NAIS-Net: Stable Deep Networks from Non-Autonomous Differential Equations

Marco Ciccone * [1, 2], Marco Gallieri * † [2], Jonathan Masci [2], Christian Osendorfer [2], Faustino Gomez [2]

marco.ciccone@polimi.it, { marco, jonathan, christian, tino } @ nnaisense.com

*: The authors equally contributed. †: The author derived the mathematical results.

[1]: Dipartimento di Elettronica, Informatica e Bioingegneria - Politecnico di Milano. [2]: NNAISENSE SA, Lugano, CH.

Introduction

Training Very Deep Networks has been made possible thanks to the use of additive non-linear transformations (Skip-connections), such as in Highway [1] and Residual Networks [2] :

$$x(k+1) = x(k) + f\left(x(k), \theta(k)\right), 1 \leq k \leq K. \tag{1}$$

- Skip-connections solve vanishing gradient problem.
- Output normalization is required to train (e.g BatchNorm).
- The semantics of the forward path are still unclear (iterative estimation).

Note : Very Deep Networks sharing this structure can be considered as **Dynamical Systems**. Indeed, Eq. 1 can be seen as the Forward Euler Discretization of the ODE $\dot{x} = f(x)$.

Idea : Use **Control Theory** to analyze the behavior of these networks in terms of the stability of their underlined dynamical system. We want the network **to have** a stable behavior such that the propagation of the state do not fluctuate.

Residual Networks are Autonomous Dynamical Systems.

- Input is connected only to the first layer.
- Stability means output $\rightarrow 0$ for each input : **useless for ML applications.**

Idea : Use **input connections** to define Non-Autonomous Systems.

Background : Stability Theory

Asymptotic Stability for Non-Linear Systems

A system is said to be asymptotically stable in $\mathcal{X}$ if there exists a $\bar{x}$ and $\mathcal{KL}$-function β such that $\forall x(0) \in \mathcal{X},\ k \geq 0$:

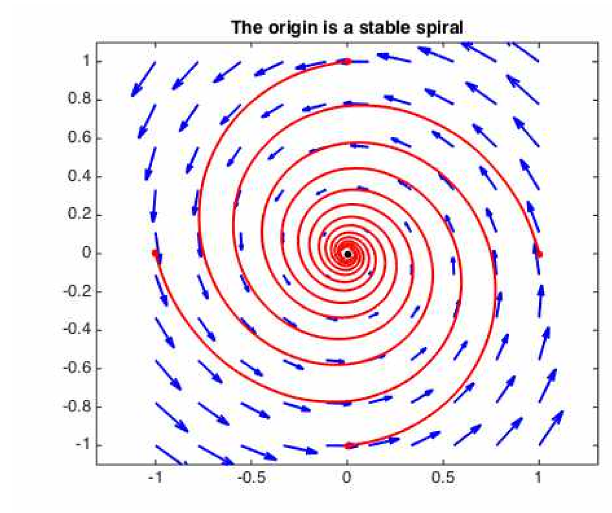
$$\|x(k) - \bar{x}\| \leq \beta(\|x(0) - \bar{x}\|, k). \tag{2}$$

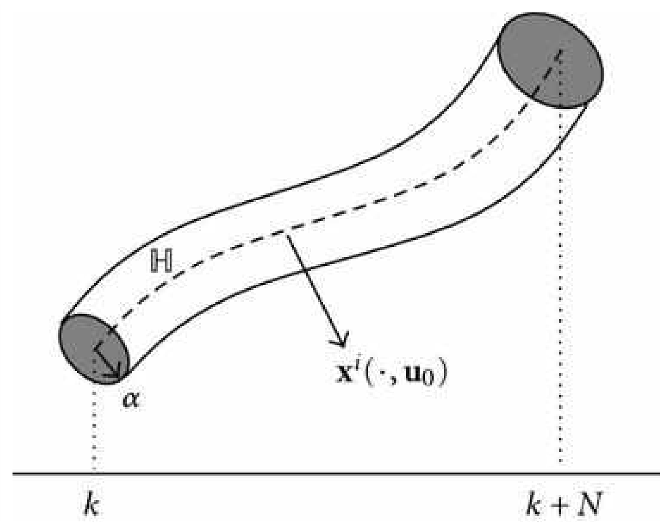
The vector $\bar{x}$ is called a steady state. β have to be strictly decreasing in k with $\lim_{k\to\infty} \beta(\cdot, k) \to 0$.

Input-Output Stability for Non-Linear Systems [3]

A system is said to be input-output stable (IOS) wrt bounded additive input perturbations, w, while $x \in \mathcal{X}$ if there exists a $\mathcal{KL}$-function β and a $\mathcal{K}_\infty$ function γ such that $\forall x(0) \in \mathcal{X}$:

$$\|x(k) - \bar{x}\| \leq \beta(\|x(0) - \bar{x}\|, k) + \gamma(\|w\|). \tag{3}$$





NAIS-Net block : Non-Autonomous Residual Layer

NAIS-Net fully-connected block is defined by the following **non-autonomous system** :

$$x(k+1) = x(k) + h\sigma\Big(Ax(k) + Bu + b\Big), \tag{4}$$

where $x \in \mathbb{R}^n$ is the latent state, $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ are the *hidden state* and input transfer matrices, $h \in (0, 1]$, $b \in \mathbb{R}^n$. Activation σ is tanh or ReLU.

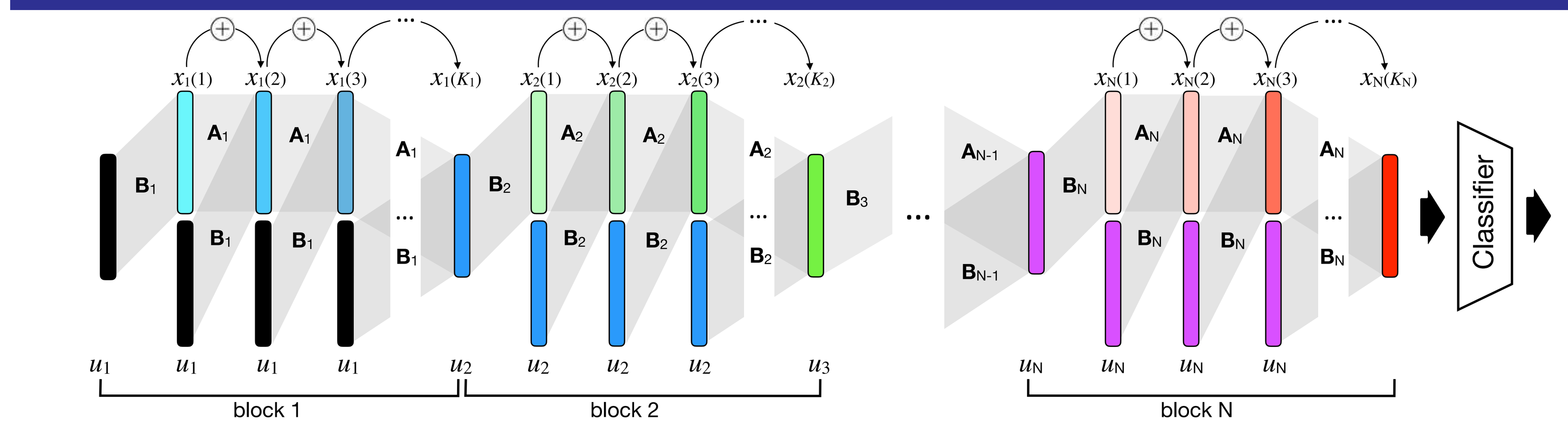
If $B = 0$, $x(0) = u$, then we have a classic ResNet (autonomous).

The **state-transfer Jacobian** for layer k is :

$$J(x(k), u) = \frac{\partial x(k+1)}{\partial x(k)} = I + h\underbrace{\frac{\partial\sigma(\Delta x(k))}{\partial \Delta x(k)}}_{\text{residual Jacobian}}A, \tag{5}$$

where $\Delta x(k)$ is the argument of the activation function σ. Same holds for convolutional layers, where A is Toeplitz.

NAIS-Net : Non-Autonomous Input-Output Stable Architecture



- **NAIS-Net architecture** is a cascade of a time-invariant dynamical systems.
- Each block is an **iterative process** as the first layer in the i-th block, $x_i(1)$, is unrolled K_i times.
- The skip connections from the input, u_i, to all layers in block i make the process **non-autonomous**.
- **Latent space dynamics :** each block is modeling the trajectories of the input in different latent space.
- **IO-stability** and **asymptotic stability** make the trajectories to be bounded with respect to noise perturbations.
- Each block converges to input-dependent attractors (latent representations).

NAIS-Net block Stability

Take an *arbitrarily* small scalar $\underline{\sigma} > 0$ and define the set :

$$\mathcal{P} = \left\{(x, u) : \frac{\partial\sigma_i(\Delta x(k))}{\partial \Delta x_i(k)} \geq \underline{\sigma},\ \forall i \in [1, 2, \ldots, n]\right\}. \tag{6}$$

Stability Condition (from Lyapunov indirect method)

For any small scalar $\underline{\sigma} > 0$, the *state* Jacobian, $J(x, u)$, satisfies :

$$\bar{\rho} := \sup_{(x,u)\in\mathcal{P}} \rho(J(x, u)) < 1, \tag{7}$$

where $\rho(\cdot)$ is the spectral radius.

Theorem 1 (Asymptotic stability for shared weights)

If $\bar{\rho} < 1$, then the NAIS-Net block is Asymptotically Stable :

- For *tanh*, $\bar{x} = -A^{-1}(Bu + b)$.
- For ReLU, $\bar{x}$ is *continuous, piecewise affine* in $x(0)$ and u. The network is Locally Asymptotically Stable with respect to each $\bar{x}$.

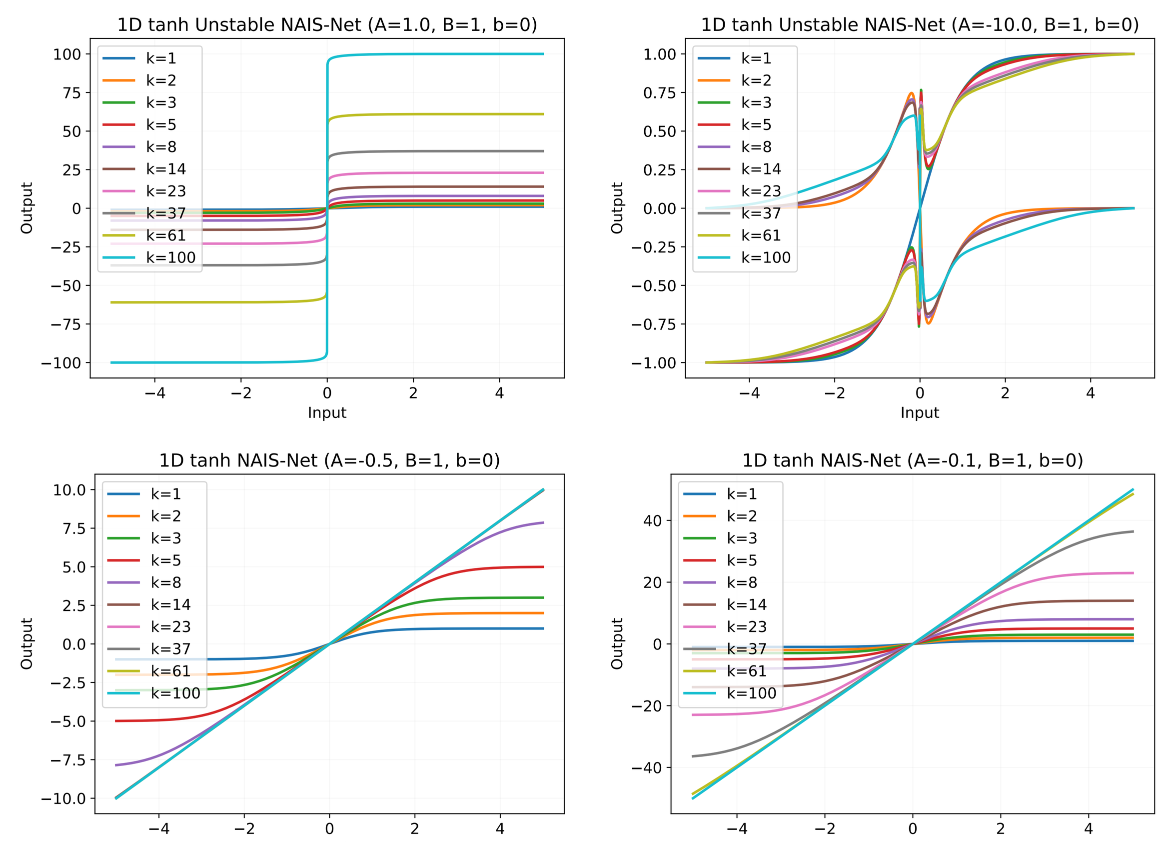
If $\bar{\rho} < 1$, then the NAIS-Net block is Input-Output Stable :

$$\lim_{k\to\infty} \|x(k) - \bar{x}\| \leq \gamma(\|w\|) \tag{8}$$

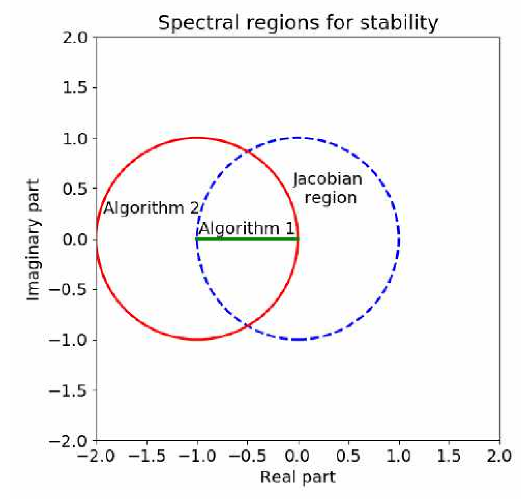
The Input-output gain is :

$$\gamma(\|w\|) = \underbrace{h\frac{\|B\|}{(1-\bar{\rho})}}_{L_w<\infty}\|w\|. \tag{9}$$

L_w is a Lipschitz constant for infinite layers.



Stability Implementation



Fully Connected Stability Reprojection

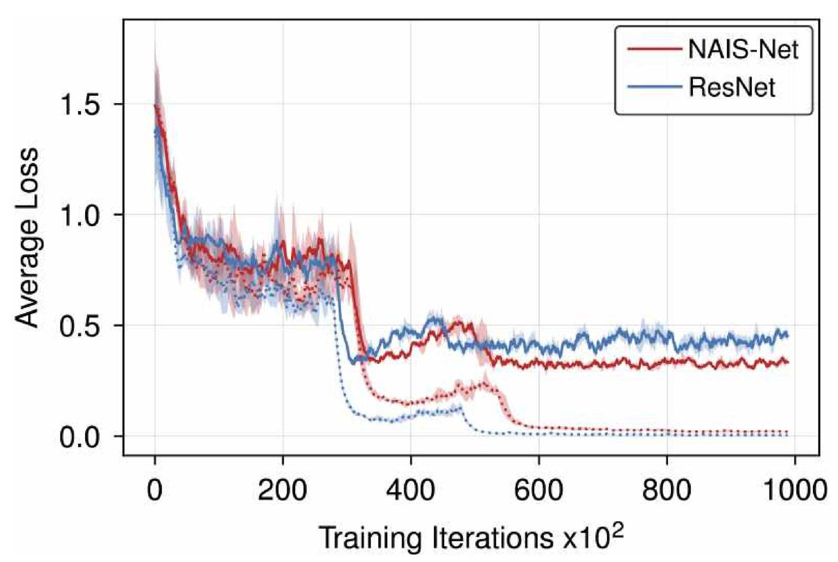
Input : $R \in \mathbb{R}^{\tilde{n}\times n}$, $\tilde{n} \leq n$, $\delta = 1 - 2\epsilon \in (0, 1)$.
if $\|R^T R\|_F > \delta$ **then**
  $\tilde{R} \leftarrow \sqrt{\delta}\frac{R}{\sqrt{\|R^T R\|_F}}$
else
  $\tilde{R} \leftarrow R$
end if
Output : $\tilde{R}$

CNN Stability Reprojection

Input : $\delta \in \mathbb{R}^{N_C}$, $C \in \mathbb{R}^{n_X\times n_X\times N_C\times N_C}$, and $0 < \epsilon < \eta < 1$
for each feature map c **do**
  $\tilde{\delta}_c \leftarrow \max\Big(\min\left(\delta_c, 1-\eta\right), -1+\eta\Big)$
  $\tilde{C}^c_{i_{\text{centre}}} \leftarrow -1 - \tilde{\delta}_c$
  if $\sum_{j\neq i_{\text{centre}}}\left|C^c_j\right| > 1 - \epsilon - |\tilde{\delta}_c|$ **then**
    $\tilde{C}^c_j \leftarrow \left(1 - \epsilon - |\tilde{\delta}_c|\right)\frac{C^c_j}{\sum_{j\neq i_{\text{centre}}}\left|C^c_j\right|}$
  end if
end for
Output : $\tilde{\delta}$, $\tilde{C}$

- Each weight matrix A needs to satisfy $\rho\left(I + h\frac{\partial\sigma(\Delta x(k))}{\partial\Delta x(k)}A\right) < 1$. Because of the Identity sum the stability region is translated (similarly as in Forward Euler).
- Proposed reprojection algorithms can be used with any gradient based optimization method to constrain the weights in the stability region.

Results for Image Classification - CIFAR10-100



Model	CIFAR-10 train/test	CIFAR-100 train/test
ResNet	99.86±0.03	97.42 ± 0.06
	91.72±0.38	66.34 ± 0.82
NAIS-Net1	99.37±0.08	86.90 ± 1.47
	91.24±0.10	65.00 ± 0.52
NAIS-Net10	99.50±0.02	86.91 ± 0.42
	91.25±0.46	66.07 ± 0.24

CIFAR10-100 : accuracy averaged over 5 runs.

- NAIS-Net has a better lower generalization gap wrt ResNet, as a consequence of robustness to input perturbations.
- NAIS-Net can be trained **without requiring batch normalization at each step**.

Pattern-Dependent Processing Depth

- Thanks to stability, NAIS-Net can be unrolled for a variable number of steps until convergence.
- NAIS-Net adapts its depth systematically according to the characteristics of the data.
- Images with similar visual characteristics induce different final depths.
- The depth of the network can be considered as an additional degree of freedom of the model.

R. K. Srivastava, K. Greff, and J. Schmidhuber, "Highway networks," *arXiv preprint arXiv :1505.00387*, May 2015.

K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Dec 2016, pp. 770–778.

H. K. Khalil, *Nonlinear Systems*, 3rd ed. Pearson Education, 2014.