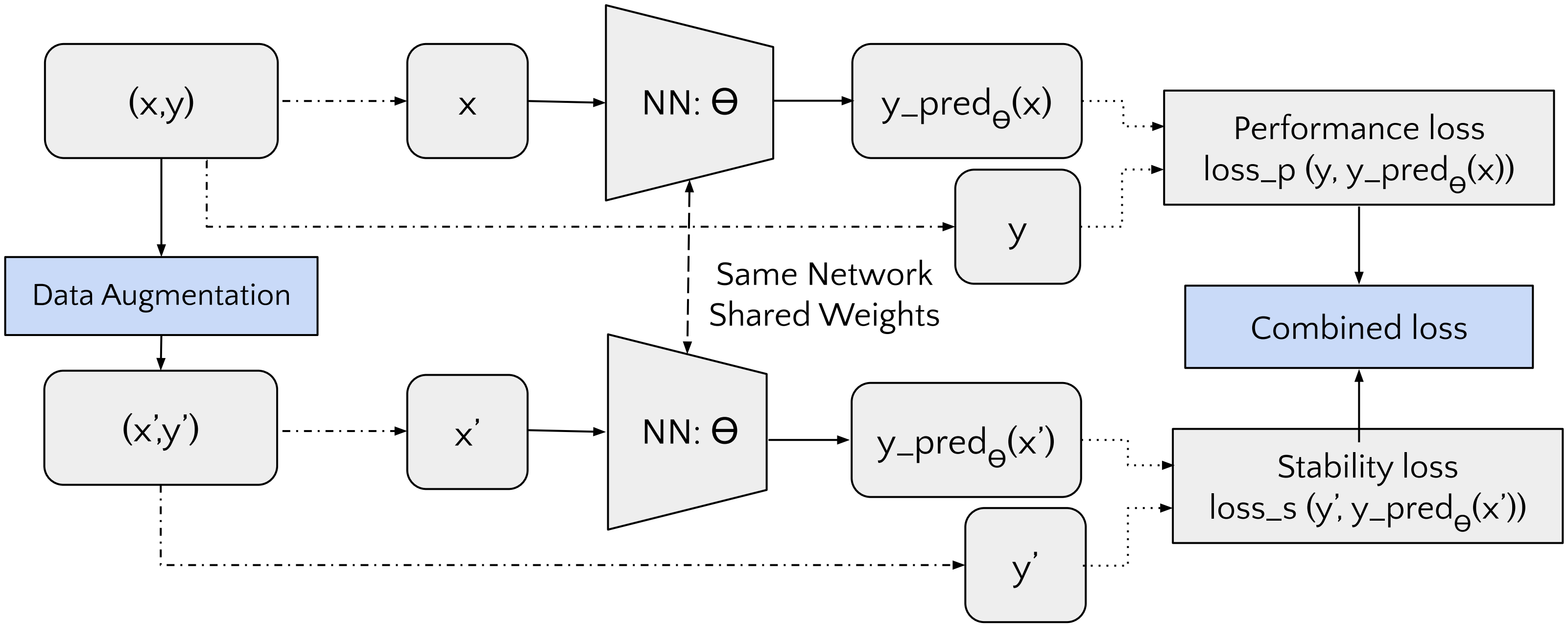
Certified Training (Meta Networks)

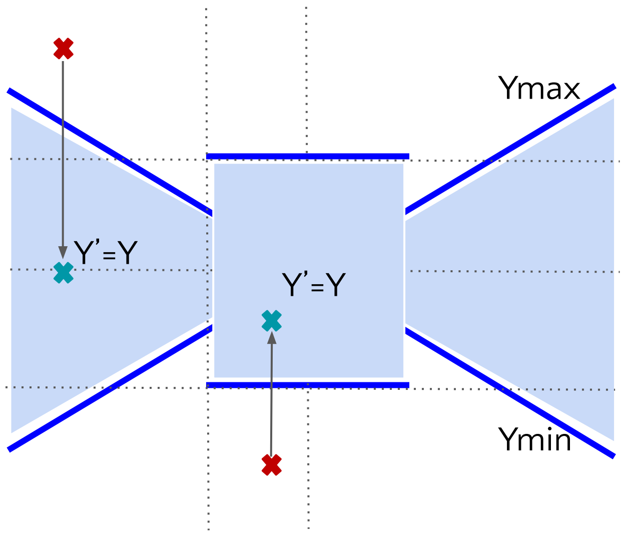
Certified training use Incomplete Formal Methods as a Meta Model to provide formal guarantees about a model's robustness against domain-specific perturbations.

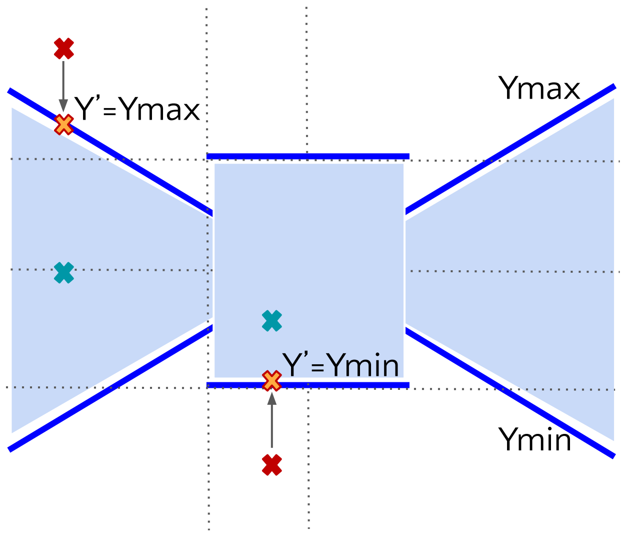
Data Augmentation

Artificially increase the size of the dataset by applying domain-specific transformations on the input and output data. It introduces stability invariance.

Enhance stability during design/training via targeted data augmentation



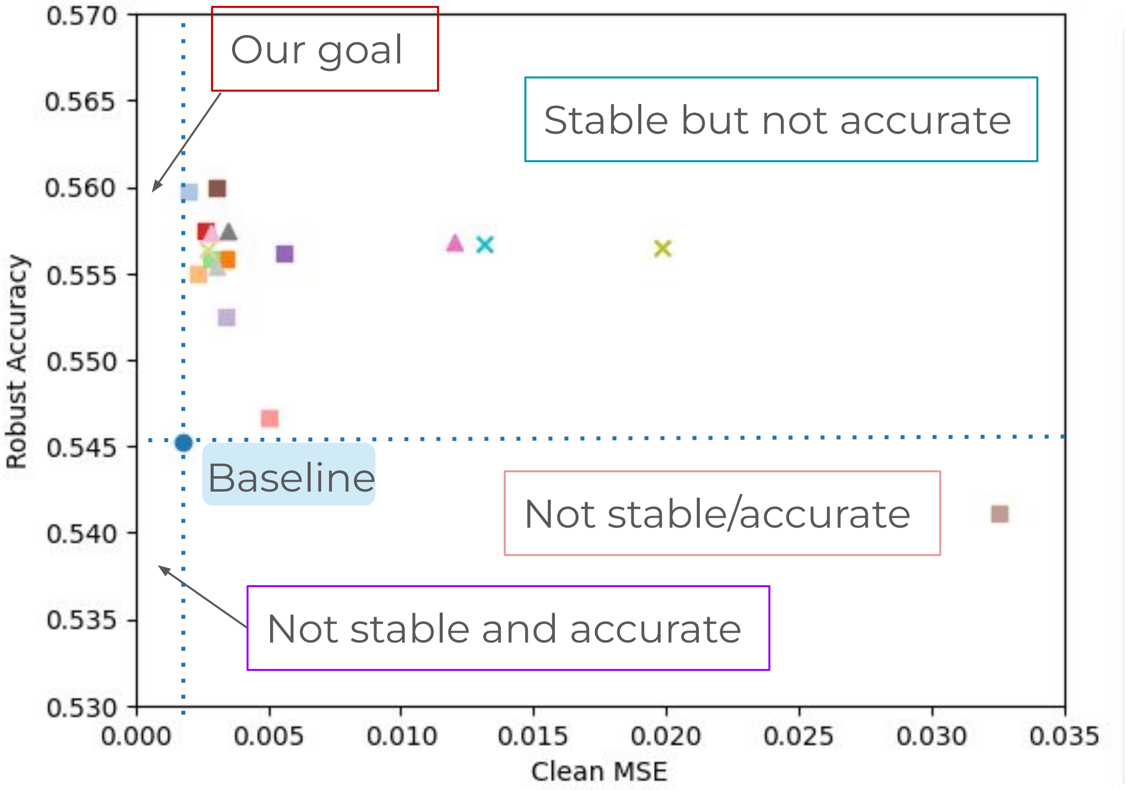




Groundtruth
use the groundtruth label of the initial input

Stability clipping
clip the prediction of the adversarial input to lie within the stability bounds (Ymin, Ymax)

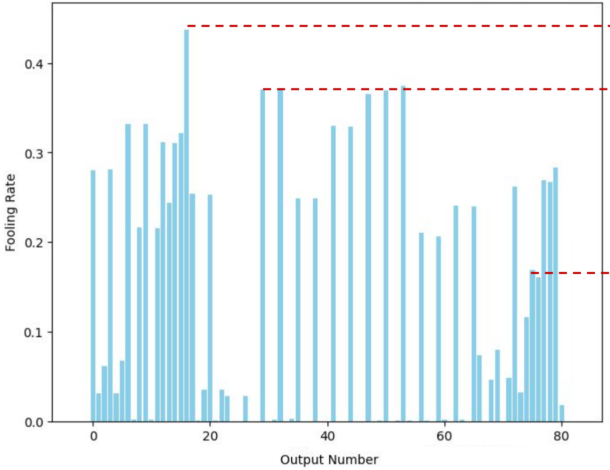
Results & Analysis : Impact of DA on Robustness



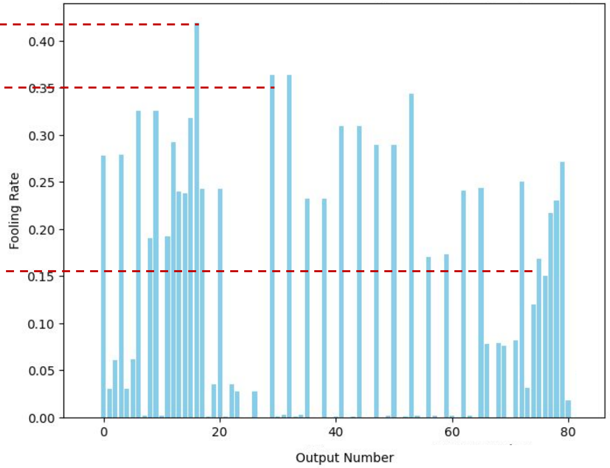
Results & Analysis : Robustness of each output

Fooling Rate = 1 - Robust_accuracy

Baseline



Data Augmentation



- Most of the outputs are **naturally Robust**
- About **40%** of outputs are **problematic**

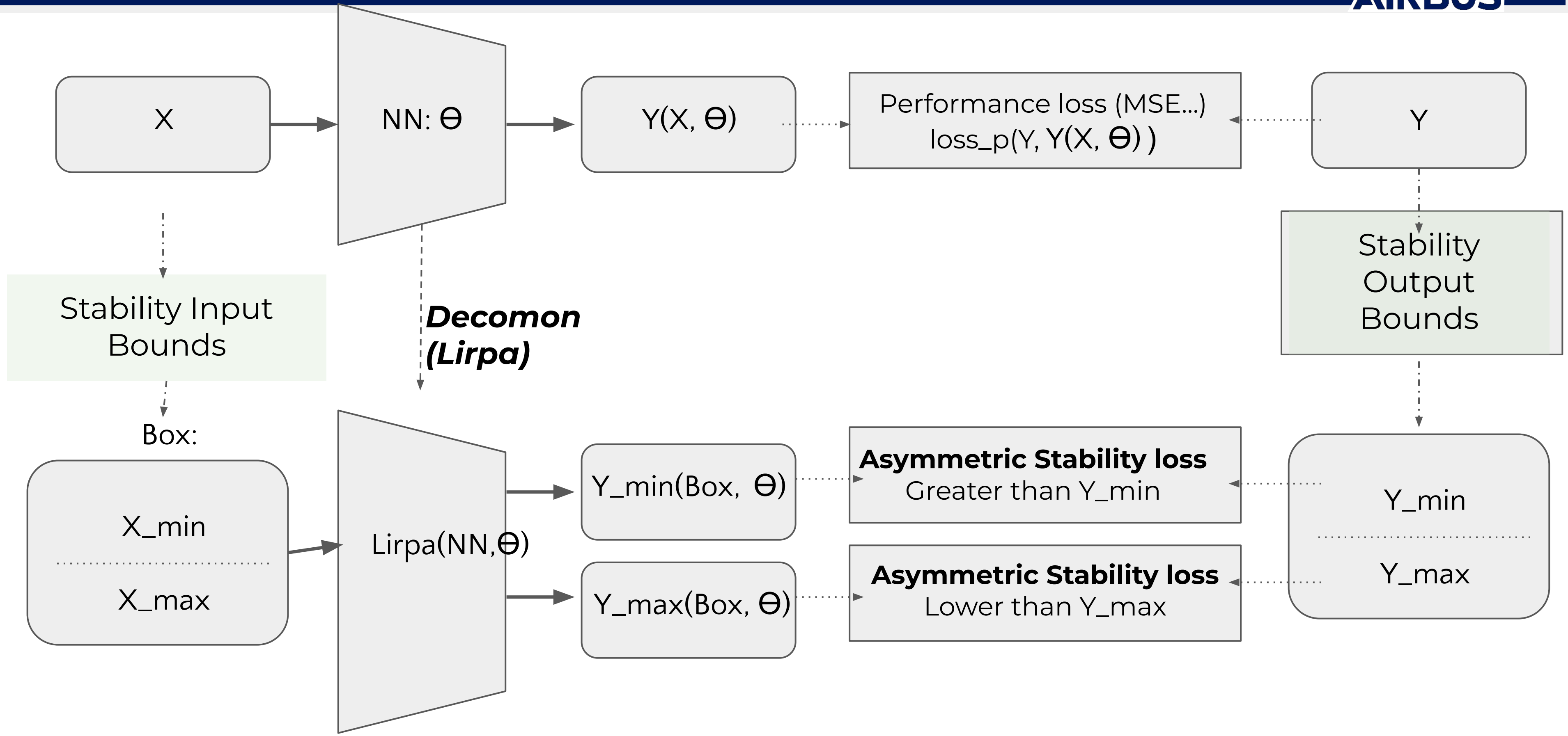
➔ Can we **target** those outputs ?

AIRBUS

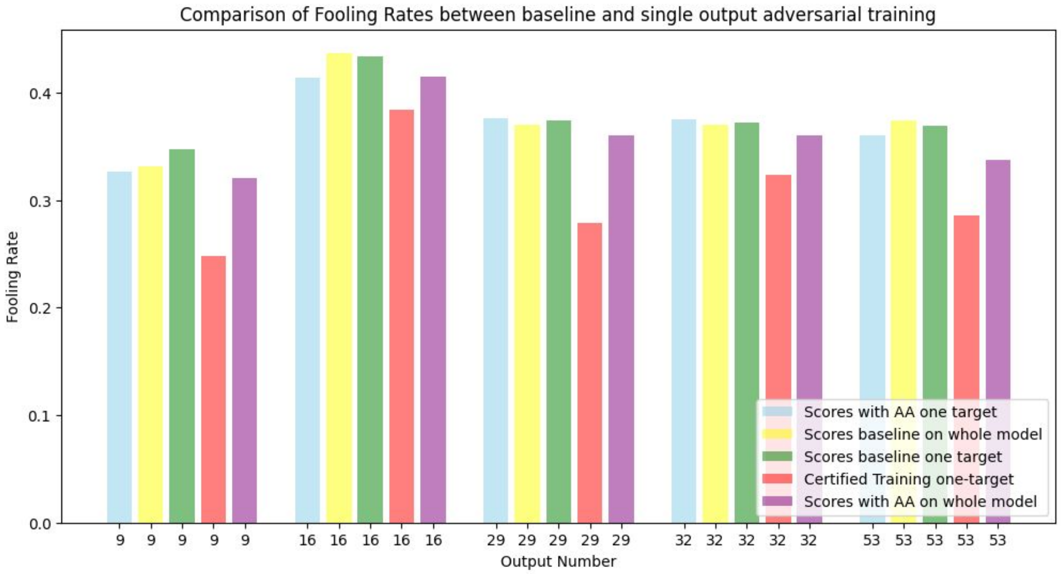
Certified Training (Meta Networks)

Certified training use Incomplete Formal Methods as a Meta Model to provide formal guarantees about a model's robustness against domain-specific perturbations.

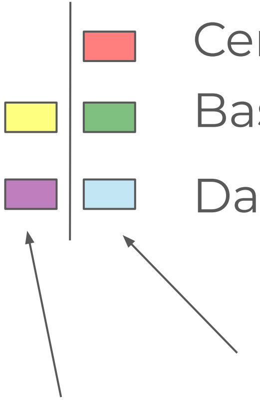
Training Pipeline : Stability with Certified training



Results & Analysis : Certified Training on single-output models



- **CT for single-output models** for the 5 problematic outputs



Certified training

Baseline trainings

Data Augmentation

single-output

multi-output

Promising results ! 5-10% drop in the fooling rates compared to the previous models

Surrogate Neural Networks Local Stability for Aircraft Predictive Maintenance, FMICS 2024

Thomas Deltort

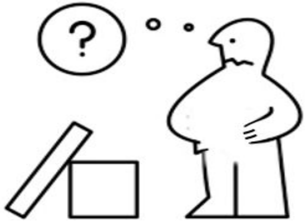
Ryma Boumazouza

Guillaume Poveda

Marion Cécile Martin

Audrey Galametz

ANITI EVENT: Hands on Verification
6th March 2025



https://github.com/airbus/Airobas

https://github.com/airbus/decomon