Convolutional Filters and Neural Networks with Non Commutative Algebras

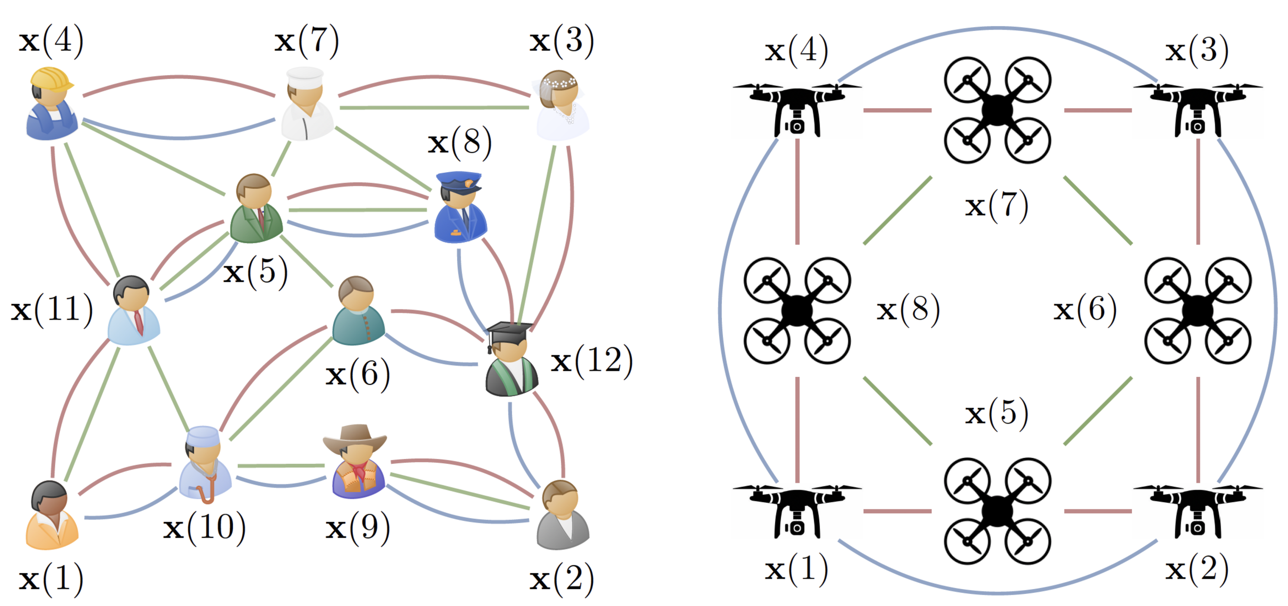
Alejandro Parada-Mayorga, Landon Butler, and Alejandro Ribeiro

Penn, University of Pennsylvania

Berkeley, University of California

Introduction

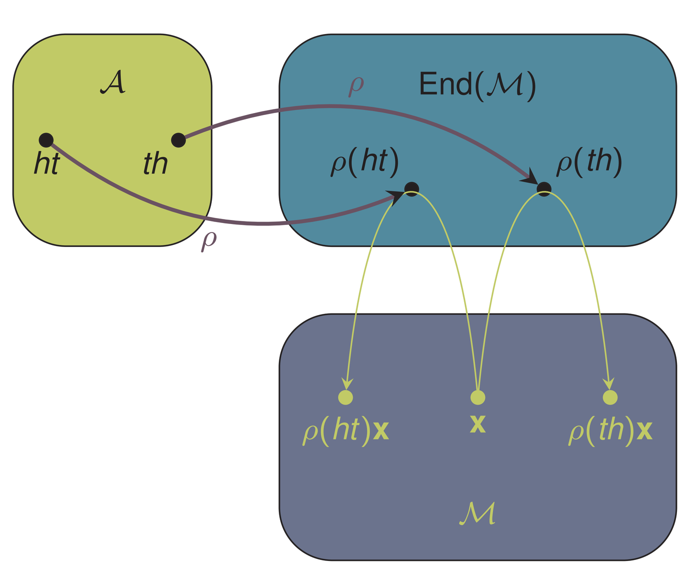
Non commutative signal models arise naturally in scenarios where the information of interest is processed by collection of non commutative operators and their compositions.



Prominent examples of non commutative signal models appear in heterogeneous networked systems associated with autonomous systems, social networks, and non commutative groups.

Noncommutative models in Algebraic Signal Processing (ASP)

- An Algebraic SP model: $(\mathcal{A}, \mathcal{M}, \rho)$
- $\mathcal{A}$: Algebra with unity where filters $h \in \mathcal{A}$
- $\mathcal{M}$ is a vector space
  - $\Rightarrow$ Contains signal $\mathbf{x}$ we want to process
- ρ: Homomorphism from $\mathcal{A}$ to $\mathcal{M}$
  - $\Rightarrow$ $\mathcal{M}$: Space of Endomorphisms of $\mathcal{M}$
  - $\Rightarrow$ Instantiates the abstract filter h in $\mathrm{End}(\mathcal{M})$



$\mathbf{y} = \rho(h)\mathbf{x}$: **convolution** between $h \in \mathcal{A}$ and $\mathbf{x} \in \mathcal{M}$

Note: An algebra is simply a vector space where there is also defined a notion of **product** that is closed. A classical example of a non commutative algebra is the algebra of matrices of size $n \times n$ where the algebra product is the ordinary product of matrices.

If $\mathcal{A}$ has generators $g_1, \ldots, g_m$, then the operators $\mathbf{S}_1 = \rho(g_1), \ldots, \mathbf{S}_m = \rho(g_m)$ are the independent variables of the filters in the signal model, which we refer to as the **shift operators**. For instance, if $\mathcal{A}$ has two generators, a convolutional filter could be $p(\mathbf{S}_1, \mathbf{S}_2) = \mathbf{S}_1^2 + \mathbf{S}_1\mathbf{S}_2 + \mathbf{S}_2^6\mathbf{S}_1\mathbf{S}_2^8$, where $\mathbf{S}_1$ and $\mathbf{S}_2$ do not commute. If $\mathbf{x} \in \mathcal{M}$ is a signal, filtering $\mathbf{x}$ by $p(\mathbf{S}_1, \mathbf{S}_2)$ produces the signal $\mathbf{y} = p(\mathbf{S}_1, \mathbf{S}_2)\mathbf{x}$. The operators $\mathbf{S}_i = \rho(g_i)$ capture structural properties of the domain of the signals in $\mathcal{M}$.

Frequency Representations

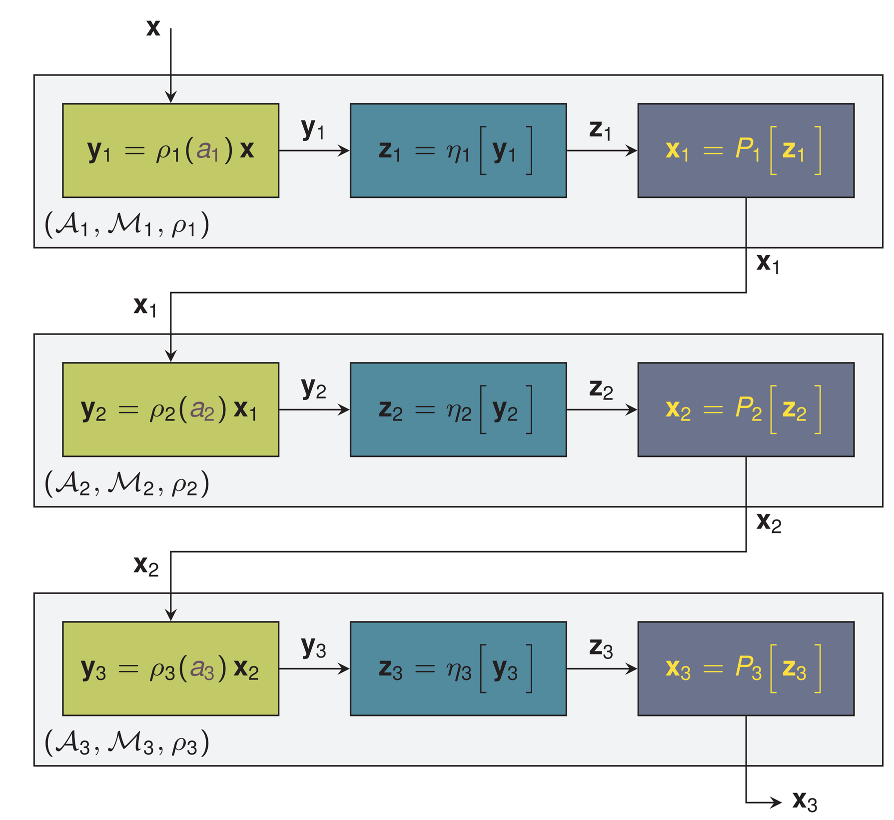
Let the shift operators $\{\mathbf{S}_i\}_{i=1}^m$ be diagonalizable, with $\mathbf{S}_i = \mathbf{U}\mathrm{diag}\left(\boldsymbol{\Sigma}_1^{(i)}, \ldots, \boldsymbol{\Sigma}_\ell^{(i)}\right)\mathbf{U}^T$, with $\boldsymbol{\Sigma}_j^{(i)} \in \mathbb{R}^{p_j \times p_j}$, and $\mathbf{U}$ orthogonal. If $d = \max_j\{p_j\}$ and $\boldsymbol{\Lambda}_i \in \mathbb{R}^{d \times d}$, we say that the polynomial matrix function

$$p(\boldsymbol{\Lambda}_1, \ldots, \boldsymbol{\Lambda}_m) : \left(\mathbb{R}^{d \times d}\right)^m \to \mathbb{R}^{d \times d}, \tag{1}$$

is the spectral representation of the filter $p(\mathbf{S}_1, \ldots, \mathbf{S}_m)$, where $\left(\mathbb{R}^{d \times d}\right)^m$ is the m-times cartesian product of $\mathbb{R}^{d \times d}$.

Non commutative convolutional architectures

- Stacked layered structure
- Each layer $\Rightarrow$ Specific ASM $(\mathcal{A}_\ell, \mathcal{M}_\ell, \rho_\ell)$
- Map from layer ℓ to $\ell + 1$ by $\sigma_\ell = P_\ell \circ \eta_\ell$
- P_ℓ: Pooling and η_ℓ: Pointwise nonlinearity
- σ_ℓ is considered Lipschitz with $\sigma_\ell(0) = 0$



General Perturbations in Algebraic Non commutative Models

Perturbation Model

We describe the perturbations as deformations on the shift operators of the ASM. Then, if $\mathbf{S}$ is a shift operator in $(\mathcal{A}, \mathcal{M}, \rho)$ we derive a perturbed version of $\mathbf{S}$ as

$$\tilde{\mathbf{S}} = \mathbf{S} + \mathbf{T}(\mathbf{S}), \tag{2}$$

where $\mathbf{T}(\mathbf{S}_i) = \mathbf{T}_{0,i} + \mathbf{T}_{1,i}\mathbf{S}_i$, with $\left\|\mathbf{T}_{i,r}\right\|_F \leq \delta \left\|\mathbf{T}_{i,r}\right\|$, where $\delta > 0$.

- **Intuition:** We aim to use the filter $p(\mathbf{S}_1, \ldots, \mathbf{S}_m)$, but due to the perturbation we end up using the filter $p\left(\tilde{\mathbf{S}}_1, \ldots, \tilde{\mathbf{S}}_m\right)$ $\Rightarrow$ Same polynomial expression, different independent variables.

Stability of Convolutional Filters

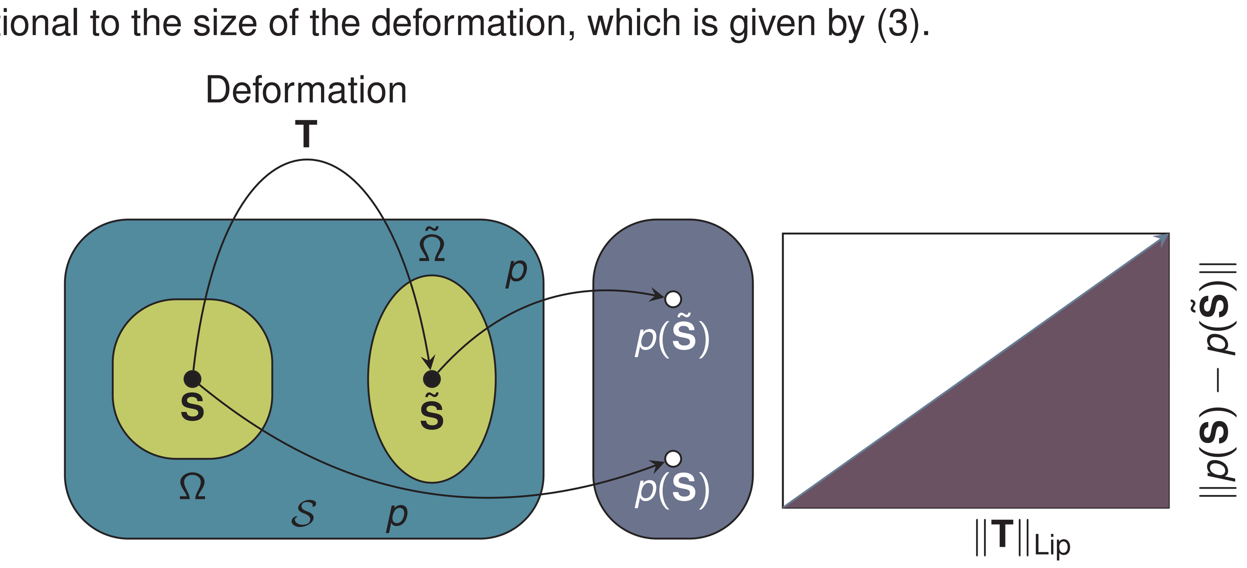
Let $p(\mathbf{S}_1, \ldots, \mathbf{S}_m)$ be a convolutional filter and $p(\tilde{\mathbf{S}}_1, \ldots, \tilde{\mathbf{S}}_m)$ its perturbed version. Then, we say that p is stable to deformations if there exist constants $C_0, C_1 > 0$ such that

$$\left\|p(\mathbf{S})\mathbf{x} - p(\tilde{\mathbf{S}})\mathbf{x}\right\| \leq \left[C_0 \sup_{\mathbf{S}\in\mathcal{S}} \|\mathbf{T}(\mathbf{S})\| + C_1 \sup_{\mathbf{S}\in\mathcal{S}} \|D_{\mathbf{T}}(\mathbf{S})\| + \mathcal{O}\left(\|\mathbf{T}(\mathbf{S})\|^2\right)\right] \|\mathbf{x}\|, \tag{3}$$

for all $\mathbf{x} \in \mathcal{M}$. In (3) $D_{\mathbf{T}}(\mathbf{S})$ is the Fréchet derivative of the perturbation operator $\mathbf{T}$.

Note: The right hand side of (3) is a norm called the *Lipschitz norm*, $\|\mathbf{T}\|_{Lip}$, which provides the standard measure for diffeomorphisms acting between arbitrary spaces.

- **Intuition:** The notion of stability indicates that the size of the change in the the filter $p(\mathbf{S}_1, \ldots, \mathbf{S}_m)$ is proportional to the size of the deformation, which is given by (3).



Stability to Deformations in Non commutative Convolutional Architectures

Stability Theorem

Let $\{(\mathcal{A}_\ell, \mathcal{M}_\ell, \rho_\ell; \sigma_\ell)\}_{\ell=1}^L$ be a **non commutative** convolutional architecture with mapping operator $\{(\mathcal{A}_\ell, \mathcal{M}_\ell, \tilde{\rho}_\ell; \sigma_\ell)\}_{\ell=1}^L$, and let $\Phi\left(\mathbf{x}, \{\mathcal{F}_\ell\}_1^L, \{\tilde{\mathcal{S}}_\ell\}_1^L\right)$ its perturbed version. If the filters $\{\mathcal{F}_\ell\}_1^L$ in the network are **Lipschitz** and **integral Lipschitz** we have

$$\left\|\Phi\left(\mathbf{x}, \{\mathcal{P}_\ell\}_1^L, \{\mathcal{S}_\ell\}_1^L\right) - \Phi\left(\mathbf{x}, \{\mathcal{P}_\ell\}_1^L, \{\tilde{\mathcal{S}}_\ell\}_1^L\right)\right\| \leq \boldsymbol{\Delta}\|\mathbf{x}\|,$$

where $\boldsymbol{\Delta}$ is given by

$$\boldsymbol{\Delta} = C\delta m\left(L_0^{(\ell)} \sup_{\mathbf{S}_{i,\ell}} \|\mathbf{T}^{(\ell)}(\mathbf{S}_{i,\ell})\| + L_1^{(\ell)} \sup_{\mathbf{S}_{i,\ell}} \|D_{\mathbf{T}^{(\ell)}}(\mathbf{S}_{i,\ell})\|\right), \tag{4}$$

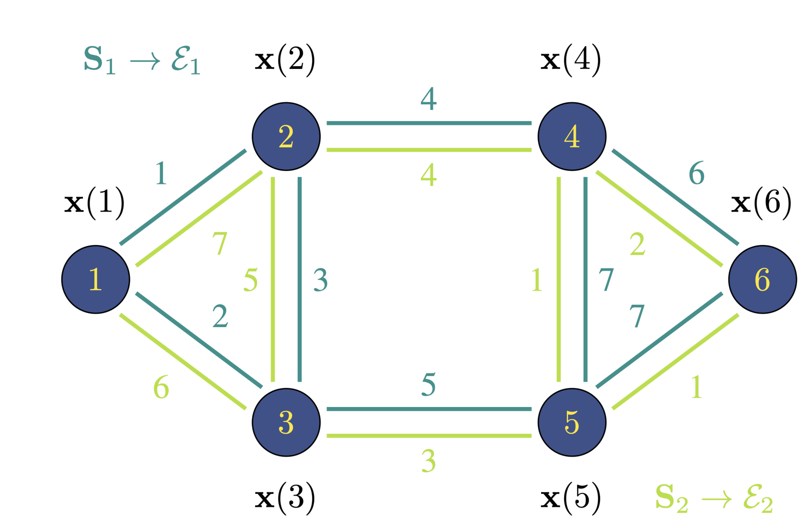
with (ℓ) indicating quantities and constants associated to the layer ℓ, and C a fixed constant.

Note: Integral Lipschitz filters behave like constant functions on high frequency components.

- The stability comes at the price of reducing the selectivity of the filters. However, this is compensated by the pointwise nonlinearities, which redistribute frequency information.

Numerical Experiments with Multigraph Neural Networks

- A salient instantiation of the ASP model:
  - $\Rightarrow$ $\mathcal{A}$: Set of non commutative polynomials over generators
  - $\Rightarrow$ $\mathcal{M}$: Vector space of node signals
  - $\Rightarrow$ ρ: Mapping of generators to shift operators (matrix representation of edges)



With generators t_1, t_2, shift operators $\mathbf{S}_1, \mathbf{S}_2$, and node signal $\mathbf{x}$, a filter may look like:

$$\rho\left(t_1^2 + t_1t_2 + 2t_2t_1 + t_2^2 + 1\right)\mathbf{x} = \left(\mathbf{S}_1^2 + \mathbf{S}_1\mathbf{S}_2 + 2\mathbf{S}_2\mathbf{S}_1 + \mathbf{S}_2^2 + 1\right)\mathbf{x}$$

We consider a recommendation system task using the MovieLens dataset, constructing a multigraph over movies with edges representing genre similarity and rating similarity. Estimates of the rating similarities are formed using samples from the training set, which we vary in size to perturb the corresponding operator.

Three architectures are used: a multigraph filter MultiFilter, a multigraph neural network MultiGNN, and a multigraph neural network regularized by its filters' integral Lipschitz constant MultiGNN (IL). We measure the change in RMSE / convolutional output upon using the perturbed operators.