DATE 23
DESIGN, AUTOMATION & TEST IN EUROPE
17 – 19 April 2023 · Antwerp, Belgium
The European Event for Electronic System Design & Test

Highly-Bespoke Robust Printed Neuromorphic Circuits

Haibin Zhao, Brojogopal Sapui, Michael Hefenbrock, Zhidong Yang, Michael Beigl and Mehdi B. Tahoori

Karlsruhe Institute of Technology (KIT)

Outline

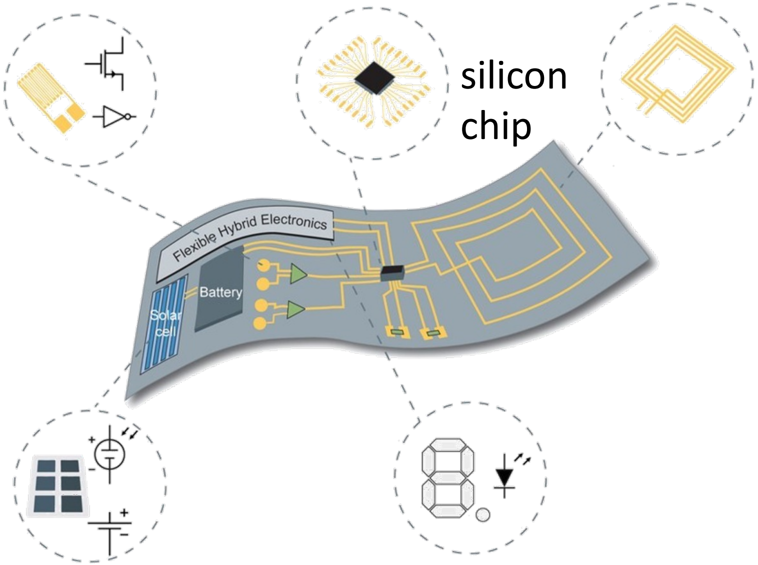
- **Printed Electronics**
- **Printed Neuromorphic Circuits**
- **Learnable Nonlinear Circuits**
- **Variation-Aware Training of pNNs**
- **Experiment**
- **Conclusion**

Printed Electronics – Overview

printed sensors

printed antennas

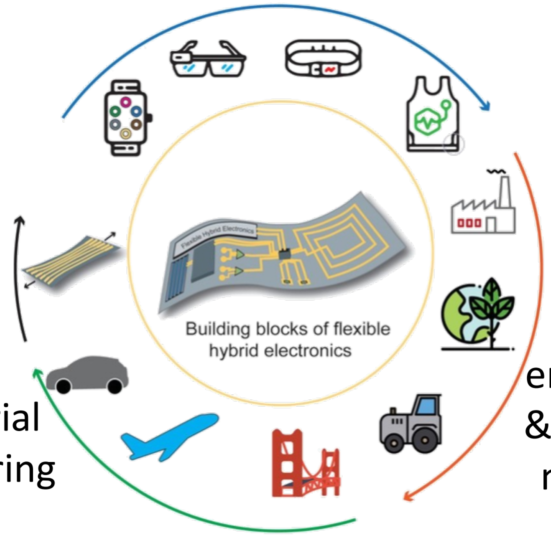
silicon chip



printed energy storage & harvester

printed displays

wearable devices



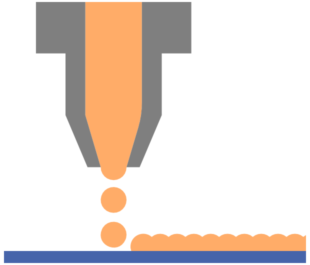
industrial monitoring

environment & agriculture monitoring

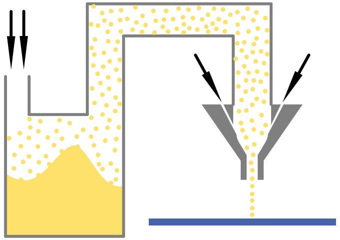
Source: Khan Y, *et al*. “A New Frontier of Printed Electronics: Flexible Hybrid Electronics”. *Advanced Materials,* 2020

Printed Electronics – Customizability

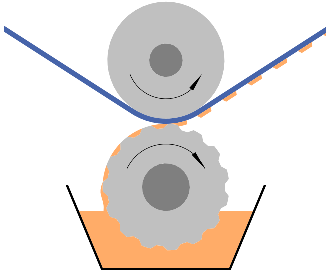
- **High customizability**



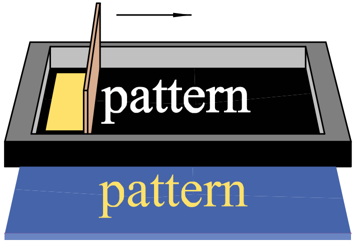
Inkjet printing



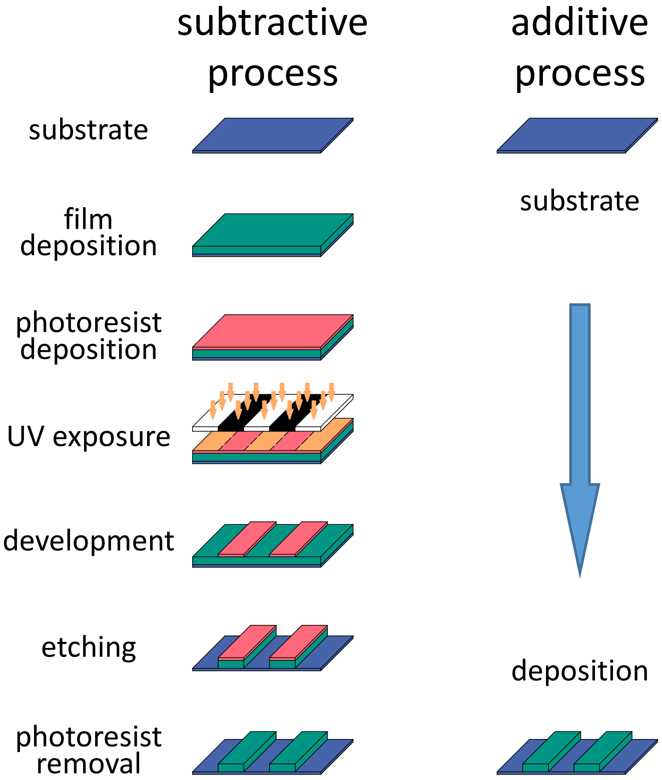
Aerosol jet printing



Gravure printing

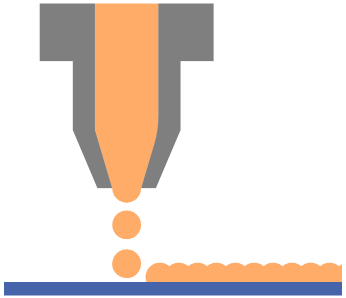


Screen printing



Printed Electronics – Variation

- **Printing variation**
  - **Definition**
    - **discrepancy between actual printed values and desired values**
  - **Reason**
    - **Printing technologies**
      - **Minimal printing resolution determined by, e.g., smallest droplet in ink-jet printing**
    - **Physical properties of**
      - **Functional inks**
      - **Substrates**
    - **Environmental conditions**



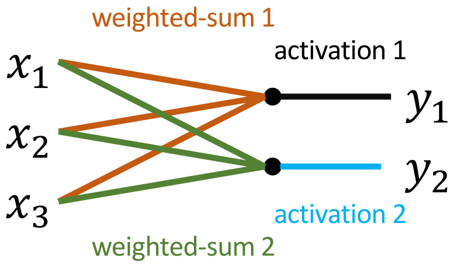
Inkjet printing

Outline

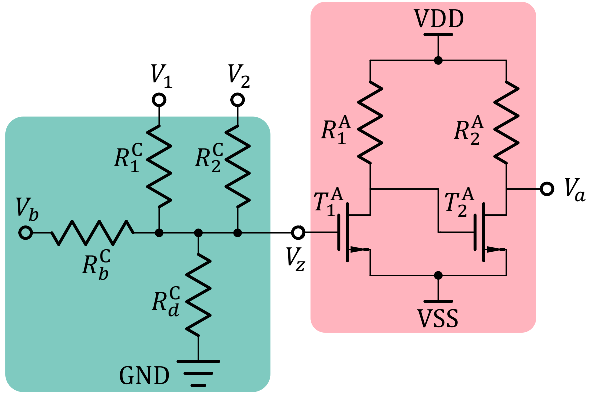
- Printed Electronics
- **Printed Neuromorphic Circuits**
- Learnable Nonlinear Circuits
- Variation-Aware Training of pNNs
- Experiment
- Conclusion

Printed Neuromorphic Circuit – Motivation

- **Conventional digital NNs are infeasible for PE**
  - **Large feature size**
  - **Low integration density**
  - **Low device count**



Components	Number of transistors	
	4-bit digital NN	Analog NN
Input converter	185	-
Weighted-sum	265	≤ 4
Activation	10	2



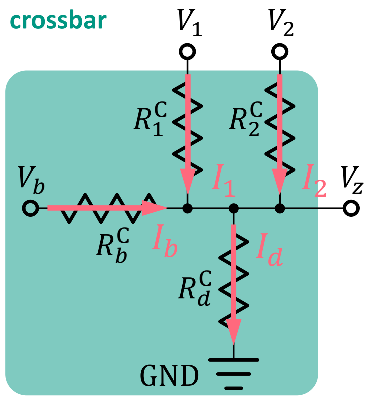
- **Analog neuromorphic circuits were developed**

Printed Neuromorphic Circuit – Primitives

- **Resistor crossbar for weighted-sum**

$$V_z = \frac{g_1}{G}V_1 + \frac{g_2}{G}V_2 + \frac{g_b}{G}V_b$$

where $g_i = \frac{1}{R_i}$**,** G **is the sum of** g_i**,** $V_b \equiv 1V$



- **Output** V_z **is the weighted-sum of input voltages** V_i
- **Weights and bias represented by conductance values**

Printed Neuromorphic Circuit – Primitives

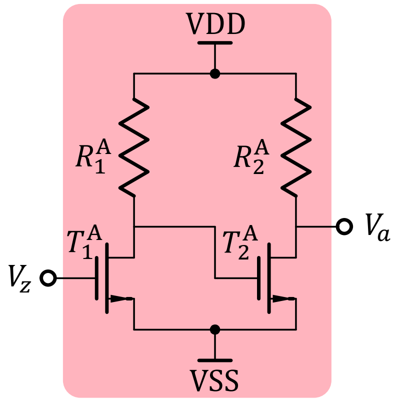
- **Printed tanh-like (ptanh) activation circuit**
  - **Tanh-like characteristic curve**

$$V_a = \mathrm{ptanh}(V_z) = \eta_1^{\mathrm{A}} + \eta_2^{\mathrm{A}} \cdot \tanh\left(\left(V_z - \eta_3^{\mathrm{A}}\right) \cdot \eta_4^{\mathrm{A}}\right)$$

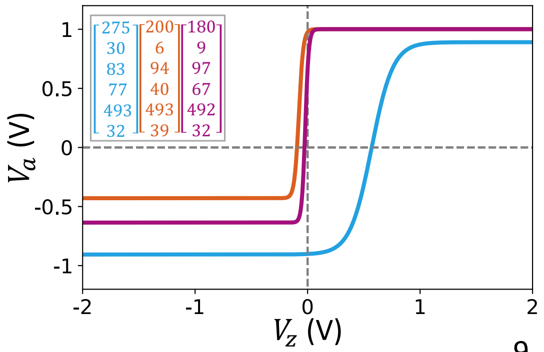
  - $\boldsymbol{\eta}^{\mathrm{A}} = \left[\eta_1^{\mathrm{A}}, \eta_2^{\mathrm{A}}, \eta_3^{\mathrm{A}}, \eta_4^{\mathrm{A}}\right]$ **is the auxiliary parameter**
    - **translates and scales the** $\tanh$ **function**
    - **determined by circuit components**

$$\boldsymbol{\omega}^{\mathrm{A}} = [R_1^{\mathrm{A}}, R_2^{\mathrm{A}}, \underbrace{W_1^{\mathrm{A}}, L_1^{\mathrm{A}}}_{T_1^{\mathrm{A}}}, \underbrace{W_2^{\mathrm{A}}, L_2^{\mathrm{A}}}_{T_2^{\mathrm{A}}}]$$

ptanh circuit



Printed Tanh-like Function (ptanh)



Printed Neuromorphic Circuit – Primitives

- **Negative weight circuit**
  - **Recall – resistor crossbar**

$$\frac{g_i}{G} V_i =: w_i V_i$$

    - **Problem: weight (conductance) is only positive**
    - **Solution: invert V_i and still use positive conductance**

$$(-w_i) \cdot V_i = w_i \cdot (-V_i) \leftarrow w_i \cdot \mathrm{neg}(V_i)$$

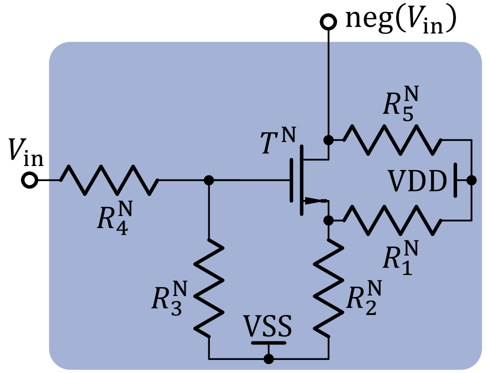
where

$$\mathrm{neg}(V_\mathrm{in}) = -\left(\eta_1^\mathrm{N} + \eta_2^\mathrm{N} \cdot \tanh\left(\left(V_z - \eta_3^\mathrm{N}\right) \cdot \eta_4^\mathrm{N}\right)\right)$$

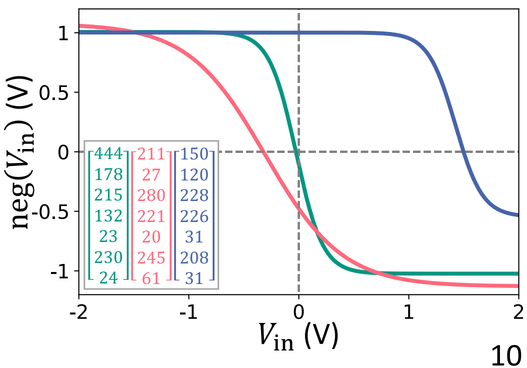
  - $\boldsymbol{\eta}^\mathrm{N} = \left[\eta_1^\mathrm{N}, \eta_2^\mathrm{N}, \eta_3^\mathrm{N}, \eta_4^\mathrm{N}\right]$ **is the auxiliary parameter**
    - **determined by circuit components**

$$\boldsymbol{\omega}^\mathrm{N} = \left[R_1^\mathrm{N}, R_2^\mathrm{N}, R_3^\mathrm{N}, R_4^\mathrm{N}, R_5^\mathrm{N}, W^\mathrm{N}, L^\mathrm{N}\right]$$

negative weight circuit



Negative Weight Function (neg)



Printed Neuromorphic Circuit – Network

- **Resistor crossbar**

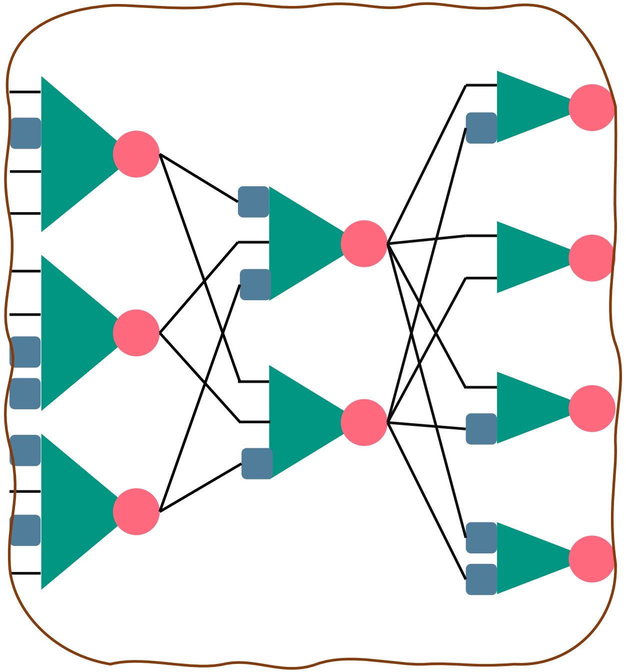
$$V_z = \frac{g_1}{G} V_1 + \frac{g_2}{G} V_2 + \frac{g_b}{G} V_b$$

- **Printed tanh-like circuit**

$$V_a = \eta_1^{\mathrm{A}} + \eta_2^{\mathrm{A}} \cdot \tanh\left(\left(V_z - \eta_3^{\mathrm{A}}\right) \cdot \eta_4^{\mathrm{A}}\right)$$

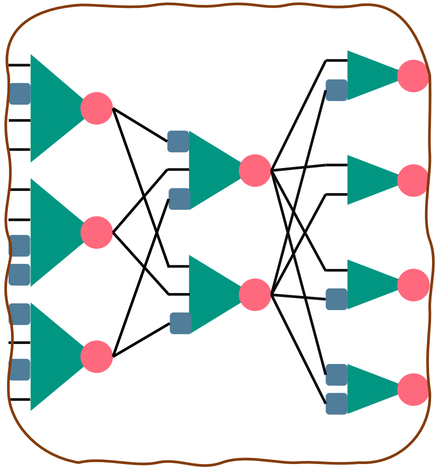
- **Negative weight circuit**

$$\mathrm{neg}(V_{\mathrm{in}}) = -\left(\eta_1^{\mathrm{N}} + \eta_2^{\mathrm{N}} \cdot \tanh\left(\left(V_z - \eta_3^{\mathrm{N}}\right) \cdot \eta_4^{\mathrm{N}}\right)\right)$$



Printed Neuromorphic Circuit – Design

Printed Neuromorphic Circuit



machine learning

model

Printed Neural Network

input V learnable g_i

$\text{neg}(\cdot)$

$$\boxed{w_i = \frac{g_i}{G}}$$

$$V_z = w_1 V_1 + w_2 V_2 + b$$

activation

$$V_a = \text{ptanh}(V_z)$$

well-trained g_i

fabrication

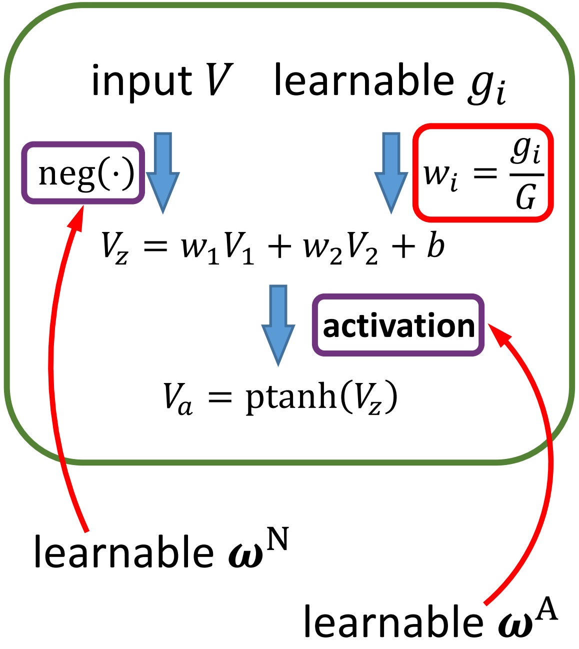
Outline

- Printed Electronics
- Printed Neuromorphic Circuits
- **Learnable Nonlinear Circuits**
- Variation-Aware Training of pNNs
- Experiment
- Conclusion

Learnable Nonlinear Circuits – Motivation

- **Previous work**
  - **Only conductance in crossbar are learnable**
  - **Nonlinear circuits are predefined**
  - **High customization provided by PE is not leveraged**
- **This work**
  - **Bespoke design of nonlinear circuits to the target tasks**
  - **Variation-aware design considering the components in nonlinear circuits**

Printed Neural Network



Learnable Nonlinear Circuits – Challenges

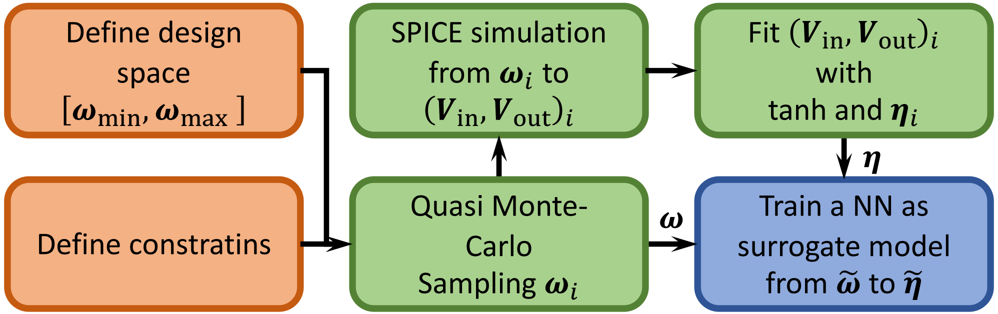
- **To optimize physical quantities ω^{N} and ω^{A}**
  - **Involve ω^{N} and ω^{A} into pNN**
  - **So far** $V_a = \eta_1^{\mathrm{A}} + \eta_2^{\mathrm{A}} \cdot \tanh\left(\left(V_z - \eta_3^{\mathrm{A}}\right) \cdot \eta_4^{\mathrm{A}}\right)\right)$

    $$\mathrm{neg}(V_{\mathrm{in}}) = -\left(\eta_1^{\mathrm{N}} + \eta_2^{\mathrm{N}} \cdot \tanh\left(\left(V_z - \eta_3^{\mathrm{N}}\right) \cdot \eta_4^{\mathrm{N}}\right)\right)\right)$$
  - **Challenge**
    - **transformation from ω to η**
    - **differentiable model for gradient-based optimization**
  - **Solution: ML model (surrogate nonlinear circuit model)**
- **To consider the constraints on ω for**
  - **Desired tanh-like shape**
  - **…**

$$\boldsymbol{\omega}$$
$$\downarrow$$
$$\boldsymbol{\omega}_{\mathrm{feasible}}$$
$$\downarrow$$
$$\boldsymbol{\eta}$$
$$\downarrow$$
$$\mathrm{neg}(\cdot)/\mathrm{ptanh}(\cdot)$$
$$\downarrow$$
pNN

Learnable Nonlinear Circuits – Methodology

- **Neural network based surrogate nonlinear circuit model**
  - **Goal: differentiable transformation**
    from physical quantities ω to auxiliary parameter η
  - **Methodology**

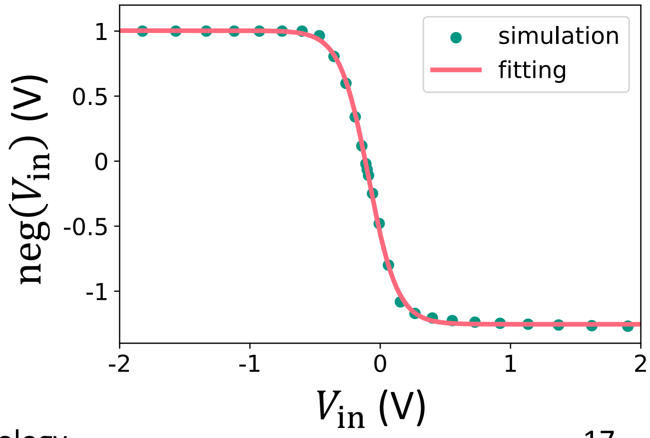


Learnable Nonlinear Circuits – Methodology

- **Methodology**
  - **Define design space & constraints**
  - **Sample $\boldsymbol{\omega}_i$ from suitable design space,** $i = 1, \dots, 10000$
  - **Obtain characteristic curves from SPICE simulation**
  - **Obtain $\boldsymbol{\eta}_i$ by fitting curves**
  - **Dataset**

    input $\boldsymbol{\omega}$ and target output $\boldsymbol{\eta}$
  - **Train NNs for $\boldsymbol{\omega} \mapsto \boldsymbol{\eta}$**

	R_1^{N} (Ω)	R_2^{N} (Ω)	R_3^{N} $(k\Omega)$	R_4^{N} $(k\Omega)$	R_5^{N} $(k\Omega)$	W^{N} (μm)	L^{N} (μm)
minimal	10	5	10	8	10	200	10
maximal	500	250	500	400	500	800	70
inequality	$R_1^{\mathrm{N}} > R_2^{\mathrm{N}}$		$R_3^{\mathrm{N}} > R_4^{\mathrm{N}}$		-	-	-



Learnable Nonlinear Circuits – Methodology

- **Integration of surrogate models into pNNs with constraints**
  - **Constraints**
    - **MIN-MAX constraints** $\omega \in [\omega_{\min}, \omega_{\max}]$
      - **unconstrained optimization variable -> sigmoid -> denormalization**

$$\mathfrak{w} \in \mathbb{R}$$
$$\mathrm{sigmoid}(\mathfrak{w}) \in (0,1)$$
$$\underbrace{\mathrm{sigmoid}(\mathfrak{w})(\omega_{\max} - \omega_{\min}) + \omega_{\min}}_{\omega} \in (\omega_{\min}, \omega_{\max})$$

    - **Inequality constraints** $R_1 > R_2$**, with** $R_1 \in [(R_1)_{\min}, (R_1)_{\max}]$**,** $R_2 \in [(R_2)_{\min}, (R_2)_{\max}]$
      - R_1**: same as MIN-MAX constraint**
      - R_2**: introducing intermediate variable** $\mathfrak{w}$ **-> sigmoid** $\xrightarrow{R_1}$ R_2

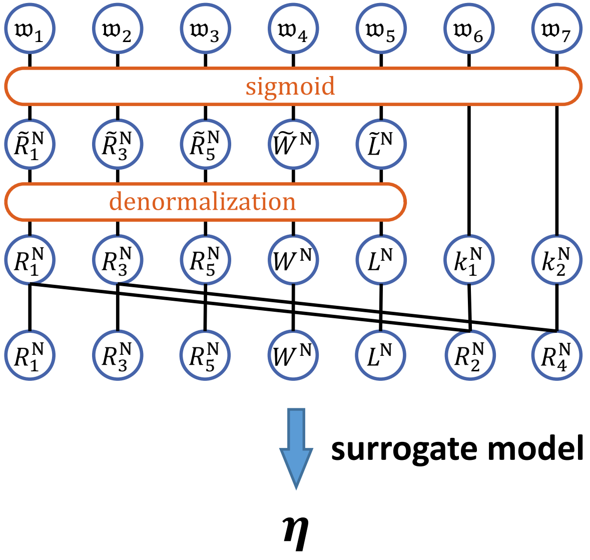
$$\mathfrak{w} \in \mathbb{R}$$
$$\mathrm{sigmoid}(\mathfrak{w}) \in (0,1)$$
$$\underbrace{\mathrm{sigmoid}(\mathfrak{w})(R_1 - (R_2)_{\min}) + (R_2)_{\min}}_{R_2} \in \left((R_2)_{\min}, R_1\right)$$

Learnable Nonlinear Circuits – Methodology

- **Integration of surrogate models into pNNs with constraints**
  - **Example of negative weight circuit**
    - **Constraints**
      - **MIN-MAX constraints on each variable**
      - **Inequality constraints**

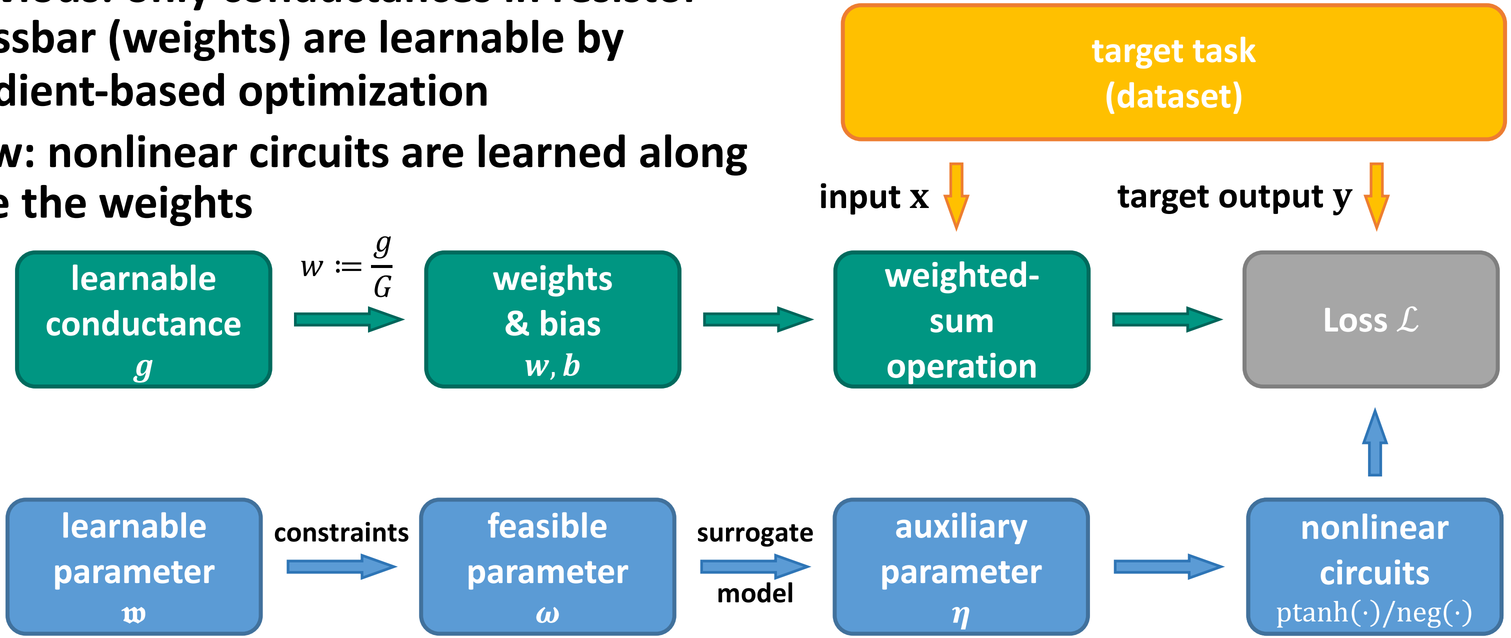
        $R_1^{\mathrm{N}} > R_2^{\mathrm{N}}$ **and** $R_3^{\mathrm{N}} > R_4^{\mathrm{N}}$
      - **learnable parameter corresponding to**

        $\left[R_1^{\mathrm{N}}, R_3^{\mathrm{N}}, R_5^{\mathrm{N}}, W^{\mathrm{N}}, L^{\mathrm{N}}, k_1^{\mathrm{N}}, k_2^{\mathrm{N}}\right]$
    - **Sigmoid -> denormalization ->** $R_1^{\mathrm{N}}, R_3^{\mathrm{N}}, R_5^{\mathrm{N}}, W^{\mathrm{N}}, L^{\mathrm{N}}$
    - **Intermediate** $\mathfrak{w}$ **-> sigmoid ->** R_2^{N} **and** R_4^{N}
    - **Surrogate model** $\boldsymbol{\omega} \mapsto \boldsymbol{\eta}$



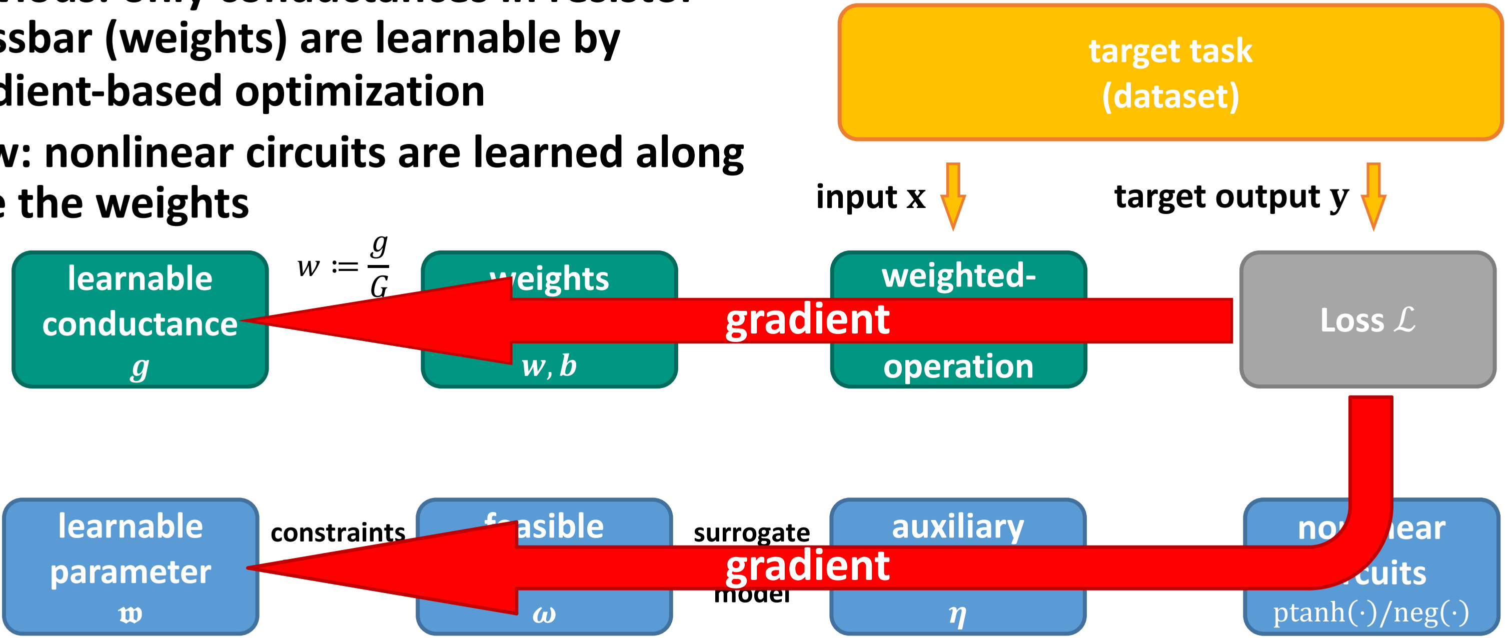
Learnable Nonlinear Circuits – Summary

- **pNN with learnable nonlinear circuits**
  - **Previous: only conductances in resistor crossbar (weights) are learnable by gradient-based optimization**
  - **Now: nonlinear circuits are learned along side the weights**



Learnable Nonlinear Circuits – Summary

- **pNN with learnable nonlinear circuits**
  - **Previous: only conductances in resistor crossbar (weights) are learnable by gradient-based optimization**
  - **Now: nonlinear circuits are learned along side the weights**

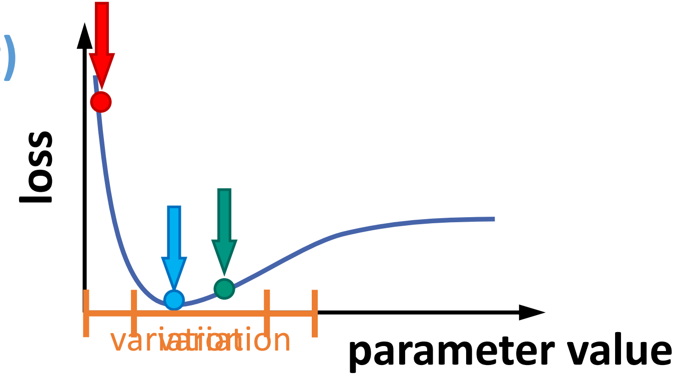


Outline

- Printed Electronics
- Printed Neuromorphic Circuits
- Learnable Nonlinear Circuits
- **Variation-Aware Training of pNNs**
- Experiment
- Conclusion

Variation-Aware Training – Motivation

- **Influence of printing variation**
  - **Discrepancy of component values**
  - **Discrepancy of weights/bias/activation functions...**
  - **Discrepant (wrong) output than desired output**
- **Variation-aware training of pNNs**
  - **Variation-unaware training (nominal training)**
    - **Possibly bad case with high loss**
  - **Variation-aware training**
    - **Robust against printing variation**



Variation-Aware Training – Objective Function

- **Loss function (objective function)**
  - **Variation-unaware training (nominal training)**

$$\text{Loss} = \mathcal{L}(\boldsymbol{g}, \boldsymbol{\mathfrak{w}}, \mathbf{x}, \mathbf{y})$$

    - $\mathcal{L}(\cdot)$: **cross entropy loss (performance of the classification task)**
    - $\boldsymbol{g}$: **learnable conductance (weights & bias)**
    - $\boldsymbol{\mathfrak{w}}$: **learnable parameter for nonlinear circuit**
    - $\mathbf{x}, \mathbf{y}$: **input data & target output in dataset**

$$\begin{aligned}\text{Loss} &= \mathbb{E}_{\boldsymbol{\varepsilon}}\{\mathcal{L}(\boldsymbol{g}, \boldsymbol{\mathfrak{w}}, \boldsymbol{\varepsilon}, \mathbf{x}, \mathbf{y})\} \\ &= \int_{\boldsymbol{\varepsilon}} \mathcal{L}(\boldsymbol{g}, \boldsymbol{\mathfrak{w}}, \boldsymbol{\varepsilon}, \mathbf{x}, \mathbf{y}) \mathrm{d}\boldsymbol{\varepsilon} \\ &\approx \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}(\boldsymbol{g}, \boldsymbol{\mathfrak{w}}, \boldsymbol{\varepsilon}', \mathbf{x}, \mathbf{y})\end{aligned}$$

  - **Variation-aware training**
    - **Stochastic modeling of parameter $\boldsymbol{\varepsilon}_g \odot \boldsymbol{g}$ and $\boldsymbol{\varepsilon}_\omega \odot \boldsymbol{\omega}$, where**
      - **Each element in $\boldsymbol{\varepsilon}$ follows e.g., $\mathcal{U}[0.9,1.1]$ to represent $\pm 10\%$ printing variation**
      - **$\odot$: element-wise product**
    - **Modified training objective: expected loss w.r.t. printing variation $\boldsymbol{\varepsilon}$**
    - **Monte-Carlo approximation for integration**

Outline

- Printed Electronics
- Printed Neuromorphic Circuits
- Learnable Nonlinear Circuits
- Variation-Aware Training of pNNs
- **Experiment**
- Conclusion

Experiment – Setup

- 13 benchmark datasets
- Learnable nonlinear circuit (LNC)
  - Baseline: fixed nonlinear circuit
- Variation-aware training (VAT)
  - $\pm 5\%$ for low printing variation
  - $\pm 10\%$ for high printing variation
  - Baseline: variation-unaware training
- Evaluation metrics
  - Accuracy of the task (expectation accuracy)
  - Robustness against variation (standard deviation of accuracy)
- Ablation study to analyze the contribution of both approaches

Experiment – Result

- **Result**

Expectation and Variance of Accuracy

Learnable non-linear circuit	Variation-aware training	ϵ_{test}	
		5%	10%
✓	✓	0.809 ± 0.023	0.786 ± 0.029
✓	✗	0.752 ± 0.095	0.697 ± 0.130
✗	✓	0.731 ± 0.053	0.691 ± 0.080
✗	✗	0.678 ± 0.085	0.626 ± 0.118

Overall Improvement

(learnable nonlinear circuits + variation-aware training)

	ε = 5%	ε = 10%
accuracy	19%	26%
robustness	73%	75%

Contribution in Accuracy Improvement

	5%	10%
LNC	58%	52%
VAT	42%	48%

Contribution in Robustness Improvement

	5%	10%
LNC	↓	↓
VAT	≈100%	≈100%

- **LNC and VAT provide comparable accuracy improvement**
- **Almost all the robustness improvement is provided by VAT**

LNC: learnable nonlinear circuit
VAT: variation-aware training

Outline

- Printed Electronics
- Printed Neuromorphic Circuits
- Learnable Nonlinear Circuits
- Variation-Aware Training of pNNs
- Experiment
- **Conclusion**

Conclusion

- Printed electronics provides complementary advantages
  - Compared to traditional silicon-based VLSI technologies
- Low device count, large feature sizes, large latencies
  - Constraints for printed circuits
- Printed analog neuromorphic circuits
  - Analog computing to reduce device count
- Highly-bespoke robust printed neuromorphic circuits
    - 20% - 25% accuracy improvement depending on 5% - 10% variation
    - 75% robustness improvement
  - Learnable nonlinear circuits (LNC) for bespeaking target tasks
    - Provides comparable accuracy improvement as VAT
    - Slightly reduces robustness
  - Variation-aware training (VAT) in both crossbar and nonlinear circuits
    - Provides comparable accuracy improvement as LNC
    - Provides almost all the improvement in robustness

Thank you for your attention

Haibin Zhao, Brojogopal Sapui, Michael Hefenbrock, Zhidong Yang
Michael Beigl, Mehdi B. Tahoori
Karlsruhe Institute of Technology (KIT)