WAISE

SAFECOMP 2022

Safe Design of Stable Neural Networks for Fault Detection in Small UAVs

Kavya Gupta

Co-authors:
Jean-Christophe Pesquet (CS)
Fateh Kaakai and Beatrice Pesquet-Popescu (Thales LAS France)

6th September 2022

CentraleSupélec | université PARIS-SACLAY

CPIS Inria

THALES

Motivation:

Ubiquitous use of AI systems - increased interest from the industry.

- Examples:
  - Air Traffic Management, Aeronautics, Autonomous Vehicles , Medical Systems, Space
  - All safety critical systems requiring certification from competent authorities
- DNNs can be fooled - certification issues:
  - Instability to perturbations.
  - Unstable to adversarial examples changing few pixels may lead to entirely different results
- Solution:
  - Develop “safe-by-design” AI systems using neural networks
  - Quantifying the stability property of neural network model
  - Develop design methods for ML engineers

CentraleSupélec | université PARIS-SACLAY

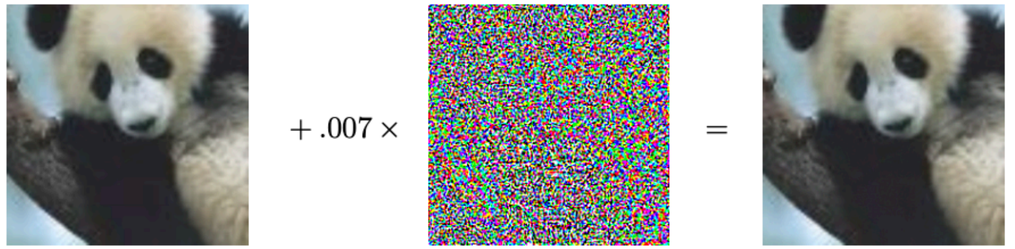
CPIS Inria

THALES

Vulnerability in DNNs:

Pose a huge threat to the performance of DNNs

- Adding a subtle perturbation to the input of the NN produces an incorrect output, while human eyes cannot recognise the difference.



the model considers the original image to be a "panda" (57.7%). After adding a slight perturbation it is classified as a "gibbon" by the same model with 99.3% confidence, while the human eyes completely cannot distinguish the differences [2014]

1. Goodfellow, I. J., Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Adversarial Robustness:

- Provide only Empirical Robustness.
- Gives superficial sense of stability to the systems.
- Can be easily broken with new stronger and aggressive attacks.
- Example : adversarial training.

Certified Robustness:

Provable defences

- Provide formal robustness guarantees that any norm bounded adversary won’t be able to alter the prediction of the neural network.
- Stronger notion of stability
- Example : Randomized smoothing , abstract interpretation , optimising for lower Lipschitz constant.

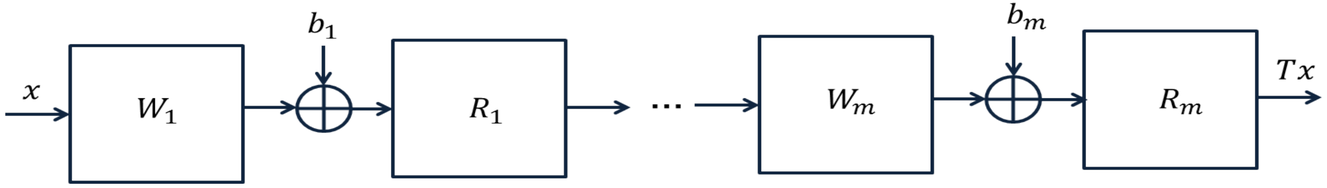
Lipschitz Bounds of Neural Networks:

- Tight estimation of Lipschitz constant - used for robustness certification and stability analysis.
- Estimating Lipschitz constant both accurately and efficiently remains a challenge

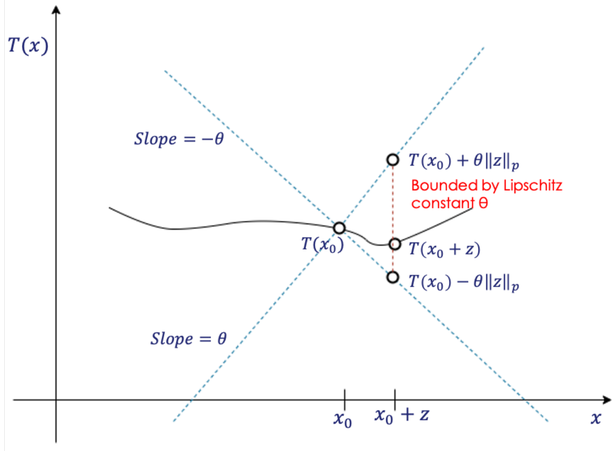
- Lipschitz bounds
  - An upper bound on the ratio between the variations of the outputs and the variations of the inputs of a function T, thus measuring the sensitivity of the function w.r.t to input perturbations z.
  - For an m-layered feed forward network T, with N_i neurons at layer $i \in \{1,\dots,m\}$, $x \in \mathbb{R}^{N_0}$ is the inputs $T(x) \in \mathbb{R}^{N_m}$ is the associated output , θ is the Lipschitz constant of T if:

$$\|T(x+z) - T(x)\| \leq \theta \|z\|$$

for every input $x \in \mathbb{R}^{N_0}$ and perturbation $z \in \mathbb{R}^{N_0}$



m-layered fully connected network architecture.



Various Lipschitz Bounds:

Method	Main Formula	Properties
First Upper Bound [2014]	$\overline{\theta}_m = \|W_m\|_{\mathrm{S}} \|W_{m-1}\|_{\mathrm{S}} \cdots \|W_1\|_{\mathrm{S}}$	Spectral Norm , loose bound
CPLip [2020]	$\vartheta_m = \sup_{\Lambda_1 \in \mathcal{D}_1, \ldots, \Lambda_{m-1} \in \mathcal{D}_{m-1}} \|W_m \Lambda_{m-1} \cdots \Lambda_1 W_1\|_{\mathrm{S}}$	l_p Norm, $p \in [1, +\infty]$ not scalable to broad networks
LipSDP[2019]	$\underset{(\rho_m, Q_1, \ldots, Q_{m-1}) \in C}{\text{minimize}} \quad \rho_m$	Restricted to l_2 norm , more scalable to broad networks
LipOpt-k [2020]	$\vartheta_m = \sup_{\Lambda_1 \in \mathcal{D}_{N_1}, \ldots \Lambda_{m-1} \in \mathcal{D}_{N_{m-1}}} \|W_1^\top \Lambda_1 \cdots \Lambda_{m-1} W_m^\top\|_{p^*}$	l_p Norm , $p \in [1, +\infty]$

Computing the exact Lipschitz constant with differentiable activation function is NP hard and is restricted to fully connected networks

1. Patrick L Combettes and Jean-Christophe Pesquet, "Lipschitz certificates for neural network structures driven by averaged activation operators," *SIAM Journal on Mathematics of Data Science*, vol. 2, pp. 529–557, 2020.
2. Mahyar Fazlyab, Alexander Robey, Hamed Hassani, Manfred Morari, and George Pappas, "Efficient and accurate estimation of Lipschitz constants for deep neural networks," in *Advances in Neural Information Process- ing Systems*, 2019, pp. 11423–11434.
3. Fabian Latorre, Paul Rolland, and Volkan Cevher, "Lipschitz constant estimation of neural networks via sparse polynomial optimization," *arXiv preprint arXiv:2004.08688*, 2020.

Controlling Stability:

Aiming for tighter Lipschitz Bounds

- Parseval Networks with Orthogonal weights with Lipschitz constant of 1
- Spectral Normalisation such as in GANs
- 1-Lipschitz networks can be very restrictive in terms of clean performance
- Trade-off between stability and performance

Spectral Normalisation:

- DNNs are trained with layer-wise spectrally normalised weight matrices
- At each layer: $\tilde{v} \leftarrow W\tilde{u}/\|W\tilde{u}\|_2 \qquad \tilde{u} \leftarrow W^T\tilde{v}/\|W^T\tilde{v}\|_2 \qquad \sigma(W) \approx \tilde{v}^T W\tilde{u}$

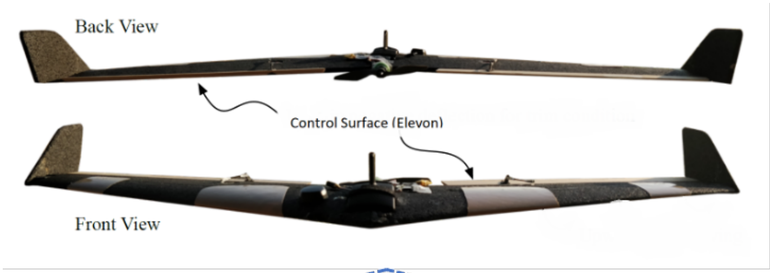
$$W_{SN} = W/\sigma(W)$$

- This might be too restrictive and lead to poor performance of the network
- To relax this: $$W_{SN} = W/\sigma(W) \,.\, \theta_{target}^{1/m}$$
- Where θ_{target} is the desired Lipschitz constant of the DNN

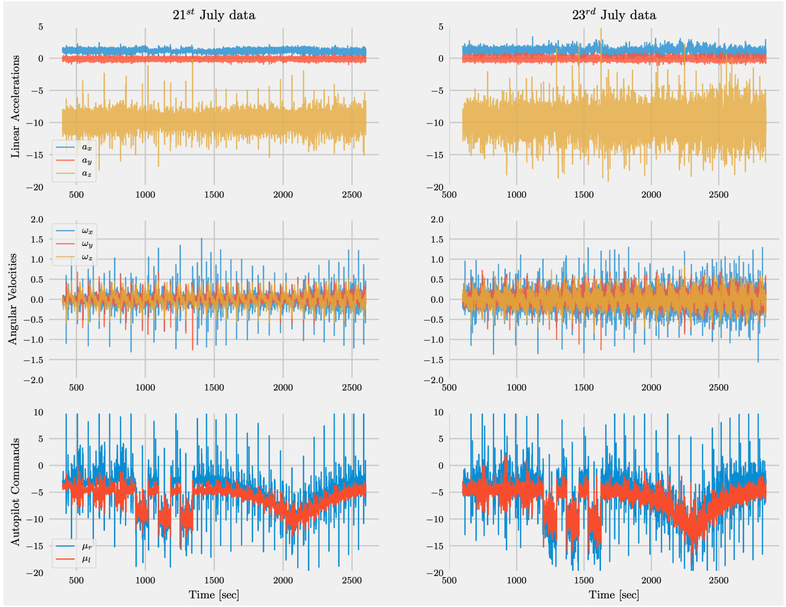
Unmanned Aerial Vehicle use case:

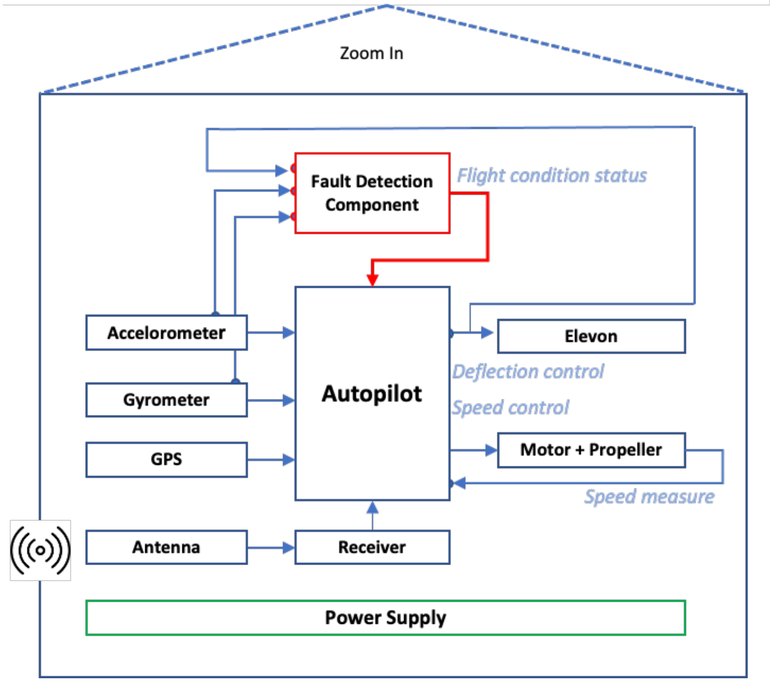
Collaboration within EUROCAE WG114

Fault Detection in drones/UAVs : Detection of Fault in the Right and Left Elevon.
Task: Binary and Multi-class classification
SOA: SVM [2020]



Input	0	a_x
	1	a_y
	2	a_z
	3	ω_x
	4	ω_y
	5	ω_z
	6	u_{com_r}
	7	u_{com_l}
Output	Class	Nominal or faulty

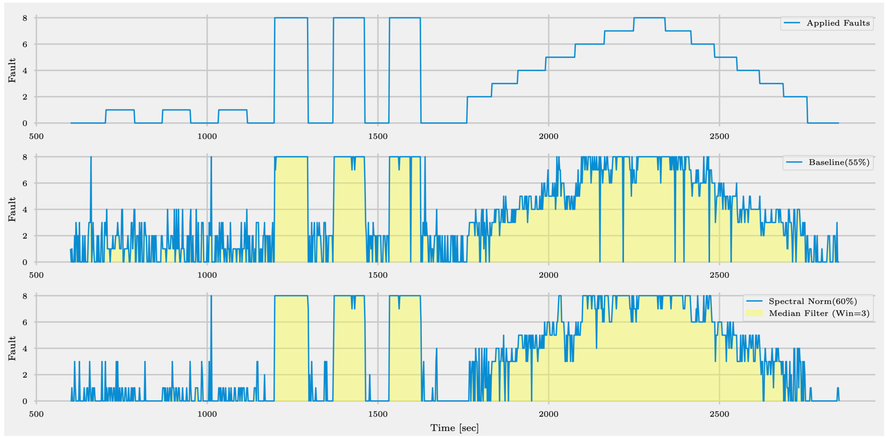




Spectral Normalisation based Stability Loop:

Inputs : training dataset $(X_{\text{train}}, Y_{\text{train}})$, validation dataset $(X_{\text{valid}}, Y_{\text{valid}})$, target Lipschitz constant L_{target}, target accurcay $\text{acc}_{\text{target}}$, neural network f, multiplicative factor $\alpha > 1$, maximum iteration number $n_{\max}$
Output : optimal neural network T_{opt}, Lipschitz constant L_{opt}, and accuracy acc_{opt}
Algo :
$L \leftarrow L_{\text{target}}$, $n \leftarrow 0$
$T \leftarrow \mathsf{Train}_{\text{SN}}(L, X_{\text{train}}, Y_{\text{train}})$
$L_{\text{est}} \leftarrow \text{LipEst}(f)$
$\text{acc} \leftarrow \mathsf{Predict}(f, X_{\text{valid}}, Y_{\text{valid}})$
while $(n < n_{,\max})$ and $(L_{\text{est}} < L_{\text{target}})$ and $(\text{acc} < \text{acc}_{\text{target}})$ **do**
    $L \leftarrow \alpha\, L$
    $T \leftarrow \mathsf{Train}_{\text{SN}}(L, X_{\text{train}}, Y_{\text{train}})$
    $L_{\text{est}} \leftarrow \mathsf{LipEst}(f)$
    $\text{acc} \leftarrow \mathsf{Predict}(f, X_{\text{valid}}, Y_{\text{valid}})$
    $n \leftarrow n + 1$
end while
$T_{\text{opt}} \leftarrow T$
$L_{\text{opt}} \leftarrow L_{\text{est}}$
$\text{acc}_{\text{opt}} \leftarrow \text{acc}$
if $n = n_{,\max}$ **then**
    Notification: No trade-off found between Stability and Accuracy
else
    Notification: Trade-off found between Stability and Accuracy
end if

Results:



Results for Neural network training
LC (Baseline) = 37
LC (Spectral Normalization) = 5

Model	Baseline		Spectral	
	Acc. (%)	L	Acc. (%)	L
(10)	66.4	12.26	75.1	3.7
(10,6)	71.0	27.56	75.3	3.8
(50,10,6)	73.3	37.76	**76.7**	**3.7**

Binary Classification

Model	Baseline		Spectral	
	Acc. (%)	L	Acc. (%)	L
(50,10)	69.7	31.44	**72.65**	3.8
(100,10)	69.54	34.17	71.07	**3.6**
(100,75,10)	68.9	69.06	71.93	3.7

Multi-class Classification

Spectral Normalisation training improved both classification performance as well as stability of NN

Observations:

- NNs have better generalisation than SVM classifiers.
- Lipschitz constant increases as the number of layers and neurons in the neural network increases
- Steep decrease in the Lipschitz constant value when the models are trained with Spectral normalisation constraint
- Spectral normalised trained models are better against adversarial attacks implying since they are designed to be more stable.

Conclusion:

- Theoretical guarantees for better understanding and controlling the stability of NN
- We provide a method to generate “stable-by-design” neural network
- We maintain a stability-performance trade-off
- Main objective: ensuring the safety of new generations of industrial products based on AI

Thank you.

Email: kavya.gupta100@gmail.com

I am graduating this year and looking for full-time research positions.