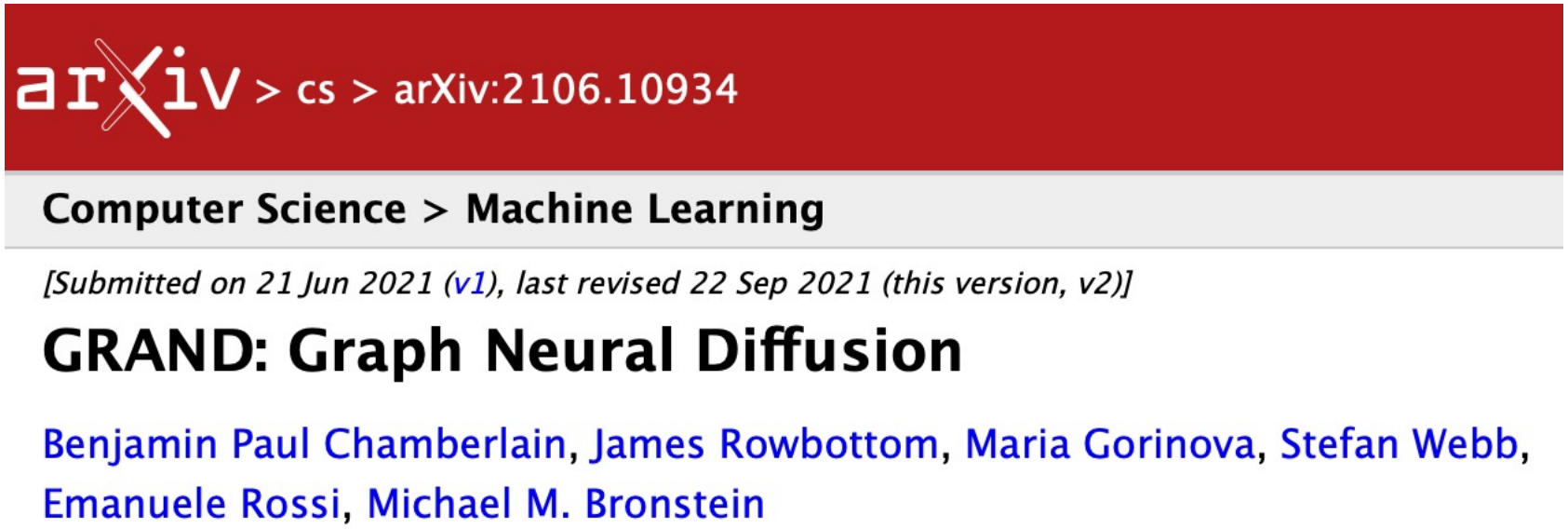
Du processus de diffusion sur les graphes aux GNN's

ou comment dépasser le Message Passing

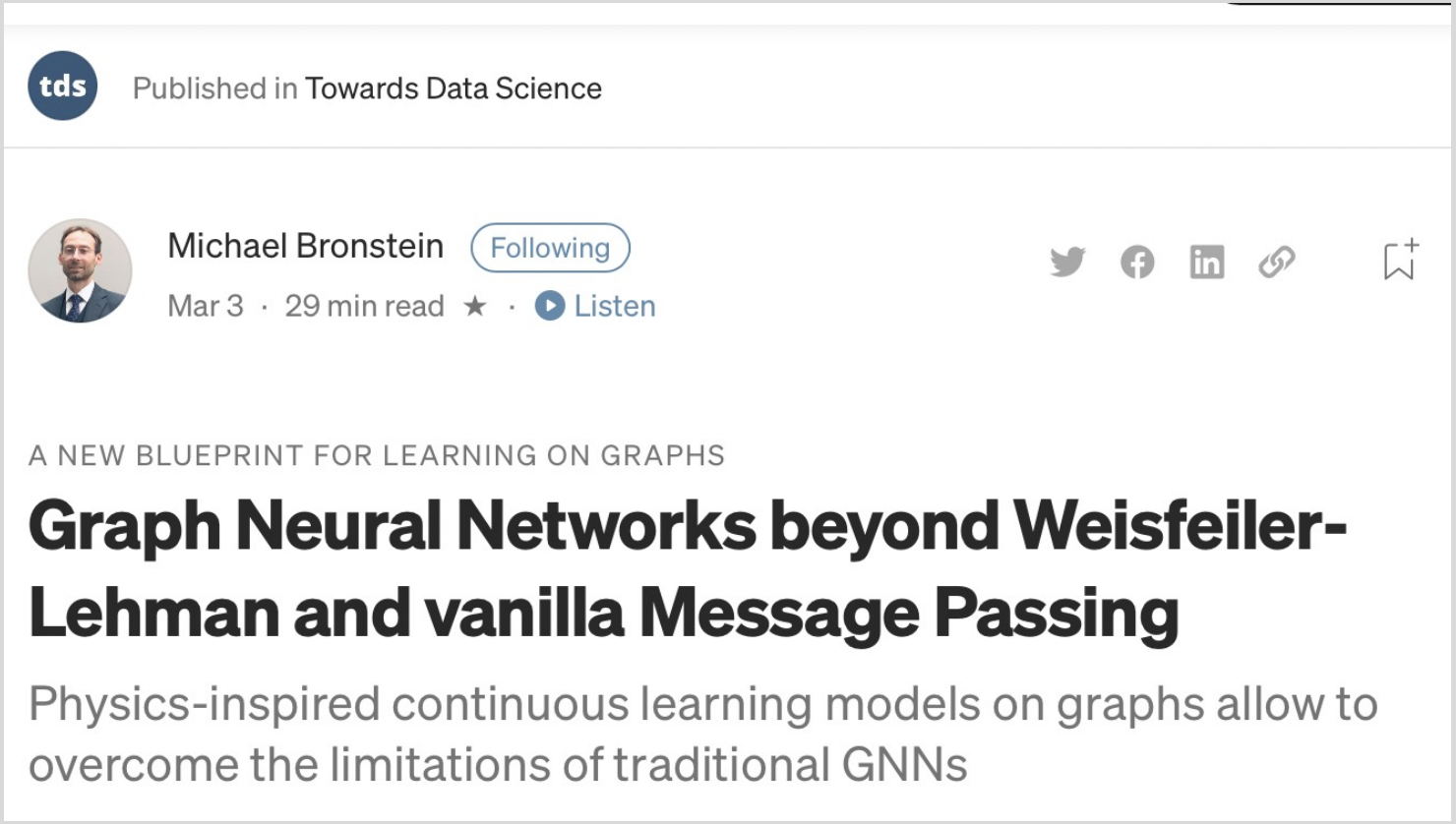
Séminaire TALia du 1 avril 2022

Source



Plan

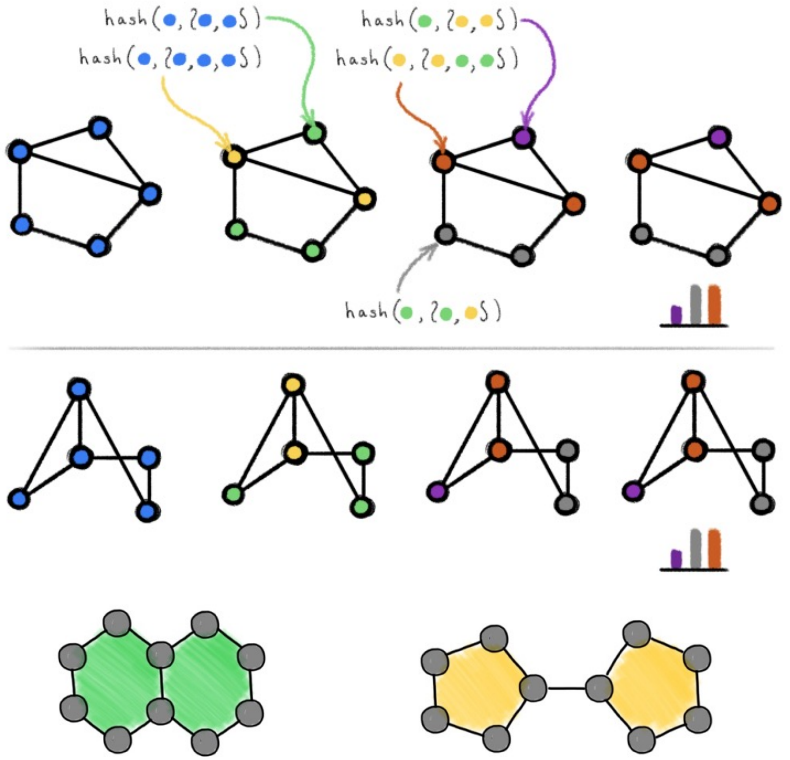
1. Limitations des GNN
2. L'équation de diffusion dans $\mathbb{R}^d$
3. L'équation de diffusion sur une variété riemannienne
4. L'équation de diffusion sur un graphe
5. Le modèle Graph Neural Diffusion
6. Comparaison avec les GNN usuels
7. Conclusion



référence

Message Passing : limitations, difficultés et choix arbitraires

$$\mathbf{h}_u^{(k+1)} = \text{UPDATE}^{(k)}\left(\mathbf{h}_u^{(k)}, \text{AGGREGATE}^{(k)}(\{\mathbf{h}_v^{(k)}, \forall v \in \mathcal{N}(u)\})\right)$$
$$= \text{UPDATE}^{(k)}\left(\mathbf{h}_u^{(k)}, \mathbf{m}_{\mathcal{N}(u)}^{(k)}\right),$$



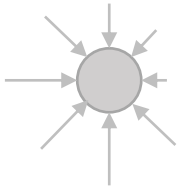
- Le paradigme **message-passing** des GNN (approches spatiales, spectrales) possède toutes les limitations de l'algorithme WL.
- **Over-smoothing** : les GNN trop profonds ne parviennent plus à distinguer le nœuds individuellement (filtre passe-bas).
- **Over-squashing** : lorsque l'information de nœuds lointains doit passer par des goulets d'étranglement
- **Hétérophile** difficile à prendre en compte (filtres passe-haut)
- L'invariance sous la renumérotation des nœuds implique l'**isotropie** dans la propagation de l'information.
- Inadéquation d'un **espace euclidien** pour représenter la notion de proximité entre nœuds dans un graphe
- Arbitraire des procédures de **rewiring** utilisées pour découpler le graphe de calcul du graphe des données (référence)
- Arbitraire des procédures de **décoration/coloriage** des nœuds (référence)

EDP : un réservoir de concepts, de résultats et de méthodes

Idée générale : envisager un GNN comme une **discrétisation**, dans le temps (couches) et dans l'espace (nœuds), d'un processus de diffusion de l'information (features x_u) définit sur une variété riemannienne.

Équation de **continuité**

$$\frac{\partial x(u,t)}{\partial t} = -\operatorname{div} \mathbf{h}(u,t)$$



$$\nabla \cdot \mathbf{h} = \frac{\partial h_1}{\partial u_1} + \frac{\partial h_2}{\partial u_2} + \frac{\partial h_3}{\partial u_3}$$

Flux (de matière, de chaleur…) lié au gradient

$$\mathbf{h}(u,t) = -g \nabla x(u,t)$$

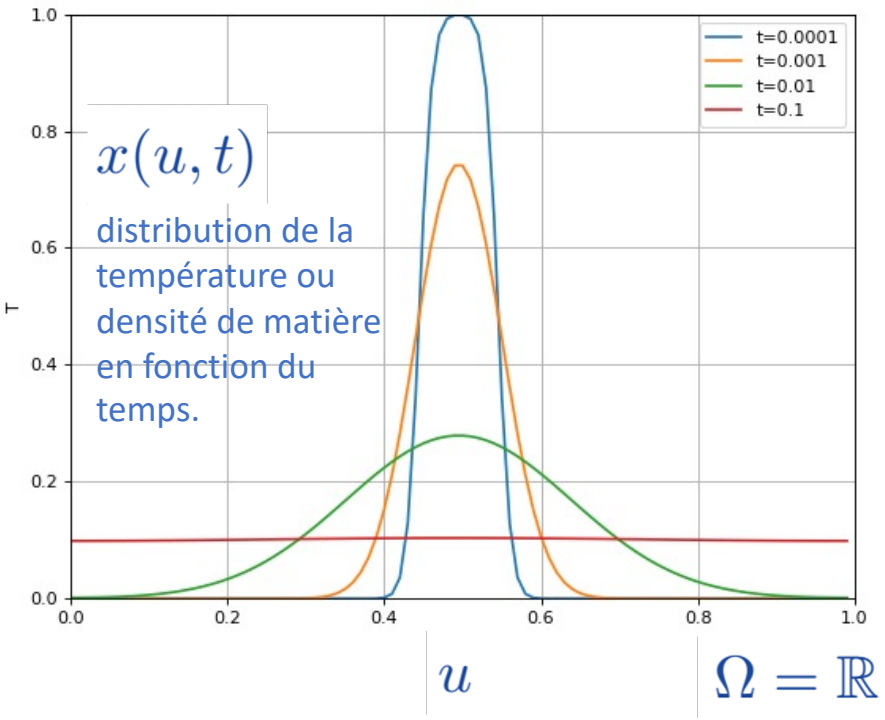
$$g = g(u, x(u,t), t)$$

coefficient de diffusion

Equation de la **diffusion**

$$\frac{\partial x(u,t)}{\partial t} = \operatorname{div}\left[g(u, x(u,t), t)\, \nabla x(u,t)\right]$$

$$x(u,0) = x_0(t)$$



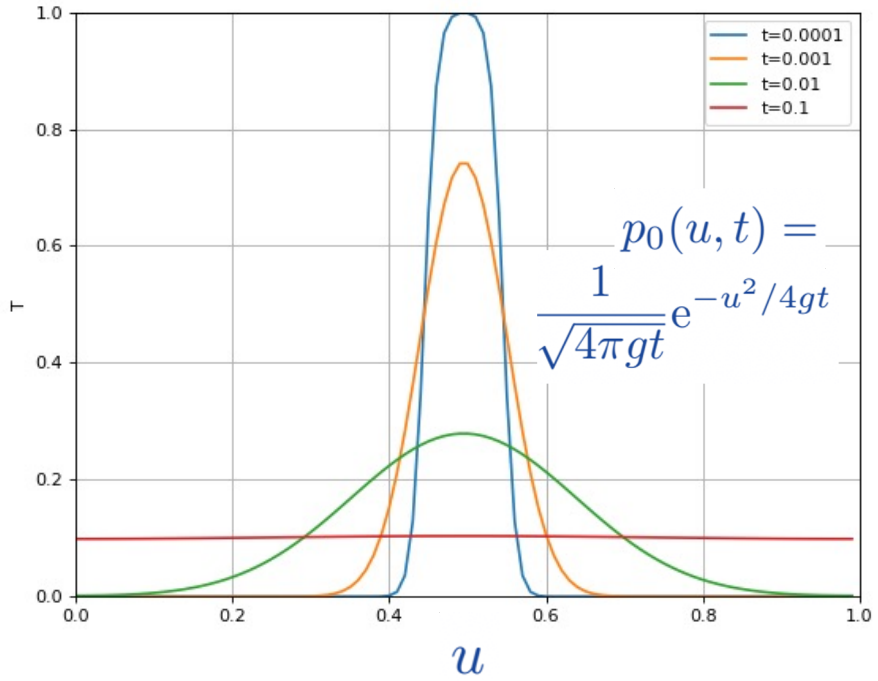
Solution de l'équation de diffusion lorsque g est constante

Pour une constante de diffusion g fixe $\dfrac{\partial x(u,t)}{\partial t} = g\,\mathrm{div}\,\nabla x(u,t) = g\,\Delta x(u,t)]$

$$p_0(u,t) = \frac{1}{\sqrt{4\pi g t}} \mathrm{e}^{-u^2/4gt}$$

$$\mathbb{E}[u^2] = \int u^2 p_0(u,t)\mathrm{d}u = 2d\,g\,t$$

Diffusion : distance moyenne parcourue en un temps $t \approx \sqrt{t}$



La solution générale s'obtient par linéarité $x(u,t) = \int x_0(v)\,p_0(u-v,t)\,\mathrm{d}v$

Diffusion sur une variété riemannienne

Graphe = discrétisation spatiale d'une **variété riemannienne** Ω (= surface généralisée avec une notion de distance locale). Permet de rendre naturelles les définitions du gradient ∇ et de la divergence $\nabla \cdot$ sur un graphe.

Fonctions scalaires sur l'espace de base Ω : $x \in \chi(\Omega)$ température, densité, pression, feature,…

Fonctions vectorielles sur l'espace de base Ω : $\mathscr{X} \in \chi(T\Omega)$ flux, direction du vent, …

Opérateur **gradient** : $\nabla : \chi(\Omega) \to \chi(T\Omega)$

Opérateur **divergence** : $\operatorname{div} : \chi(T\Omega) \to \chi(\Omega)$

Pour un espace euclidien $\Omega = \mathbb{R}^d$ on a :

$$\int_{\mathbb{R}^d} \nabla x(u) \cdot \mathscr{X}(u)\,\mathrm{d}u = \int_{\mathbb{R}^d} x(u) \operatorname{div} \mathscr{X}(u)\,\mathrm{d}u$$

Pour une variété riemannienne, on a aussi :
(à condition d'utiliser les bonnes définitions)

$$\langle\!\langle \nabla x, \mathscr{X} \rangle\!\rangle = \langle x, \operatorname{div} \mathscr{X} \rangle$$

Gradient et divergence sur un graphe

Analogues des définitions pour un graphe $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ $\quad |\mathscr{V}| = n \quad |\mathscr{E}| = e$

features de **nœud** : $x : \mathscr{V} \ni i \rightarrow x_i \in \mathbb{R}$ $\quad x \in \mathbb{R}^n$

features de **lien** : $\mathscr{X} : \mathscr{E} \ni (ij) \rightarrow \mathscr{X}_{ij} \in \mathbb{R}$ $\quad \mathscr{X} \in \mathbb{R}^e$ $\quad \mathscr{X}_{ji} = -\mathscr{X}_{ij}$

un nombre attribué à chaque direction j définie un lien du graphe à partir d'un nœud i.

$$(\nabla x)_{ij} := x_j - x_i$$

$$(\mathrm{div}(\mathscr{X}))_i := \sum_{j:(ij)\in\mathscr{E}} \mathscr{X}_{ij} = \sum_{j=1}^{n} w_{ij}\mathscr{X}_{ij}$$

$$\langle x, y\rangle := \sum_{i\in\mathscr{V}} x_i\, y_i$$

$$\langle\langle \mathscr{X}, \mathscr{Y}\rangle\rangle := \sum_{(ij)\in\mathscr{E}} \mathscr{X}_{ij}\mathscr{Y}_{ij} = \sum_{i>j} w_{ij}\mathscr{X}_{ij}\mathscr{Y}_{ij}$$

matrice d'adjacence

$$\Rightarrow \quad \langle\langle \nabla x, \mathscr{X}\rangle\rangle = \langle x, \mathrm{div}\,\mathscr{X}\rangle$$

Diffusion sur un graphe

$$\frac{\partial x(t)}{\partial t} = \operatorname{div}\left[\mathbf{G}(x(t),t)\,\nabla x(t)\right] \qquad \text{avec} \qquad \mathbf{G}(x,t) = \mathbf{G}(x) := \operatorname{diag}\left([a(x_i,x_j)]_{(ij)\in\mathscr{E}}\right) \in \mathbb{R}^{e\times e}$$

En introduisant les définitions de div et ∇ et en définissant une **matrice d'attention** on obtient la forme matricielle de l'équation de diffusion sur un graphe :

$$\mathbf{A}(x) := \left[\ \underbrace{a(x_i,x_j)}_{:=0 \text{ si } (ij)\notin\mathscr{E}}\ \right]_{i,j\in\mathscr{V}} \in \mathbb{R}^{n\times n}$$

on choisit une matrice stochastique (à droite) : somme par ligne des éléments = 1.

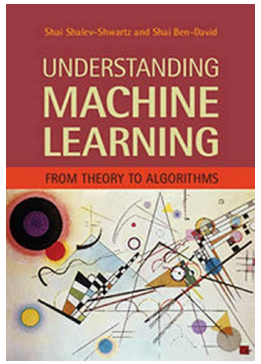
$$\boxed{\frac{\partial x(t)}{\partial t} = \left[\mathbf{A}(x(t)) - \mathbf{I}\right] x(t)}$$

Pour une matrice constante $\mathbf{A}(x(t)) = \mathbf{A}$ la solution s'écrit $x(t) = \mathrm{e}^{(\mathbf{A}-\mathbf{I})t}x(0)$

Stabilité en ML et dans l'équation de diffusion

Notion de **stabilité en ML** : un algorithme est stable si la fonction de coût ne varie pas beaucoup lorsqu'on remplace une observation par une autre dans le train set.

Un **algorithme stable généralise** : l'erreur empirique (sur le train set) est proche de l'erreur théorique (espérance sur la population).



voir le théorème 13.2 p139

PDF

Notion générale de **stabilité pour une EDP**

$$\forall \epsilon > 0, \ \exists \delta > 0 \quad \text{tel que} \quad |x(0) - \hat{x}(0)| \leq \delta \quad \Rightarrow \quad |x(t) - \hat{x}(t)| \leq \epsilon, \quad \forall t \geq 0$$

Pour la **diffusion sur un graphe** on montre seulement qu'en général

$$\min_{j \in \mathcal{V}} x_j(0) \leq x_i(t) \leq \max_{j \in \mathcal{V}} x_j(0) \quad \forall t \geq 0$$

Schémas de résolution explicites et implicites

$$\frac{\partial x(t)}{\partial t} = [\mathbf{A}(x(t)) - \mathbf{I}]\, x(t)$$

à discrétiser dans le temps

Discrétisation **explicite**

compromis à trouver entre le nombre d'itérations K et le pas τ

$$\frac{x^{(k+1)} - x^{(k)}}{\tau} = \underbrace{\left[\mathbf{A}(x^{(k)}) - \mathbf{I}\right]}_{:= \bar{\mathbf{A}}(x^{(k)})} x^{(k)}$$

correspond approximativement à la règle de propagation d'un GAT avec une skip-connexion

temps discrétisé correspond au n° d'une couche de convolution d'un GNN.

$$x^{(k+1)} = \left[\mathbf{I} + \tau \bar{\mathbf{A}}(x^{(k)})\right] x^{(k)}$$

trouver un compromis entre la nb K d'itérations et le pas d'intégration τ.

schéma de calcul **stable seulement** pour $0 < \tau < 1$

Discrétisation **implicite**

$$\frac{x^{(k+1)} - x^{(k)}}{\tau} = \left[\mathbf{A}(x^{(k)}) - \mathbf{I}\right] x^{(k+1)}$$

matrice à inverser = prix à payer pour la stabilité

$$x^{(k)} = \left[\mathbf{I} - \tau \bar{\mathbf{A}}(x^{(k)})\right] x^{(k+1)}$$

schéma de calcul **inconditionnellement stable** pour tout $\tau > 0$

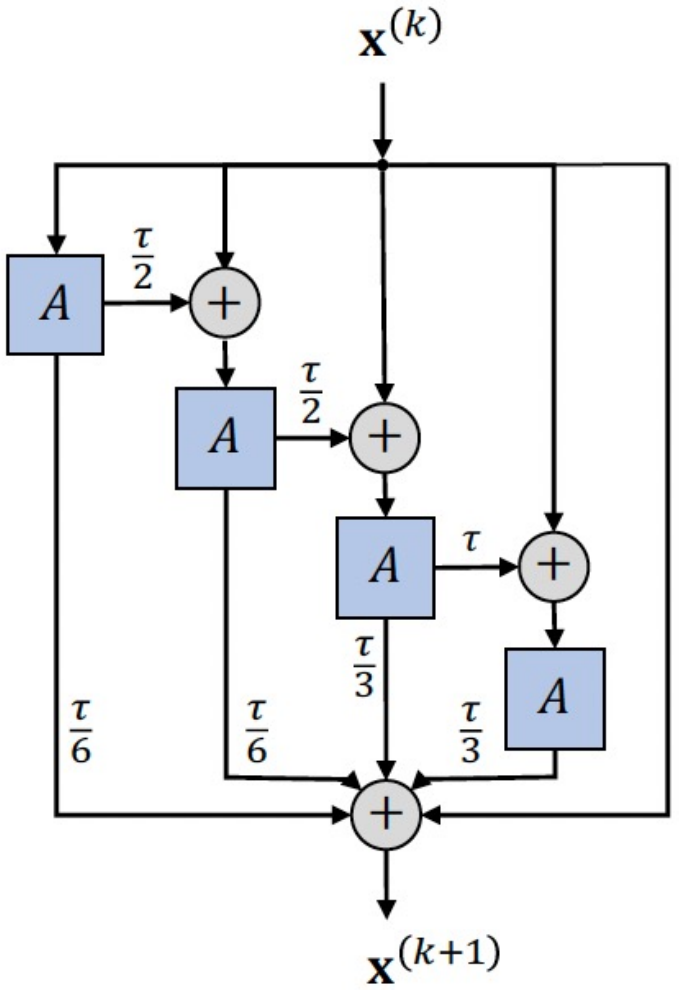
Schémas de résolution multi-étapes

Schéma **Runge-Kutta 4** : utilisation de temps intermédiaire
Revient à définir des skip-connexions particulières dont l'efficacité a été démontrée en pratique.

$$f(\mathbf{x}, t) = \bar{A}(\mathbf{x}_t)\mathbf{x}_t$$

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \frac{1}{6}\tau\left(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right)$$

$$\mathbf{k}_1 = f(\mathbf{x}_t, t)$$
$$\mathbf{k}_2 = f(\mathbf{x}_t + \tau\mathbf{k}_1/2, t + \tau/2)$$
$$\mathbf{k}_3 = f(\mathbf{x}_t + \tau\mathbf{k}_2/2, t + \tau/2)$$
$$\mathbf{k}_4 = f(\mathbf{x}_t + \tau\mathbf{k}_3, t + \tau)$$



Le modèle : Graph Neural Diffusion (GRAND)

Comme pour PPNP on va découpler l'apprentissage de représentations locales par un modèle **non-linéaire de type MLP** de leur **propagation** à travers le graphe.

$$Z_{\mathrm{GCN}} = \mathrm{softmax}\left(P\sigma \ldots \left(P\,\sigma(PXW^{(0)})\,W^{(1)}\right)\ldots W^{(L)}\right)$$

$$Z_{\mathrm{PPNP}} = \mathrm{softmax}\left(P_{\mathrm{ppr}}\; f_\theta(X)\right) \quad \text{où} \quad P_{\mathrm{ppr}} := \alpha\left[\mathbb{I} - (1-\alpha)P\right]^{-1}$$

3 — 2 propagation — MLP 1

Le modèle **GRAND**

Equation non linéaire si $\mathbf{A}$ **dépend de** $\mathbf{X}$.

1 $\quad \mathbf{X}(0) = \phi(\mathbf{X}_{\mathrm{in}}) \in \mathbb{R}^{n\times d}$

2 $\quad \mathbf{X}(T) = \mathbf{X}(0) + \int_0^T \frac{\partial \mathbf{X}(t)}{\partial t}\mathrm{d}t \quad$ où $\quad \frac{\partial \mathbf{X}(t)}{\partial t} = [\mathbf{A}(\mathbf{X}) - \mathbf{I}]\,\mathbf{X}$

3 $\quad \mathbf{Y} = \psi(\mathbf{X}(T))$

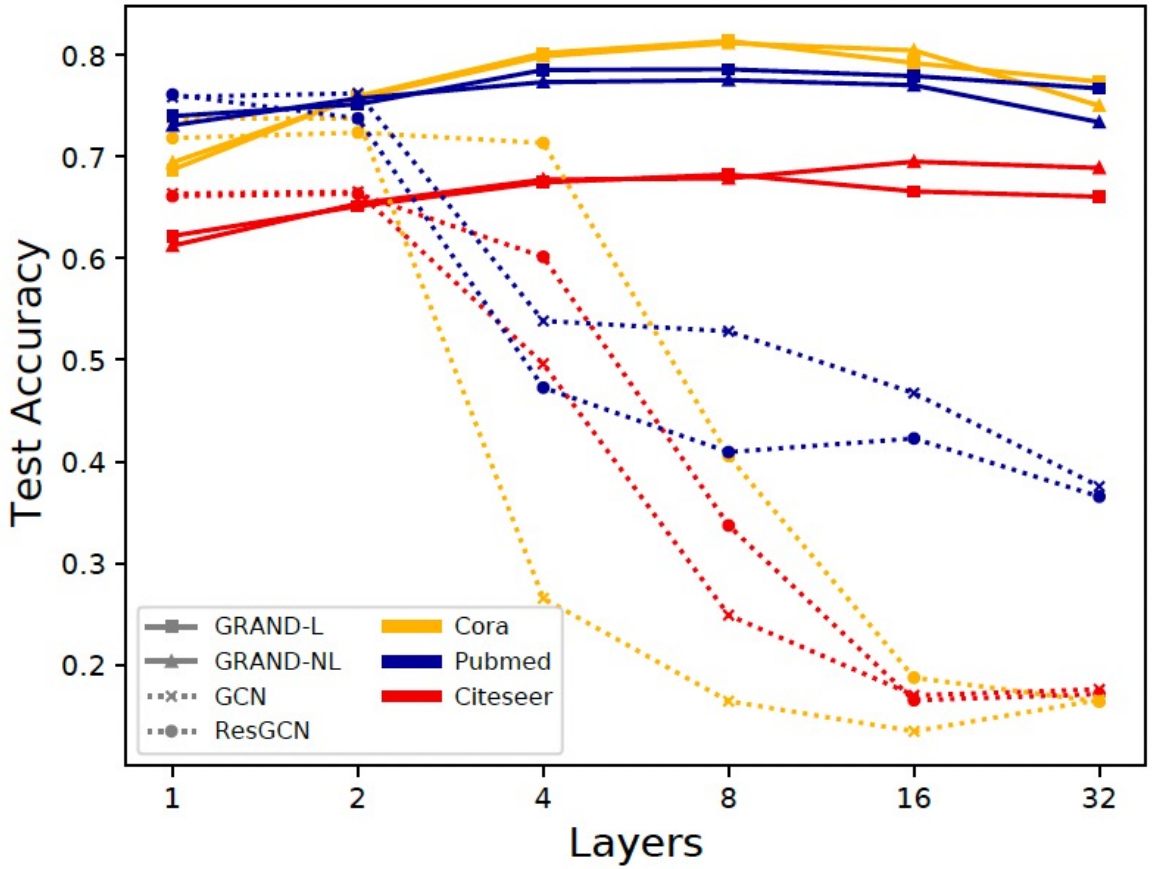
$$\mathbf{A}(\mathbf{X}) := [a(\mathbf{X}_i, \mathbf{X}_j)]_{i,j\in\mathcal{V}}$$

$$a(\mathbf{X}_i, \mathbf{X}_j) := \mathrm{softmax}\left(\frac{(\mathbf{W}_K\mathbf{X}_i)^\top \mathbf{W}_Q\mathbf{X}_j}{d_k}\right)$$

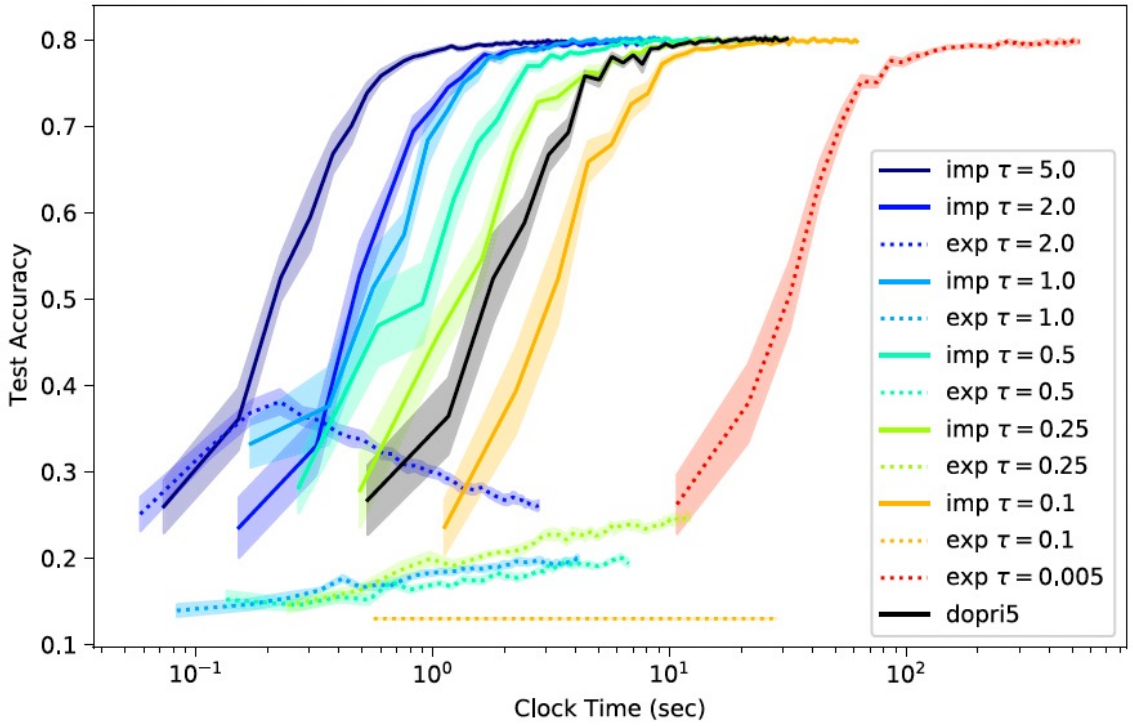
Choix à effectuer pour la propagation par diffusion :

- Matrice de diffusion $\mathbf{G}(\mathbf{X})$ $\leftrightarrow$ matrice d'attention $\mathbf{A}(\mathbf{X})$.
- Choix d'une méthode de discrétisation temporelle de l'équation de diffusion.

Résultats et comparatifs avec les GNN classiques



Robustesse à l'oversmoothing en fonction du nombre K de couches/pas d'intégration avec τ=1.



Stabilité des schémas implicites et explicites en fonction de la taille du pas d'intégration τ

Conclusion

- **GRAND** est une incarnation concrète de la stratégie qui consiste à transférer des techniques, mathématiques et numériques, développée pour la physique à l'apprentissage de représentations pour les graphes.
- C'est une partie d'une ambition beaucoup plus vaste qui entend convoquer la théorie des EDP, le control optimal, la géométrie différentielle, la topologie algébrique pour refonder les GNN's.
- **GRAND** propose une méthode systématique pour construire de nouveau modèles d'apprentissage de représentations pour les graphes, avec des méthodes adaptatives par exemple.
- **GRAND** permet d'interpréter certains modèles existant comme des cas particuliers.
- Le choix d'une méthode d'intégration numérique (discrétisation) revient à définir des connexions résiduelles (skip-connexions) dont on a de bonnes raisons de penser qu'elles sont utiles.
- Les résultats de stabilité permettent de construire des modèles très profonds.
- Le modèle **GRAND** n'utilise qu'un nombre relativement faible de paramètres, ceux qui définissent l'unique mécanisme d'attention (au lieu d'un mécanisme par couche).

Question à 1 millions d'€ : S'agit-il d'un simple « rebranding chic » d'idées anciennes ? Une montagne mathématique a-t-elle accouché d'une souris au niveau des applications ?