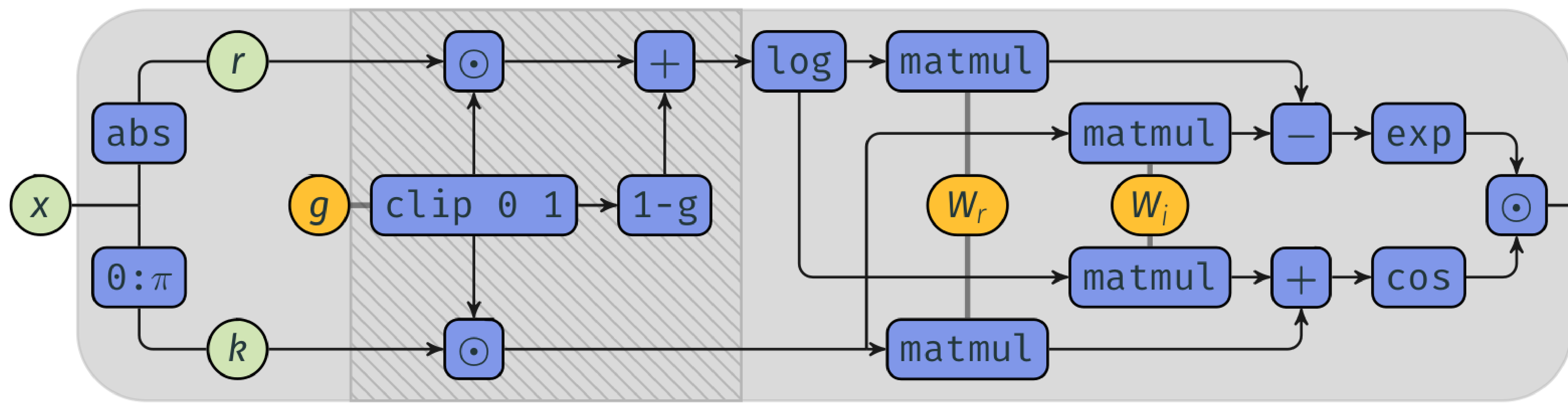
Neural Power Units - Learning to **extrapolate** functions composed of **arithmetic operations**.

Niklas Heim
Václav Šmídl
Tomáš Pevný

Czech Technical University Prague

Neural Power Unit schematic with the **relevance gate** shaded in grey



Neural Arithmetic

Neural Arithmetic (or *Arithmetic Extrapolation*) uses inductive biases to increase the extrapolation performance of neural networks on tasks in which the underlying function is *partially* composed of arithmetic operations.

Neural Power Unit (NPU)

The NPU (inspired by NALU; Trask et al.) uses complex arithmetic to correctly process negative numbers and contains a relevance gate g for more consistent convergence and sparser solutions.

$$y = \exp(W_r \log r - \pi W_i k) \odot \cos(W_i \log r + \pi W_r k),$$

where

$$r = \hat{g} \odot (|x| + \epsilon) + (1 - \hat{g}),$$

$$k_i = \begin{cases} 0 & x_i \geq 0 \\ \hat{g}_i & x_i < 0 \end{cases},$$

$$\hat{g}_i = \min(\max(g_i, 0), 1),$$

with inputs x, and trainable parameters W_r, W_i, g.
In the **RealNPU** we fix W_i to zero to obtain a highly transparent model.

Prior art

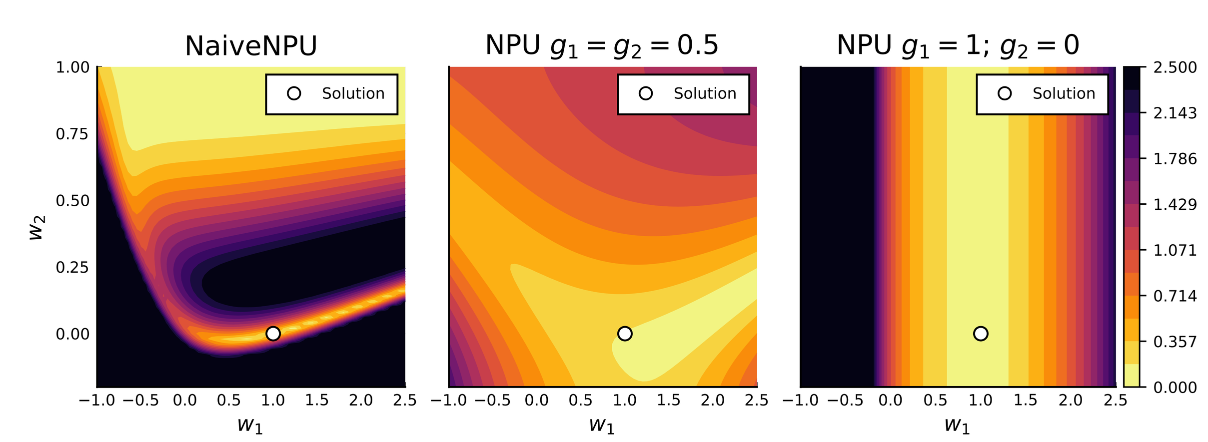
- *NALU* (Trask et al.): Can perform $(+, \times, \div)$, but cannot handle negative numbers and has convergence problems.
- *NMU & NAU* (Madsen & Johansen): Can perform addition and multiplication, but no division.

Our Contributions

- Learning **multiplication**, **division** and fractional **power functions**
- Correct processing of **negative inputs**
- **Relevance gating** for reliable convergence and sparse solutions
- A **transparent** model for applications like equation discovery (RealNPU)

Relevance gate

On a toy task of learning $f(x, y) = x$ we demonstrate that the *NaiveNPU* (without the relevance gate) has a zero gradient norm in large parts of parameter space (left plot). In the NPU (with the gate) the gradient surface becomes much more informative.



Find all the details in our paper or contact us via heimnikl@fel.cvut.cz.

Paper

Code

Acknowledgement

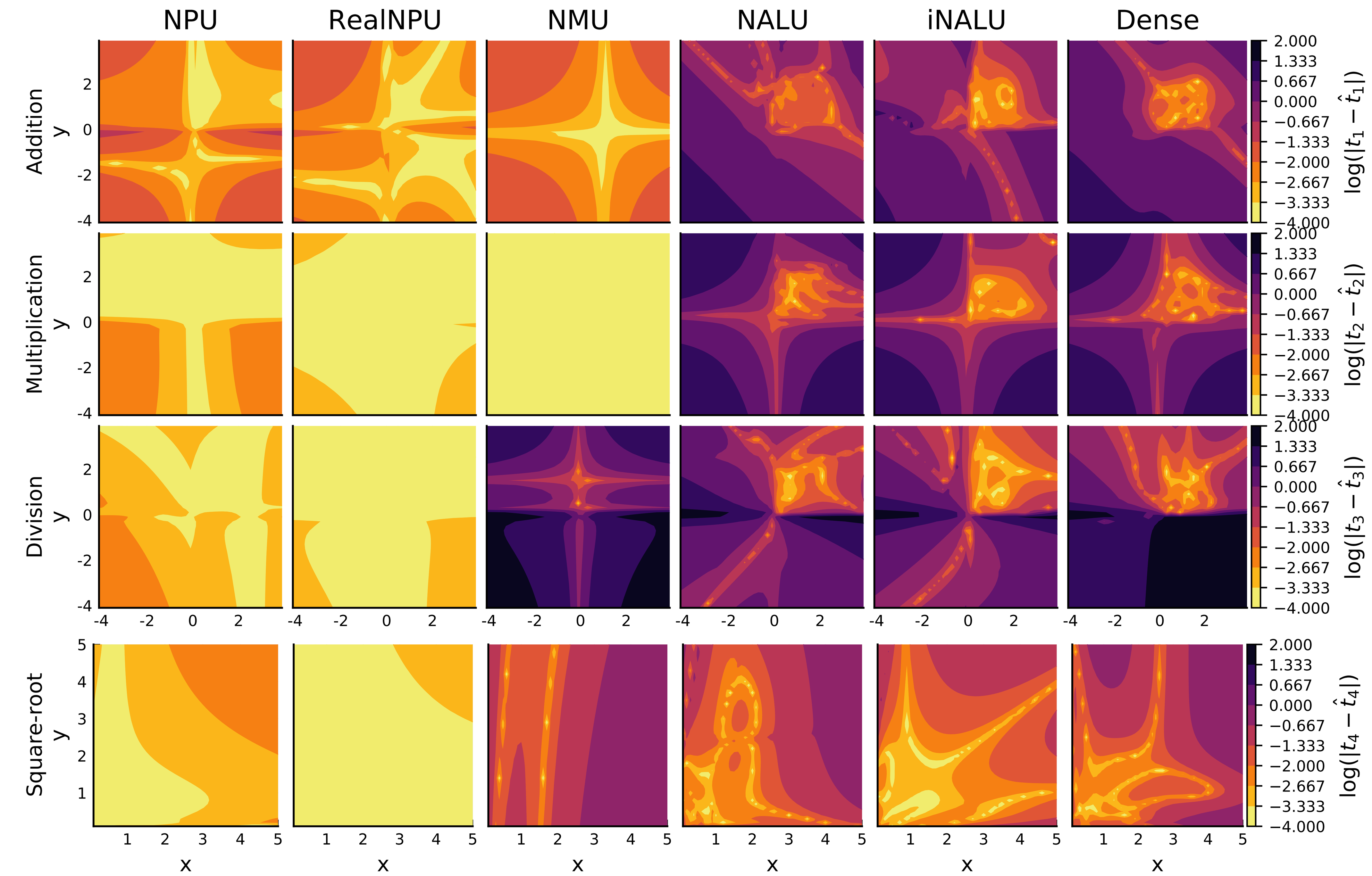
This work has been supported by the OP VVV funded project Research Center for Informatics; reg. No.: CZ.02.1.01/0.0./0.0./16_019/0000765.

Extrapolation experiment

Prediction error for different models trained to learn f on training samples $(x, y) \sim U(0.01, 2)$.

$$f(x, y) = (x + y,\ xy,\ \frac{x}{y},\ \sqrt{x}\ )^T$$

Bright colors indicate low error. All models learn the problem within the training range, but only the NPU manages to learn all operations including the fractional power $\sqrt{x}$.



Towards Equation Discovery

Our paper sketches how to use the **RealNPU** for equation discovery of ODEs with fractional powers. The RealNPU (Layer 1) on the right discovers a product of fractional powers in the first row of weights. The NAU (Layer 2) looks very similar to the parameter matrix of the *fSIR* model (Taghvaei et al.) on the left.

$$\begin{bmatrix} \dot{S} \\ \dot{I} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} -\beta & 0 & \eta \\ \beta & -\alpha & 0 \\ 0 & \alpha & \eta \end{bmatrix} \begin{bmatrix} I^\gamma S^\kappa \\ I \\ R \end{bmatrix}$$