Neural CRNs

A Learning Implementation in Chemical Reaction Networks

Rajiv Nagipogu

Reif Lab

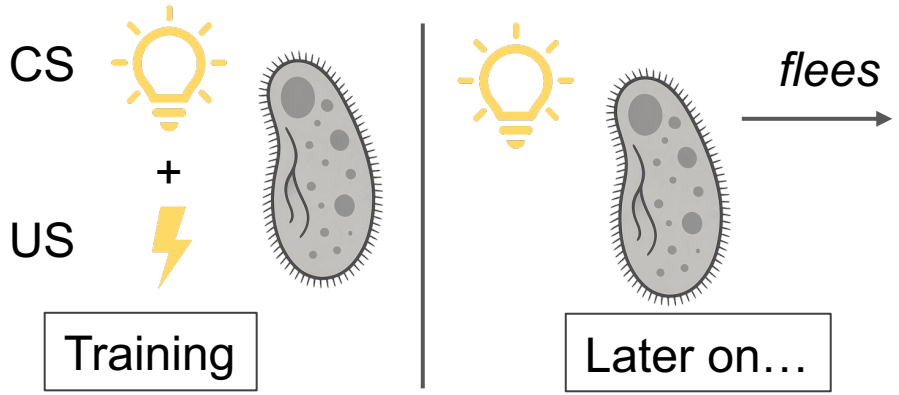
Duke Computer Science

Duke

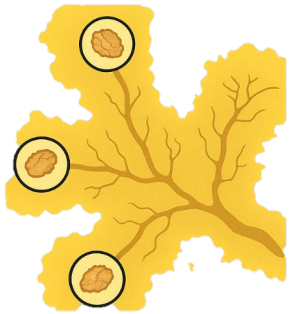
Towards adaptive biochemical circuits *in situ*

Learning and adaptation are inherent in biological systems

Paramecia could be classically conditioned



Slime molds display intelligent foraging behaviors



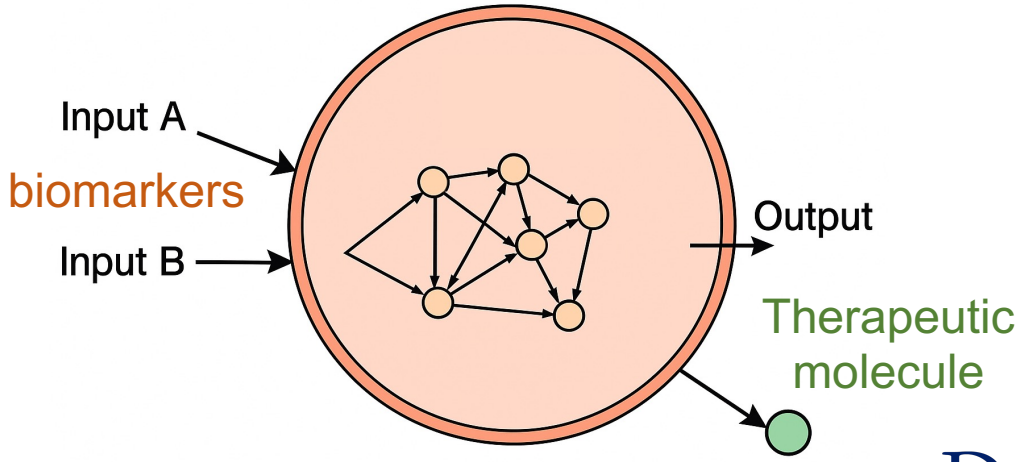
…

Adaptive behavior in synthetic circuits:

→ To monitor dynamic biochemical environments

→ Decision-making in artifical cells

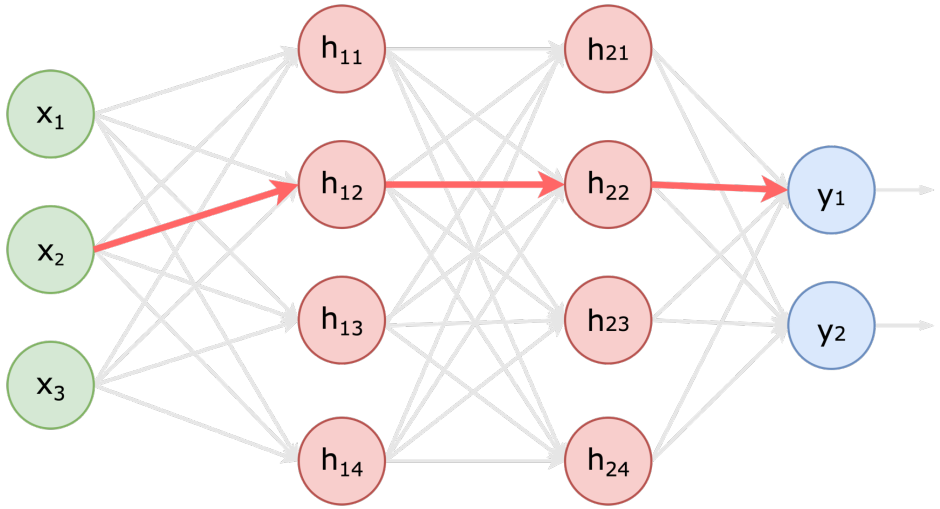
→ Adaptive therapeutics, diagnostics



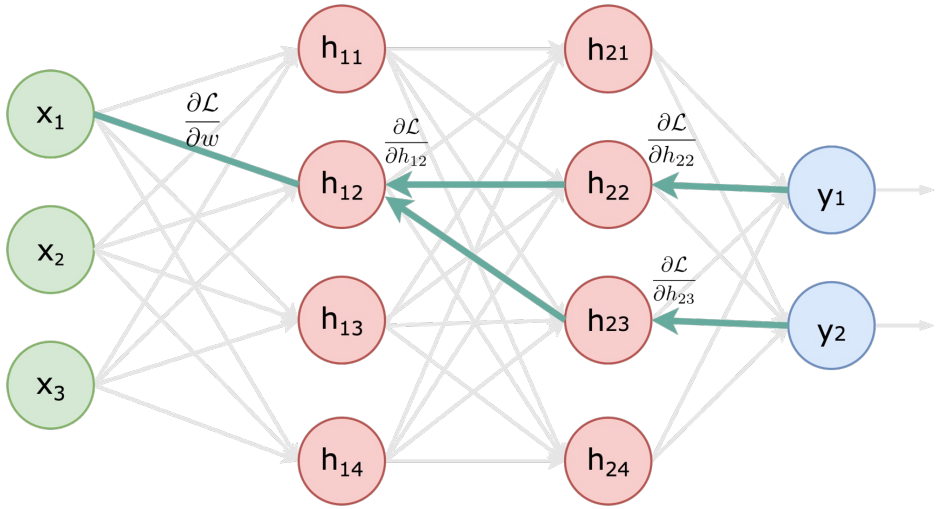
Duke

Learning is typically associated with neural networks

Compact representations and gradient descent-based learning protocols



Feedforward phase



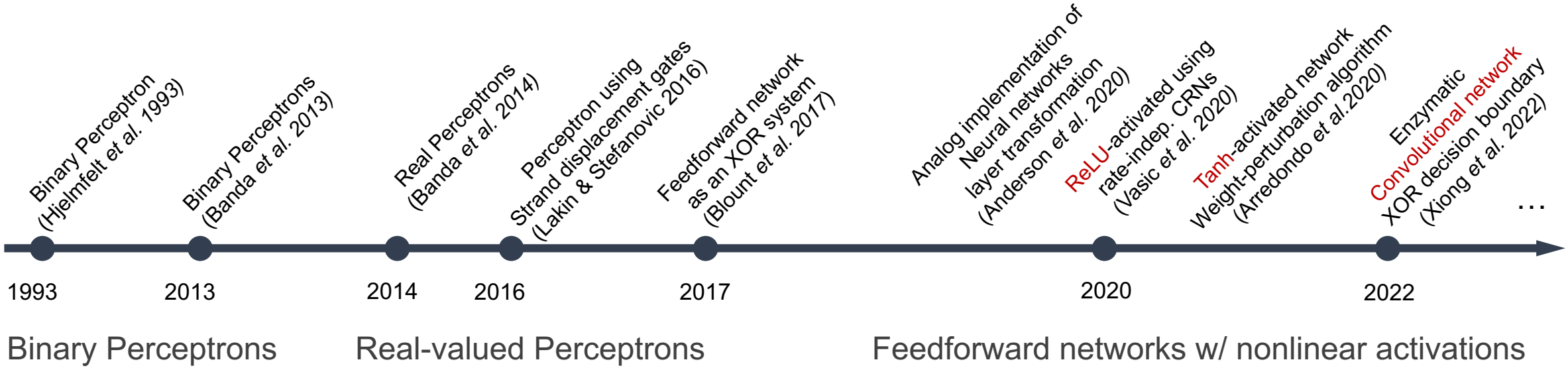
Feedback phase

Learning through *adjoint state backpropagation*

Using chain rule of partial differentiation

Duke

Chemical implementations of neural networks

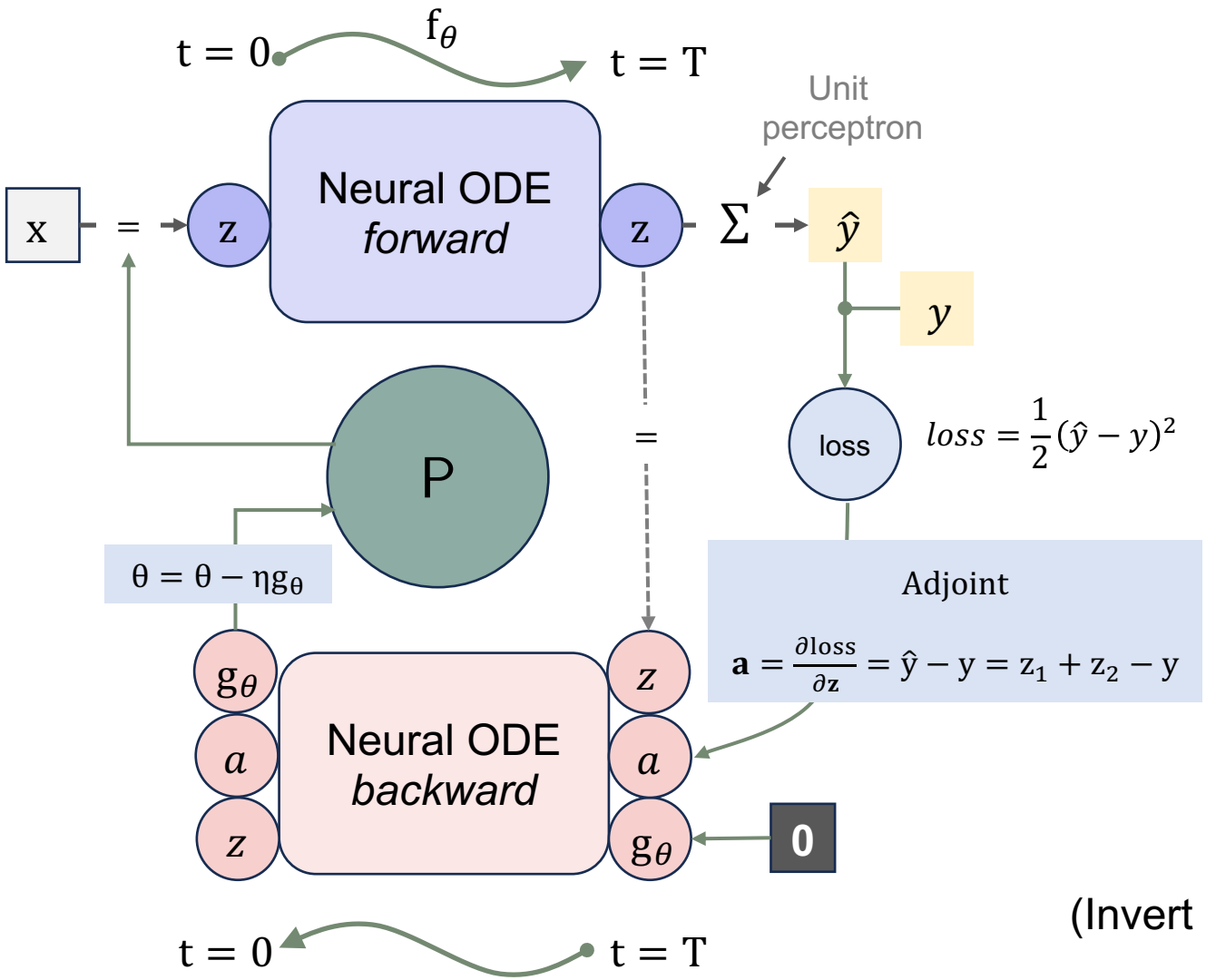


- Composing chemical implementations of constituent algebraic computations
- Requires adding multiple ‘stop-and-play’ mechanisms

[On reaction network implementations of neural networks]

Duke

Learning in Neural ODEs: The adjoint sensitivity method



Integrated in forward time

$$\frac{d\mathbf{z}}{dt} = f_\theta$$

$$\frac{d\mathbf{g}_\theta}{dt} = -\mathbf{a}^T \frac{\partial f_\theta}{\partial \theta}$$
$$\frac{d\mathbf{z}}{dt} = -f_\theta$$
$$\frac{d\boldsymbol{a}}{dt} = -\boldsymbol{a}^T \frac{\partial f_\theta}{\partial \mathbf{z}}$$

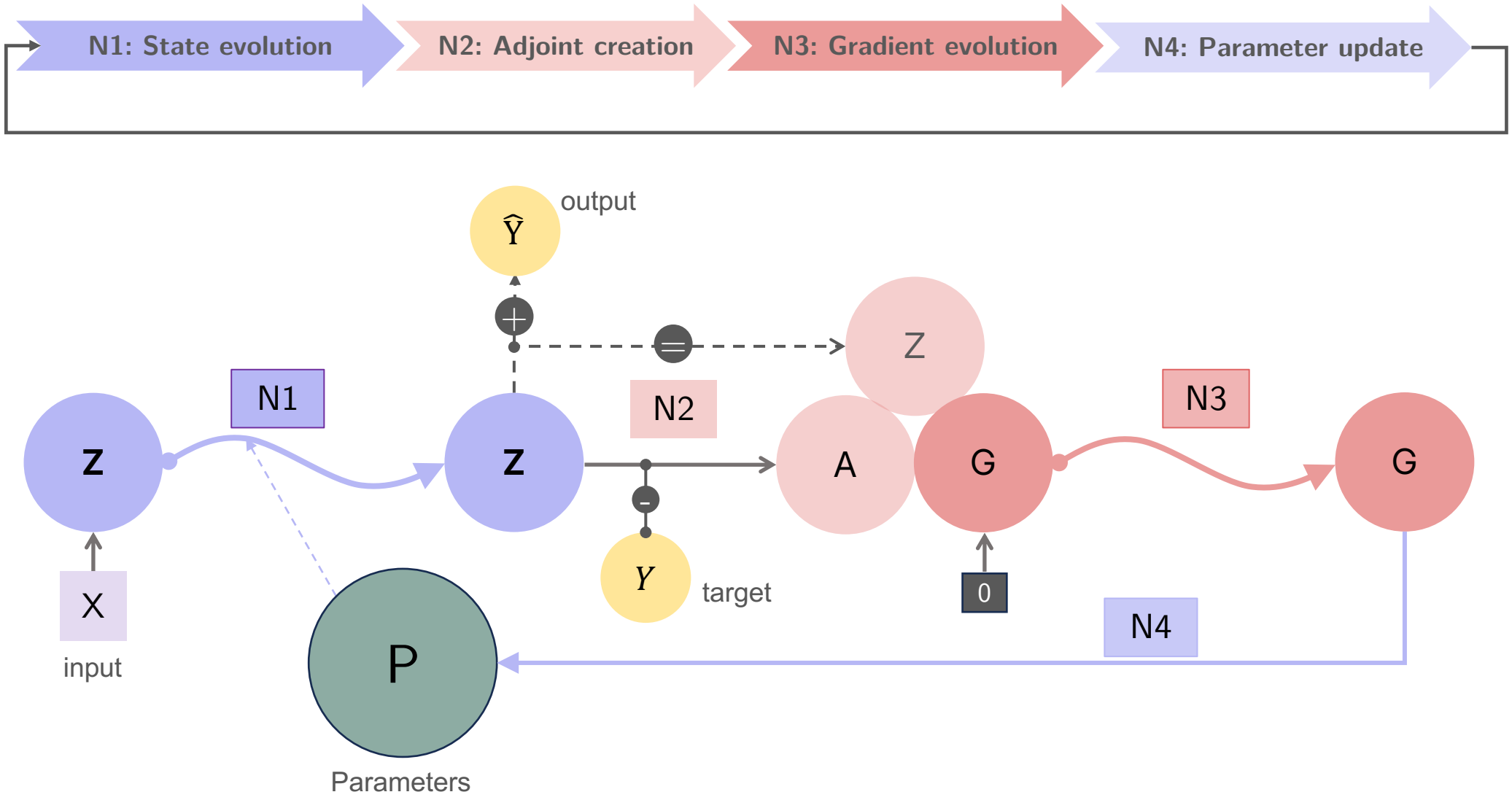
Integrated backward in time

(Invert sign parities when converting to CRNs)

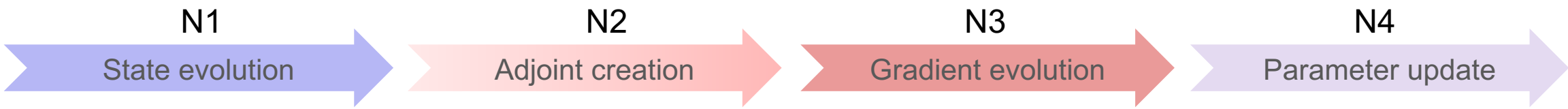
Neural Ordinary Differential Equations

Duke

Supervised learning pipeline in Neural CRNs



An example construction: $f_\theta = \theta \odot z$



$$\frac{d\mathbf{z}}{dt} = \theta \odot \mathbf{z}$$

$$P_i + Z_i \rightarrow Z_i + Z_i + P_i$$

Analog

Bimolecular

$$\boldsymbol{a} = \frac{\partial\, loss}{\partial \boldsymbol{z}}, loss = \frac{1}{2}(\hat{y} - y)^2$$

$$a = \hat{y} - y$$

$$\hat{y} = \sum_i z_i$$

$$a_i = \sum_i z_i - y$$

$$Z_i \rightarrow A^+$$
$$Y \rightarrow A^-$$

Discrete algebraic

Unimolecular

(end-point rxn.)

$$\frac{d\mathbf{z}}{dt} = -\theta \odot \mathbf{z}$$

$$\frac{d\mathbf{a}}{dt} = \mathbf{a} \odot \frac{\partial f_\theta}{\partial \boldsymbol{z}} = \mathbf{a} \odot \boldsymbol{\theta}$$

$$\frac{d\mathbf{g_\theta}}{dt} = \mathbf{a} \odot \frac{\partial f_\theta}{\partial \theta} = \mathbf{a} \odot \mathbf{z}$$

(*sign parities inverted*)

$$Z_i + P_i \rightarrow P_i$$
$$A_i^{\pm} + P_i \rightarrow 2A_i^{\pm} + P_i$$
$$A_i + Z_i \rightarrow G_i + A_i + Z_i$$

Analog

Bimolecular

$$\theta_i = \theta_i - g_\theta$$

$$G_i^+ + P_i \rightarrow \phi \text{ (fast)}$$
$$G_i^- \rightarrow P_i$$

Discrete algebraic

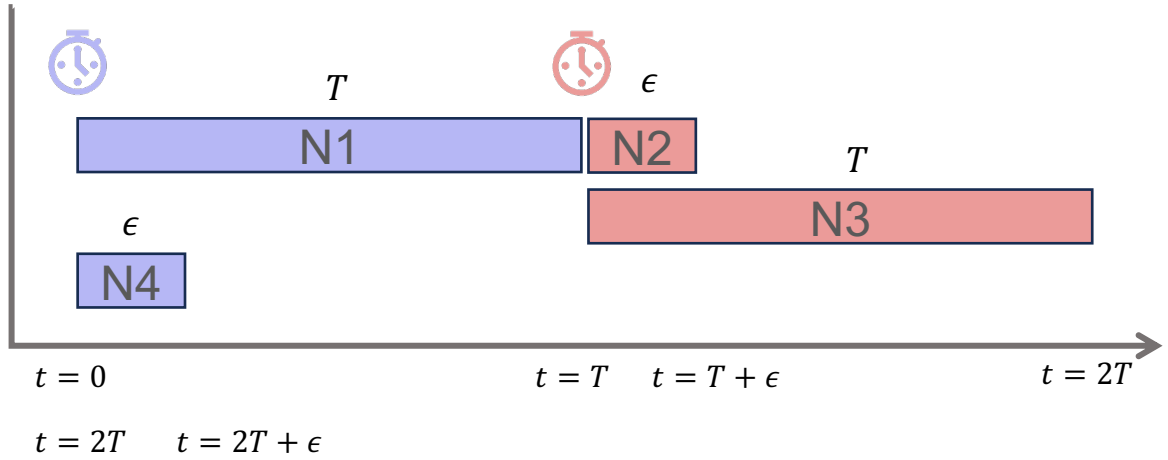
Unimolecular

(end-point rxn.)

Duke

Supervised learning pipeline within a two-phase clock system

N1	N2	N3	N4
State evolution	Adjoint creation	Gradient evolution	Parameter update
$P_i + Z_i \rightarrow Z_i + Z_i + P_i$	$Z_i \rightarrow A^+$ $Y \rightarrow A^-$	$Z_i + P_i \rightarrow P_i$ $A_i^{\pm} + P_i \rightarrow 2A_i^{\pm} + P_i$ $A_i^{\pm} + Z_i \rightarrow G_i^{\pm} + A_i^{\pm} + Z_i$	$G_i^+ + P_i \rightarrow \phi$ (fast) $G_i^- \rightarrow P_i$
Analog	Discrete algebraic	Analog	Discrete algebraic
Bimolecular	Unimolecular	Bimolecular	Unimolecular



Enable time-scale separation between unimolecular and bimolecular reactions

Duke

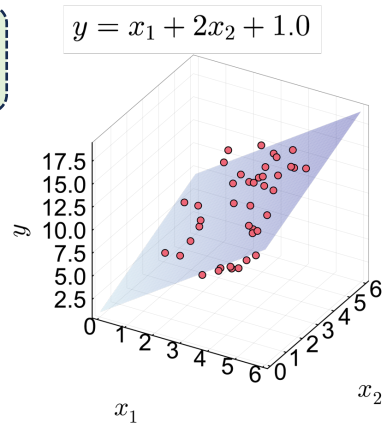
A circuit for linear regression

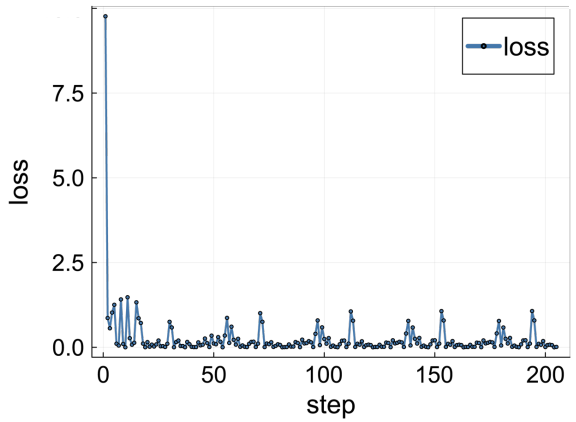
$$f_\theta = \theta \odot \mathbf{x} + \beta \quad \theta \in R^2_{>0} \text{ and } \mathbf{x} \in R^2_{>0}, \beta \in R_{>0}$$

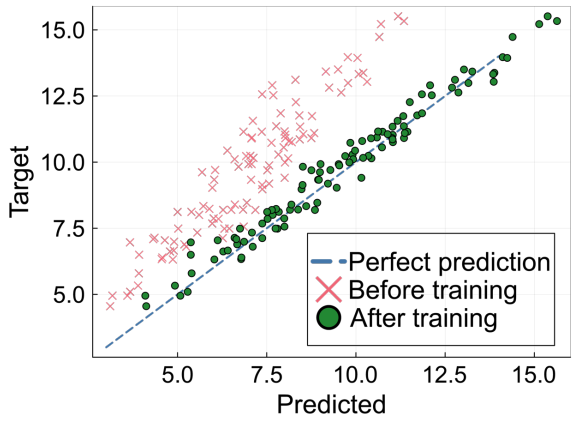
Group name	Species
Inputs	X_1, X_2
Output	Y
Parameters	P_1, P_2
Bias	B
States	Z_1, Z_2
Adjoints	$A^{\pm}$
Gradients	$G_1^{\pm}, G_2^{\pm}$
Clock signals	C_1, C_2
Total	**17**

Stage	ODE	CRN
N1	$\frac{dz_i}{dt} = \theta_i x_i + \beta$	$X_i + P_i \rightarrow Z_i + X_i + P_i$ $B \rightarrow Z_1 + Z_2$
N2	$a_i = z_1 + z_2 - y$	$Z_i \rightarrow A^+$ $Y \rightarrow A^-$
N3	$\frac{dg_i}{dt} = a_i x_i$	$X_i + A^{\pm} \rightarrow G_i^{\pm} + A^{\pm} + X_i$
N4	$\theta_i = \theta_i - g_\theta$	$G_i^+ + P_i \rightarrow \phi$ $G_i^+ \rightarrow P_i$
Total	**14**	
Flushing	$\{A^{\pm}, X\} \rightarrow \phi$	

$$y = x_1 + 2\,x_2 + 1$$



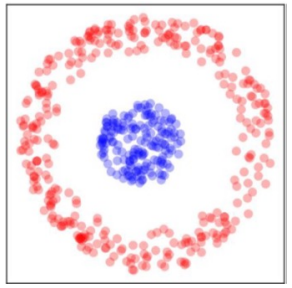




Nonlinear modeling in Neural CRNs

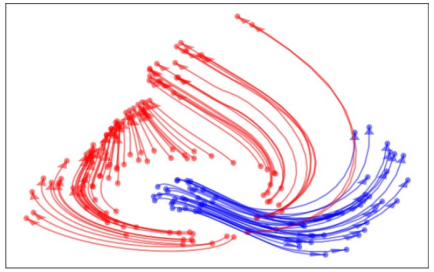
In neural networks

Use of nonlinear activation functions like ReLU



In ODE-nets

- Computations are modeled as flow trajectories
  - Must be unique and non-intersecting for each input
  - Training involves finding the right set of parameters
- A nonlinear f_θ alone cannot guarantee this.



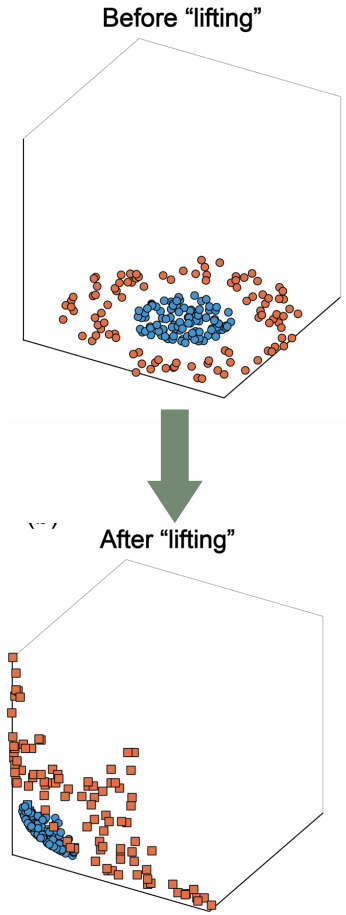
Neural ODE

[Augmented Neural ODEs]

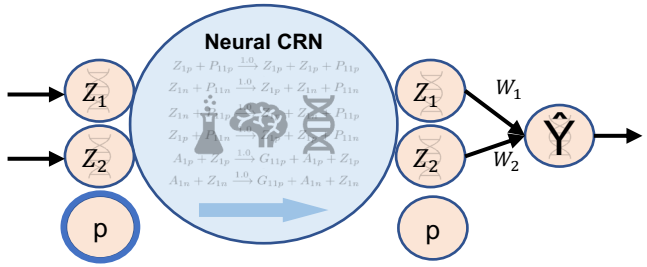
Implicit lifting and nonlinear Neural CRNs

$$f_\theta(\boldsymbol{x}, \boldsymbol{z}) = \theta \odot x \; - \alpha z \odot z \quad \text{(bimolecular)}$$

Incorporate the Implicit lifting trick

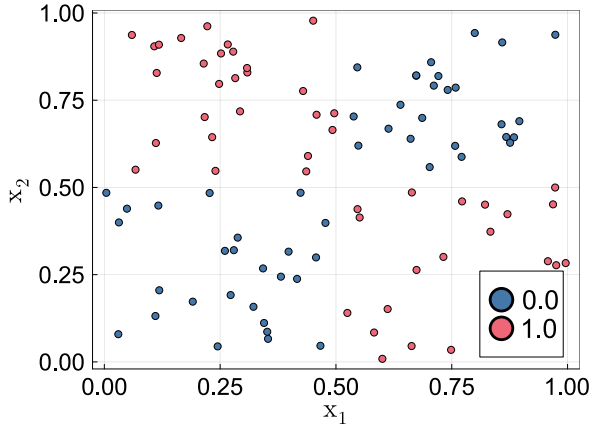


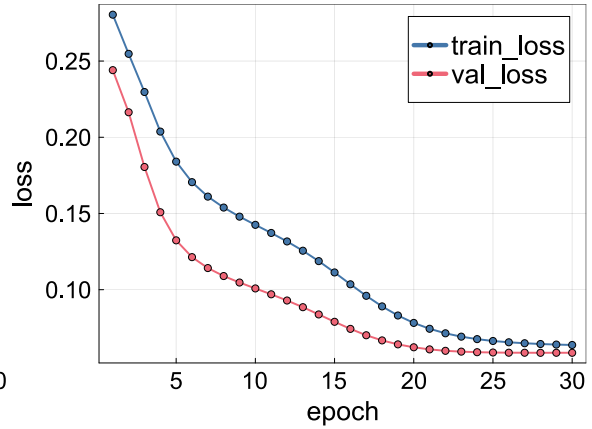
(a) “augment” dimensionality
(b) Use nonlinear f_θ

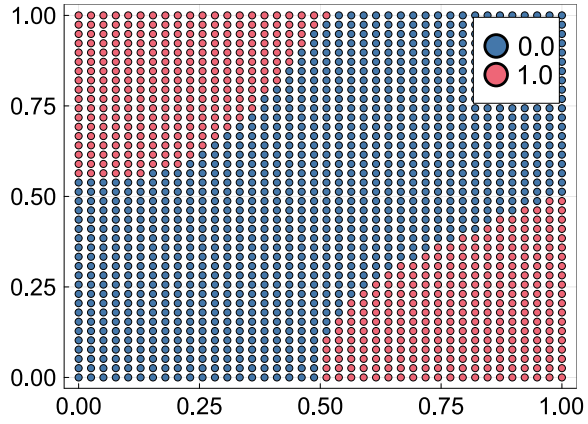


Padded dimension

Learned XOR decision boundary







First-order gradient approximation for nonlinear modeling

$$f_\theta(\boldsymbol{x}, \boldsymbol{z}) = \theta \odot x \; - \alpha z \odot z$$

Let's assume that adjoint **a** is kept constant

$$\frac{\mathrm{d}\mathbf{g_\theta}}{\mathrm{dt}} = \mathbf{a} \odot \mathbf{x}$$

$$A^{\pm} + X \rightarrow G^{\pm} + A + X$$

$$\mathbf{g_\theta} = \mathrm{T}\mathbf{a} \odot \mathbf{x} = \mathrm{T}(\hat{y} - y)x_i$$

Recall $\mathbf{a} = \hat{\mathrm{y}} - \mathrm{y}$

Feedback phase

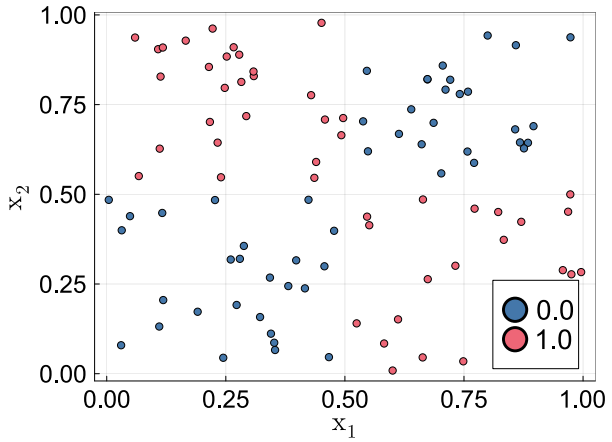
$$\frac{\mathrm{d}\mathbf{g_\theta}}{\mathrm{dt}} = \mathbf{a} \odot \mathbf{x}$$

✗ $$\frac{\mathrm{d}\mathbf{a}}{\mathrm{dt}} = -2\alpha\,(\mathbf{a} \odot \mathbf{z})$$

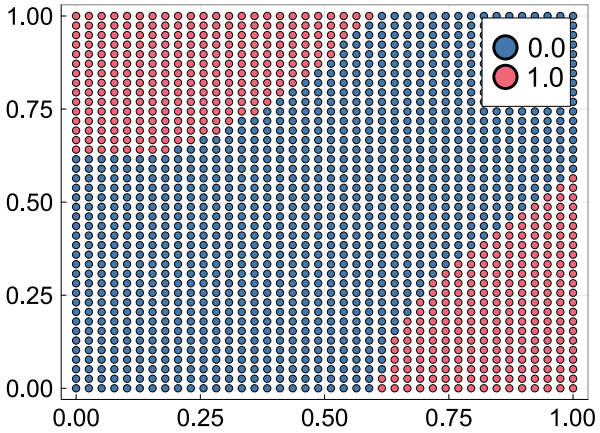
✗ $$\frac{\mathrm{d}\mathbf{z}}{\mathrm{dt}} = -\theta \odot \mathbf{x} + \alpha\mathbf{z} \odot \mathbf{z}$$

Similar to perceptron learning $\Delta\mathrm{w} = \mathrm{LR} * (\hat{\mathrm{y}} - \mathrm{y})\mathrm{x}$

XOR Task



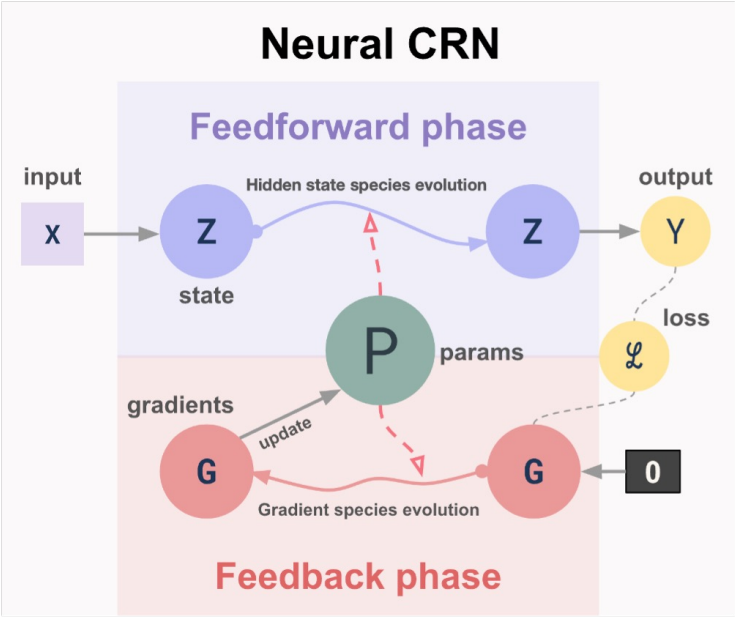
XOR Inference



Summary

- Neural CRNs: A supervised learning algorithm where chemical reactions behave as end-to-end computational units
- The whole pipeline can be executed within two clock phases
- Nonlinear modeling using unimolecular and bimolecular reactions
- Reduction in circuit size through linear approximation of gradients
- A minimal linear circuit: 14 species and 13 abstract reactions
- Possibility of Asynchronous execution (some stages)

…many more



Thank you!