Neural Lander: Stable Drone Landing Control using Learned Dynamics

Guanya Shi[1], Xichen Shi[1], Michael O'Connell[1], Rose Yu[2], Kamyar Azizzadenesheli[3], Animashree Anandkumar[1], Yisong Yue[1], Soon-Jo Chung[1]

[1]California Institute of Technology, [2]Northeastern University, [3]University of California, Irvine

gshi@caltech.edu

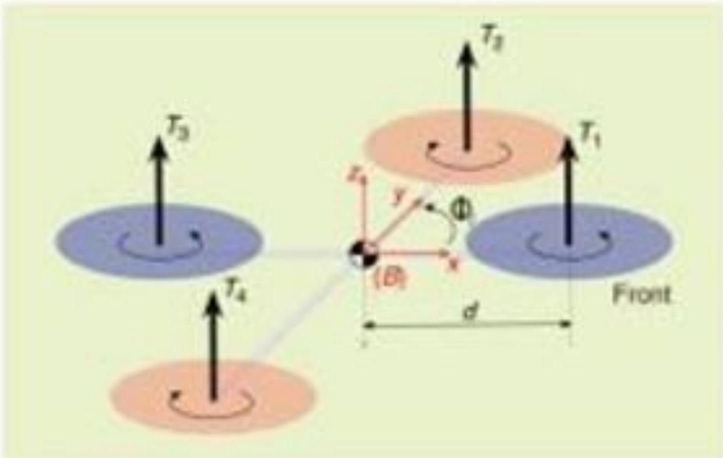
Caltech

1. Abstract arXiv:1811.08027

We present a novel deep-learning-based robust nonlinear controller (*Neural-Lander*) for stable quadrotor control during landing. Our approach blends together a nominal dynamics model coupled with a DNN that learns the high-order interactions.

1. **Sample efficiency.** By blending with a nominal model, our approach is sample efficient (5 minutes real-world data for training).
2. **Provably stability.** By spectrally normalizing the DNN to have bounded Lipschitz behavior, we design **the first DNN-based nonlinear feedback controller with stability guarantees that can utilize arbitrarily large neural nets**.
3. **Generalization.** The bounded Lipschitz behavior also enables generalization outside of its training distribution support.

2. Goal



$$\dot{\mathbf{p}} = \mathbf{v}, \qquad m\dot{\mathbf{v}} = m\mathbf{g} + R\mathbf{f}_u + \mathbf{f}_a,$$
$$\dot{R} = RS(\omega), \qquad J\dot{\boldsymbol{\omega}} = J\boldsymbol{\omega}\times\boldsymbol{\omega} + \boldsymbol{\tau}_u + \boldsymbol{\tau}_a,$$

where $\mathbf{f}_u = [0,0,T]^\top$ and $\boldsymbol{\tau}_u = [\tau_x,\tau_y,\tau_z]^\top$ are thrust and torques. The relationship between $\boldsymbol{\eta} = [T,\tau_x,\tau_y,\tau_z]^\top$ and the control input $\mathbf{u} = [n_1^2,n_2^2,n_3^2,n_4^2]^\top$ is $\boldsymbol{\eta} = B_0\mathbf{u}$:

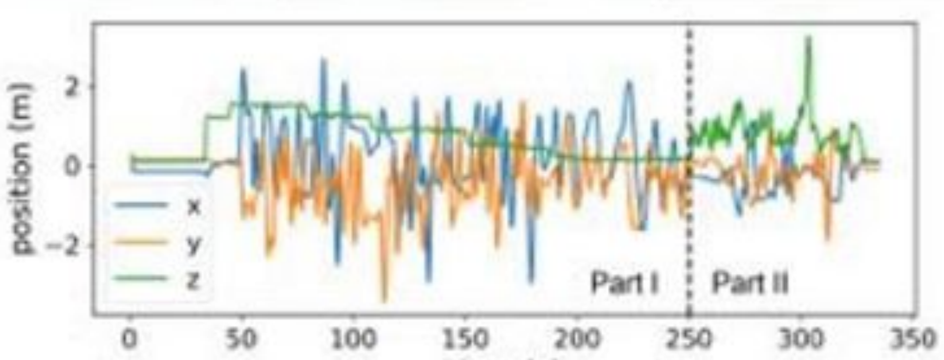
$$B_0 = \begin{bmatrix} c_T & c_T & c_T & c_T \\ 0 & c_T l_{\mathrm{arm}} & 0 & -c_T l_{\mathrm{arm}} \\ -c_T l_{\mathrm{arm}} & 0 & c_T l_{\mathrm{arm}} & 0 \\ -c_Q & c_Q & -c_Q & c_Q \end{bmatrix}.$$

Key difficulty of control: The influence of unknown disturbances $\mathbf{f}_a = [f_{a,x}, f_{a,y}, f_{a,z}]^\top$ and torques $\boldsymbol{\tau}_a = [\tau_{a,x},\tau_{a,y},\tau_{a,z}]^\top$.

Goal: Learn $\mathbf{f}_a, \boldsymbol{\tau}_a$ and then design nonlinear controller with stability guarantee.

Challenge: DNNs can be unstable and generate unpredictable output.

3. Learn high-order interactions



The goal is to estimate $\hat{\mathbf{f}}_a(\boldsymbol{\zeta},\mathbf{u})$, with $\boldsymbol{\zeta},\mathbf{u}$ being the partial states and control inputs, given data from a drone close to the ground.

Spectral normalization: The Lipschitz constant is defined as the smallest value such that $\forall\,\mathbf{x},\mathbf{x}'$:

$$\|f(\mathbf{x}) - f(\mathbf{x}')\|_2/\|\mathbf{x}-\mathbf{x}'\|_2 \le \|f\|_{\mathrm{Lip}}.$$

The spectral normalization goal is

$$\|\hat{\mathbf{f}}_a(\boldsymbol{\zeta},\mathbf{u})\|_{\mathrm{Lip}} \le 1.$$

4. Learning-based discrete nonlinear controller and stability analysis

With the learned dynamics, the desired force is $\mathbf{f}_d = \bar{\mathbf{f}}_d - \hat{\mathbf{f}}_a(\boldsymbol{\zeta},\mathbf{u})$, with $\bar{\mathbf{f}}_d$ from the nominal PD controller. The control synthesis problem here uses a non-affine input for $\mathbf{u}$:

$$B_0\mathbf{u} = \begin{bmatrix}\left(\bar{\mathbf{f}}_d - \hat{\mathbf{f}}_a(\boldsymbol{\zeta},\mathbf{u})\right)\cdot\hat{k} \\ \boldsymbol{\tau}_d\end{bmatrix}; \quad (1) \qquad \mathcal{F}(\mathbf{u}) = B_0^{-1}\begin{bmatrix}\left(\bar{\mathbf{f}}_d - \hat{\mathbf{f}}_a(\boldsymbol{\zeta},\mathbf{u})\right)\cdot\hat{k} \\ \boldsymbol{\tau}_d\end{bmatrix}. \quad (2)$$

We propose the fixed-point iterative method for solving (1): $\mathbf{u}(t) = \mathbf{u}_k = \mathcal{F}(\mathbf{u}_{k-1})$.

Contraction from spectral normalization: If $\hat{\mathbf{f}}_a(\boldsymbol{\zeta},\mathbf{u})$ is L_a-Lipschitz continuous, and $\sigma(B_0^{-1})\cdot L_a < 1$; then $\mathcal{F}(\cdot)$ is a contraction, and $\mathbf{u}_k$ converges to unique solution $\mathbf{u}^* = \mathcal{F}(\mathbf{u}^*)$.

Stability proof under assumptions:

- The desired states $\mathbf{p}_d(t)$, $\dot{\mathbf{p}}_d(t)$, and $\ddot{\mathbf{p}}_d(t)$ are bounded;
- $\mathbf{u}$ updates much faster than position controller;
- The approximation error of $\hat{\mathbf{f}}_a(\boldsymbol{\zeta},\mathbf{u})$ over the compact sets $\mathcal{Z},\mathcal{U}$ is bounded by ϵ_m.

Convergence rate and steady error related to the Lipschitz constant of $\hat{\mathbf{f}}_a(\boldsymbol{\zeta},\mathbf{u})$:

$$\|\mathbf{s}(t)\| \le \|\mathbf{s}(t_0)\|\exp\left(-\frac{\lambda - L_a\rho}{m}(t-t_0)\right) + \frac{\epsilon_m}{\lambda - L_a\rho}, \ \lambda \text{ is the control gain.} \quad (3)$$

5. Experimental results `https://youtu.be/C_K8MkC_SSQ`

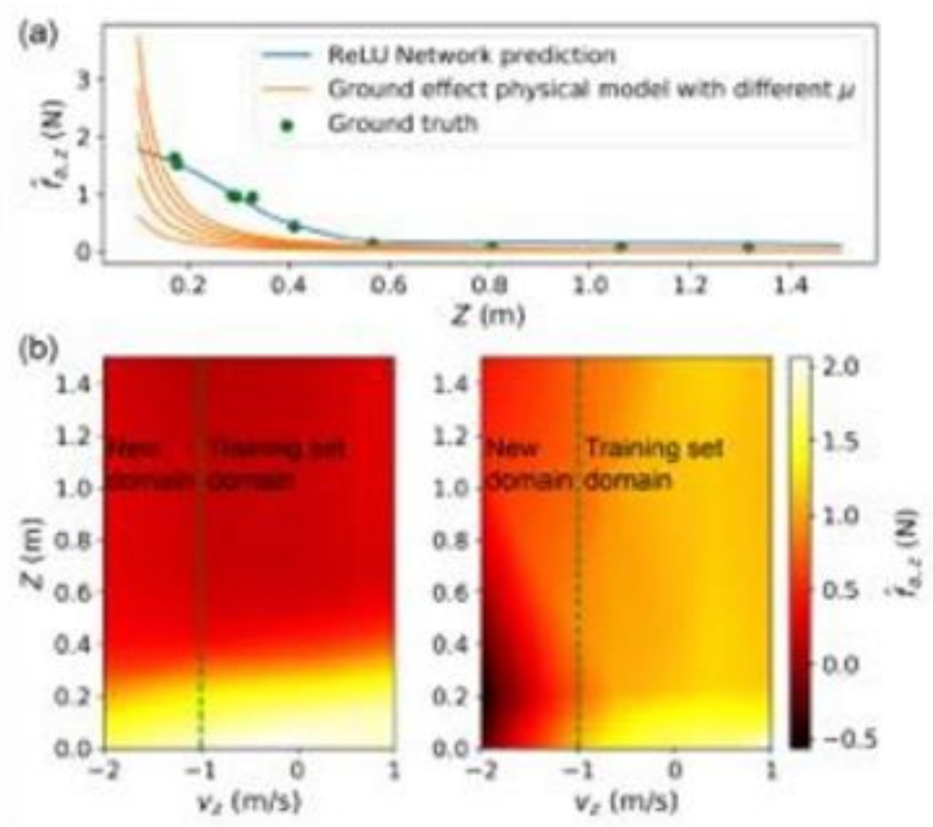
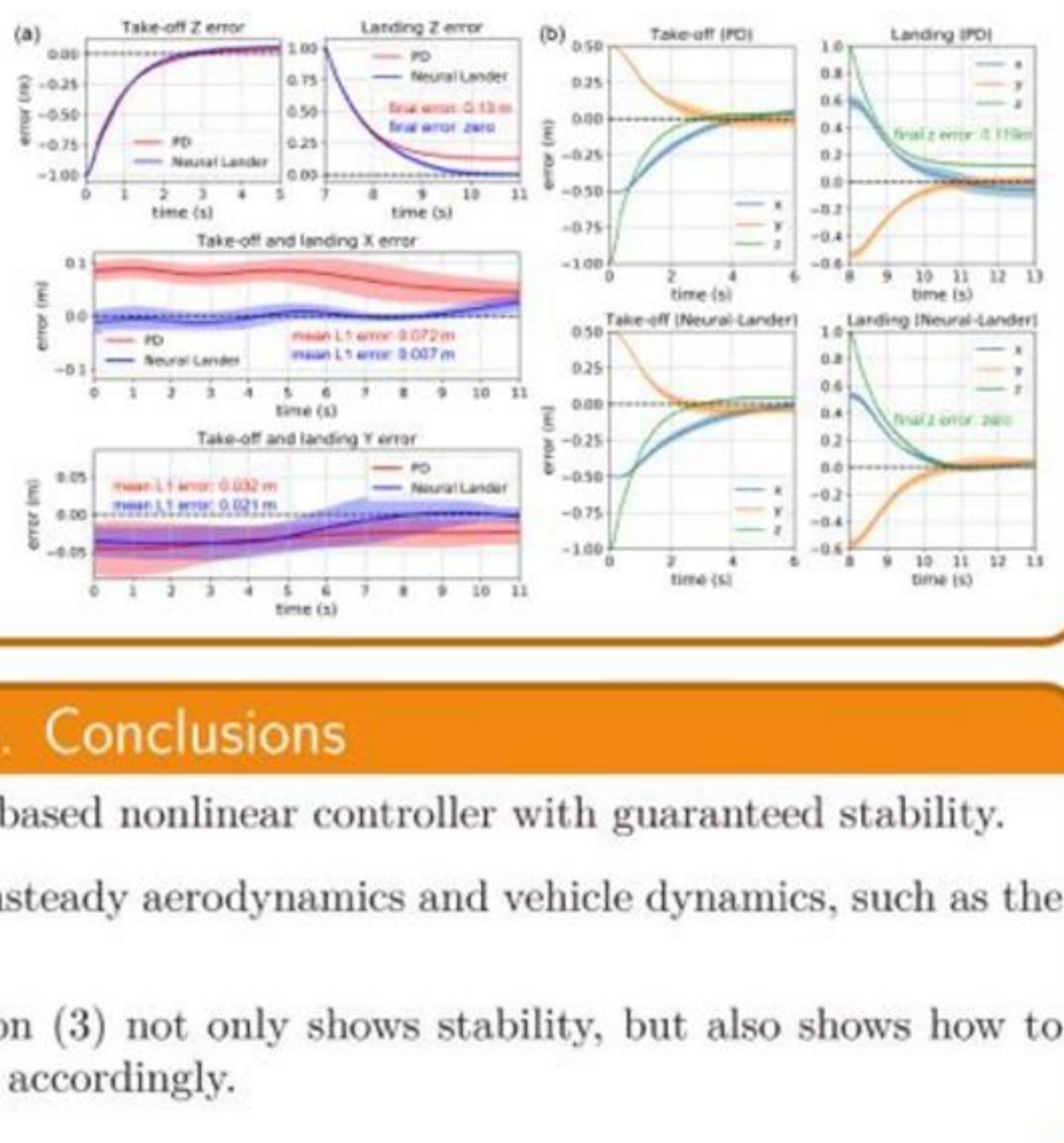


Figure 1: (a) Learned $\hat{f}_{a,z}$ compared to the ground effect model. (b) Heatmaps of learned $\hat{f}_{a,z}$. Other dimensions are fixed. (**Left**) With spectral normalization, $\|f\|_{\mathrm{Lip}} \le 1$. (**Right**) Without spectral normalization, $\|f\|_{\mathrm{Lip}} \le 4.97$.

Learning results. We estimate $\hat{\mathbf{f}}_a$ using a 4-layer ReLU network. We use spectral normalization so that $\|f\|_{\mathrm{Lip}} \le 1$. Visualization of $\hat{f}_{a,z}$ is in Figure 1.

Compared to PD. The baseline and *Neural-Lander* results, in (a) 1D landing and (b) 3D landing, are shown below. *Neural-Lander* could land precisely.

What if $\|f\|_{\mathrm{Lip}}$ too big. We observed some DNN with huge $\|f\|_{\mathrm{Lip}} \le 247$ even crushed the drone.



6. Conclusions

We present *Neural-Lander*, a learning based nonlinear controller with guaranteed stability.

1. Our method can learn coupled unsteady aerodynamics and vehicle dynamics, such as the ground effect and air drag.
2. Our theoretical result in Equation (3) not only shows stability, but also shows how to design DNNs and controller gain accordingly.