Cours 14: Opérateurs neuronaux et problèmes temporels

Emmanuel Franck*,

28 novembre, 2024
Master CMSI, M2, Strasbourg

*MACARON project-team, Université de Strasbourg, CNRS, Inria, IRMA, France

Outline

Objectif: inversion d'opérateur

- Soit un problème elliptique de la forme

$$\begin{cases} \partial_t u - \Delta u = f(t, \mathbf{x}), & \mathbf{x} \in \Omega \\ u(t, \mathbf{x}) = 0 & \mathbf{x} \in \partial\Omega \\ u(0, \mathbf{x}) = u_0(\mathbf{x}) & \mathbf{x} \in \Omega \end{cases}$$

- On voudrait prédire rapidement une solution pour un $f(t, \mathbf{x})$ ou un $u_0(\mathbf{x})$ généraux.
- Si la solution existe, on peut formellement dire qu'il existe $G : \mathcal{H}_x \rightarrow \mathcal{H}_y$ tel que

$$u(.) = G^{-1}(f(.), u_0(.))$$

- Comme précédemment peut généraliser en cherchant à approcher $G^{+}(\mathbf{a}(.)) = u(.)$ avec $\mathbf{a}$ l'ensemble des fonctions qui sont des entrées du problème (sources, conditions limites, conditions initiales, coefficients de l'EDP).
- Si on regarde précisément le problème:

$$u(t, .) = G^{-1}(u_0(.))$$

- On voit qu'il s'agit en faite d'apprendre le flot d' EDP. On retombe sur les problèmes du chapitre 5.
- **Deux cas**: flot discret ou continue en temps.

Opérateur neuronaux continue en temps

Opérateur neuronaux discret en temps

Opérateur neuronaux continue en temps

Problème temporelle et Green

- Peut t'on généraliser la théorie de Green ? oui

$$u(t,\mathbf{x}) = \int_\Omega G(t,0,\mathbf{x},\mathbf{y})u_0(\mathbf{y})d\mathbf{y} + \int_0^t \int_\Omega G(t,\tau,\mathbf{x},\mathbf{y})f(\tau,\mathbf{y})d\mathbf{y}d\tau$$
$$+ \int_0^t \int_{\partial\Omega} [u(t,\mathbf{x})\nabla G(t,0,\mathbf{x},\mathbf{y}) - G(t,0,\mathbf{x},\mathbf{y})\nabla u(t,\mathbf{x})]d\mathbf{y}d\tau$$

  avec

$$(\partial_t - \Delta)G(t,\tau,\mathbf{x},\mathbf{y}) = \delta(\mathbf{x}-\mathbf{y})\delta(t-\tau)$$

  et des conditions initiales/bords nulles.

GreenNet

Il s'agit d'un réseau à une un couche sans couche d'extrapolation et de projection ou la couche principale est $G_\theta : H^s(\Omega)^p \rightarrow H^s(\Omega)^p$ est paramétrée de la façon suivante:

$$G_\theta(\mathbf{f},\mathbf{u}_0) = \int_\Omega k_\theta(t,0,\mathbf{x},\mathbf{y})\mathbf{u}_0(\mathbf{y})d\mathbf{y} + \int_0^t \int_\Omega k_\theta(t,\tau,\mathbf{x},\mathbf{y})f(\tau,\mathbf{y})d\mathbf{y}d\tau$$

avec k_θ un réseau de type MLP, Fourier etc.

Opérateur neuronale temporel

GreenNet temporel de Bas rang

Soit **f** une fonction uniquement spatiale (condition initiale) ou dépendent du temps (source). On propose de prendre:

$$G_\theta(\mathbf{f}) = \sum_{k=1}^{R} \langle f, \psi_i \rangle_{L^2} \phi_{\theta,k}(\mathbf{x}, t)$$

avec

$$\langle f, \psi_i \rangle_{L^2} = \frac{1}{m} \sum_{j=1}^{m} \psi_{\theta,i}(\mathbf{x}_j, t_j) f(\mathbf{x}_j, t_j), \quad \text{ou} \quad \langle f, \psi_i \rangle_{L^2} = \frac{1}{m} \sum_{j=1}^{m} \psi_{\theta,i}(\mathbf{x}_j, t_j) f(\mathbf{x}_j),$$

DeepONet temporel

Soit **a** une fonction uniquement spatiale (condition initiale) ou dépendent du temps (source). On propose de prendre:

$$G_\theta(\mathbf{a}(t_1, \mathbf{x}_1), ...\mathbf{a}(t_m, \mathbf{x}_m)) = \sum_{i=1}^{R} \alpha_i b_{\theta,i}(\mathbf{x}, t)$$

avec b_θ un réseau de type MLP de $\mathbb{R}^d$ dans $\mathbb{R}^R$ et

$$t_\theta(\mathbf{a}(t_1, \mathbf{x}_1), ...\mathbf{a}(t_m, \mathbf{x}_m)) \rightarrow (\alpha_1, ..., \alpha_K)$$

un réseau de $\mathbb{R}^{q\times m}$ dans $\mathbb{R}^R$

FNO continue en temps

- Une couche de Fourier est donnée par:

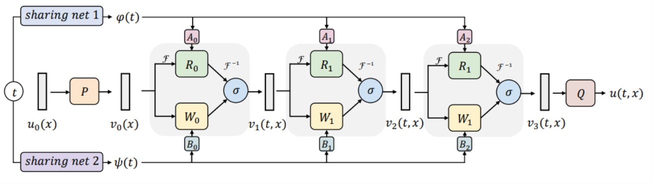
$$\mathbf{v}_{l+1} = \sigma\Big(W_l \mathbf{v}_l + (\mathcal{F}^{-1} \circ \hat{\kappa}_{\theta_l} \circ \mathcal{F})(\mathbf{v}_l)\Big)$$

- Pour l'étendre au temporel il est proposé une modulation temporelle des poids de chaque couche:

$$\mathbf{v}_{l+1} = \sigma\Big(\psi_{\theta_t}(t) W \mathbf{v}_l + (\mathcal{F}^{-1} \circ \phi_{\theta_t}(t, \mathbf{k}) \hat{\kappa}_{\theta_l} \circ \mathcal{F})(\mathbf{v}_l)\Big)$$

  avec $\mathbf{k}$ les fréquences.

- Il est proposé de prendre $\phi_{\theta_t}(t, \mathbf{k}) = \phi(t) A_l(\mathbf{k})$ avec $A(\mathbf{k})$ une matrice entrainable.
- Il est proposé de prendre $\phi_{\theta_t}(t, \mathbf{k}) = diag(\psi(t) B_l)$ avec B_l une matrice entrainable.
- les fonctions $\psi(t)$ et $\phi(t)$ sont communes à toutes les couches.



Transformeur

- Pour les transformeurs il est proposé de moduler temporellement une sous couche du modèle.

Bloc d'un opérateur neuronal de type transformeur temporel

Le bloc est donné par
$$
\begin{aligned}
\mathbf{v}^{(l-1)}(\mathbf{x}) &\leftarrow \mathrm{W}_1\mathbf{v}^{(l-1)}(\mathbf{x}) + \mathrm{A}_{\text{MultiHead}}\left(\mathbf{v}^{(l-1)}\right),\\
\mathbf{v}^{(l-1)}(\mathbf{x}) &\leftarrow \mathrm{F}_{\text{LayerNorm}}\left(\mathbf{v}^{(l-1)}(\mathbf{x})\right),\\
\mathbf{v}^{(l-1)}(\mathbf{x}) &\leftarrow \mathrm{W}_2\mathbf{v}^{(l-1)(\mathbf{x})} + \mathrm{F}_{\mathrm{NN}}\left(\mathbf{v}^{(l-1)}(\mathbf{x})\right),\\
\mathbf{v}^{(l)}(\mathbf{x}) &\leftarrow \mathrm{F}_{\text{LayerNorm}}\left(\mathbf{v}^{(l-1)}(\mathbf{x})\right),
\end{aligned}
$$
avec

$$
\left(F_{\text{LayerNorm}}\left(\mathbf{v};\gamma,\beta\right)(\mathbf{x})\right)_k = \gamma_k(t)\cdot\frac{(\mathbf{v}(\mathbf{x}))_k - m(\mathbf{v}(\mathbf{x}))}{\sqrt{\sigma^2(\mathbf{v}(\mathbf{x}))+\epsilon}} + \beta_k(t),
$$

- Choix simple: $\beta(t) = \beta_1 + \beta_2 t$ et $\gamma(t) = \gamma_1 + \gamma_2 t$
- autre choix: des réseaux de neurones type MLP.

Propriété de semi-groupe

Propriété de semi groupe

Si on cherche a prédire la solution a partir de la condition initiale sans source. Le flot qui sera approchée par l'opérateur neuronale satisfait la structure de semi-groupe.

- La fonction de coût classique est donnée par:

$$\overline{\mathcal{L}}(\theta) := \frac{1}{M(K+1)} \sum_{i=1}^{M} \sum_{k=0}^{K} \| G(t_k, a_i(\mathbf{x})) - u_i(t_k, \mathbf{x}) \|_{L^p(D)}^p,$$

  avec $u_i(t_k, \mathbf{x})$ la solution associée à la condition initiale $a_i(\mathbf{x})$

- On impose faiblement la structure de semi-groupe

$$\widehat{\mathcal{L}}(\theta) := \frac{1}{M\widehat{K}} \sum_{i=1}^{M} \sum_{k,\bar{k}=0, k\leqslant \bar{k}}^{K} \| \, u_i(t_{\bar{k}}, \mathbf{x}) - G_\theta(t_{\bar{k}} - t_k, u_i(t_k, \mathbf{x})) \, \|_{L^p(D)}^p,$$

Opérateur neuronaux continue en temps

Opérateur neuronaux discret en temps

Opérateur neuronaux discret en temps

Flot discret et opérateur neuraux

Idée

On demande à l'opérateur neuronale de prédire la solution à des temps discret.

- On se donne un opérateur neuronale $G_\theta(u)$
- Plusieurs options:
  - On prédit un temps avec un temps: $u_{t_{k+1}}(\mathbf{x}) = G_\theta(u_{t_k}(\mathbf{x}))$
  - On prédit m temps avec m temps: $u_{t_{k+m}}(\mathbf{x}), ..., u_{t_{k+1}}(\mathbf{x}) = G_\theta(u_{t_k}(\mathbf{x}), ..., u_{t_{k-m}}(\mathbf{x}))$

NO auto-régressif

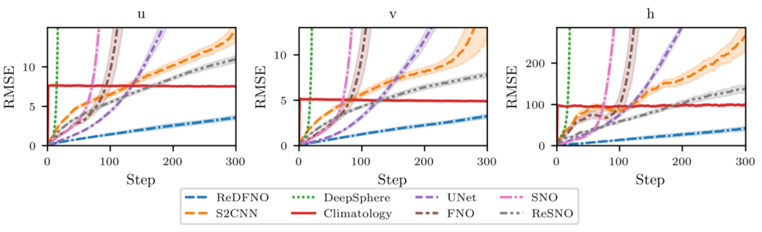
On utilise donc les opérateurs neuronaux de façon auto-régressif:

$$u_{t_n}(\mathbf{x}) = (G_\theta \circ ... \circ G_\theta)(u_0(\mathbf{x}))$$

Question centrale

Est ce ce type de prédiction est stable en temps ?

Stabilité



- Un entrainement sur plusieurs pas de temps (unrolled) peut stabiliser les opérateurs neuronaux.