

# Réseaux de Neurones Équivariants et Applications à la Prédiction de Couplage Moléculaire

Solal Attias

Septembre 2023

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>La notion d'équivariance</b>	<b>3</b>
<b>3</b>	<b>Architectures de réseaux de neurones équivariants</b>	<b>3</b>
3.1	Équivariance des réseaux neuronaux convolutifs . . . . .	3
3.2	Réseaux de convolution groupe-équivariants . . . . .	4
3.3	Réseaux neuronaux de graphe équivariants . . . . .	4
3.3.1	Réseaux de neurones de graphe . . . . .	4
3.3.2	Réseaux neuronaux de graphe $E(n)$ -équivariants . . . . .	5
3.3.3	Propriété $E(n)$ -équivariante de ce réseau . . . . .	5
<b>4</b>	<b>Application des réseaux équivariants au couplage moléculaire</b>	<b>6</b>
4.1	Présentation du problème du couplage moléculaire . . . . .	6
4.2	L' $E(n)$ -équivariance dans le problème de couplage . . . . .	7
4.3	EQUIBIND, un modèle SE(3)-équivariant de prédiction de couplage moléculaire . . . . .	7
4.3.1	Résultats d'EQUIBIND . . . . .	9
4.4	E3BIND, un modèle à la pointe . . . . .	9
<b>5</b>	<b>Conclusion</b>	<b>10</b>

# 1 Introduction

Le cheval de bataille de l'industrie pharmaceutique dans la recherche de médicaments reste, après presque un siècle, des petites molécules, ciblant le plus souvent des protéines, impliqués dans le processus morbide [1]. Leur effet vient de la perturbation de cette protéine, en venant se lier physiquement sur un lieu clé de celle-ci. Ainsi, comprendre la structure du couplage de la molécule (appelée aussi ligand) avec la protéine est extrêmement important dans la découverte de médicaments, problème que nous appellerons dans la suite, prédiction de couplage moléculaire.

Les modèles classiques de couplage moléculaire reposent sur la génération d'un très large nombre de conformations (structures 3D potentielles) de la molécule avec la protéine, qui sont ensuite notées par une fonction d'évaluation inspirée de la physique, et affinées en dernière étape [2]. Le problème principal de ces modèles est le coût en calcul considérable, nécessitant entre 2 et 40 secondes par ligand, même en utilisant des GPUs puissant pour accélérer les calculs [3], sachant que l'espace chimique à explorer est immense, de l'ordre de  $10^{60}$  molécules [4]. Il y a donc un grand intérêt dans la création de modèles d'apprentissage automatique qui puissent prédire beaucoup plus rapidement que ces modèles classiques le couplage moléculaire. Par exemple, récemment, des algorithmes d'apprentissage automatique ont permis d'explorer le couplage moléculaire entre 15 mille protéines humaines et un espace chimique synthétisable de plusieurs dizaines de milliards de molécules [5].

La pauvreté relative de données est cependant un obstacle de taille – seulement 19 mille couplage moléculaires mesurés expérimentalement sont recensés dans PDDBind [6], une base de données exhaustive, à comparer à 3.2 millions d'images dans ImageNet en 2009 [7], une base de données classique pour la reconnaissance d'images. Ainsi, capturer des symétries de ce système physique est essentiel pour pallier à ce manque de données, et à été l'un des principes cruciaux dans l'élaboration d'AlphaFold2, un modèle qui a révolutionné le problème de repliement de protéine [8]. Inspirés de cet algorithme charnière, l'utilisation de la notion d'équivariance est récurrente dans les modèles les plus performants [9, 10].

Nous allons donc dans un premier temps introduire cette notion d'équivariance pour un réseau de neurones. Nous présenterons ensuite quelques exemples d'architectures équivariantes pour des groupes de transformations plus ou moins larges, et finalement nous montrerons des exemples de modèles de prédiction de couplage moléculaire s'inspirant de ces architectures.

## 2 La notion d'équivariance

Soit  $\Phi$  un réseau de neurones (ou réseau neuronal) ou une couche d'un réseau de neurones, ici tous les deux modélisés comme une fonction, d'un espace vectoriel  $X$  dans un espace vectoriel  $Y$ , et  $G$  un groupe qui agit sur  $X$  et admet une représentation linéaire  $T$  dans  $Y$ .  $\Phi$  est alors  $G$ -équivariant si :

$$\Phi(g \cdot x) = T(g)\Phi(x), \quad \forall x \in X, g \in G$$

Contrairement aux techniques dites de *data augmentation*, où on rajoute explicitement des exemples d'équivariance dans les données d'entraînement afin que le réseau de neurones apprenne à être approximativement équivariants, le but est de concevoir des architectures qui *par construction* sont équivariantes pour le groupe d'intérêt. Par exemple, confrontés à des images, si on souhaite que notre réseau de neurones reconnaisse ces images si elles sont tournées, on pourrait faire de la *data augmentation* en démultipliant la taille de nos données pour y inclure des images tournées aléatoirement, ou on pourrait concevoir de prime-abord notre réseau de neurones pour qu'il soit  $SO(2)$ -équivariant. Ceci permet de réduire le temps d'entraînement, la taille de notre réseau de neurones (car les poids sont mieux "partagés") et bien sûr garantit l'équivariance voulue [11, 12].

Un cas particulier est celui de *l'invariance par G*, où  $T$  est alors la représentation triviale,  $T = id_Y$ , mais le cas où  $T$  est fidèle (i.e.  $T$  est injectif) est souvent plus intéressant, car cela permet de s'assurer que les variables de représentation apprises sont dans des bonnes configurations relatives.

## 3 Architectures de réseaux de neurones équivariants

### 3.1 Équivariance des réseaux neuronaux convolutifs

Les réseaux neuronaux de convolution ou *convolutional neural network* (CNN en anglais), sont une classe de modèles de référence qui, en 2012, ont percés dans la reconnaissance d'images [13], et ont été déclencheurs pour la révolution de l'apprentissage profond. Une des propriétés clés qui est attribuée au succès de ces modèles est leur équivariance par translation. Nous allons ainsi, comme exemple introductif, illustrer ceci.

En entrée d'un réseau de convolution est une image (ici infinie et en noir et blanc, pour alléger la notation), vue comme une fonction  $f : \mathbb{Z}^2 \rightarrow \mathbb{R}$ . Cette image  $f$  est convoluée avec des filtres  $\phi_i : \mathbb{Z}^2 \rightarrow \mathbb{R}$ , pour obtenir des représentations en sortie :  $[f * \phi_i](x) = \sum_{y \in \mathbb{Z}^2} f(y) * \phi_i(x - y)$ . On a donc la couche de convolution suivante :

$$\begin{aligned} \Phi : \quad \mathbb{R}^{\mathbb{Z}^2} &\rightarrow \llbracket 1, K \rrbracket \times \mathbb{R}^{\mathbb{Z}^2} \\ f &\mapsto (f * \phi_i)_i \end{aligned} \tag{1}$$

Ainsi, l'action d'une translation de l'image,  $L_t f : x \mapsto f(x - t)$  (avec  $L'_t$  l'action analogue sur les tuples d'images) se traduit par :

$$\begin{aligned} \Phi(L_t f) &= ([L_t f * \phi_i](x))_{x,i} \\ &= \left( \sum_{y \in \mathbb{Z}^2} f(y - t) * \phi_i(x - y) \right)_{x,i} \\ &= \left( \sum_{y' \in \mathbb{Z}^2} f(y') * \phi_i((x - t) - y') \right)_{x,i} \\ &= ([f * \phi_i](x - t))_{x,i} \\ &= L'_t \Phi(f) \end{aligned} \tag{2}$$

Une couche convolutive est donc équivariante par translation. Dans les CNN, il existe aussi des couches de pooling et de correction mais leur équivariance par translation (si elles sont de taille de pas 1) est immédiate (voir [12] pour plus de détails). Ceci suffit à enfin prouver qu'un réseau de neurone convolutif est équivariant dans son ensemble.

## 3.2 Réseaux de convolution groupe-équivariants

Introduit dans [12], l'architecture du réseau de convolution groupe-équivariant permet de construire un réseau  $G$ -équivariant pour un groupe  $G$  sur les images de pixels (mais peut être généralisé aux images  $N$ -dimensionnelles). On définit alors une  $G$ -convolution sur la première couche ainsi :

$$[f * \phi_i](g) = \sum_{y \in \mathbb{Z}^2} f(y) * \phi_i(g^{-1}y) \text{ pour } g \in G$$

Pour les couches suivantes, on ne fait plus des filtres sur l'espace d'entrée du réseau, mais sur le groupe  $G$ . Ainsi, dans le cas des images, les représentations ne sont donc plus par pixels, mais par élément du groupe qui opère sur les images, et on prend donc une "image de groupe"  $f : G \rightarrow E$ . On a alors :

$$[f * \phi_i](g) = \sum_{h \in G} f(h) * \phi_i(h^{-1}g) \text{ pour } g \in G$$

Notons  $\pi$  la représentation de groupe de  $G$  sur l'espace des images de groupe :

$$\text{Si } g \in \text{Hom}(\mathbb{Z}^2), \text{ alors, } \pi(g)f(x) = f(g^{-1}x), \quad \forall x \in \mathbb{Z}^2$$

Notons  $\pi'$  celle analogue pour les tuples d'images de groupes (qui s'applique sur chaque élément du tuple).

La  $G$ -équivariance de cette opération est alors analogue à celle des CNNs : prenons  $u \in G$

$$\begin{aligned} \Phi(\pi(u)f) &= ([\pi(u)f * \phi_i](g))_{g,i} \\ &= \left( \sum_{h \in G} f(u^{-1}h) * \phi_i(h^{-1}g) \right)_{g,i} \\ &= \left( \sum_{h' \in G} f(h') * \phi_i(h'^{-1}u^{-1}g) \right)_{g,i} \\ &= ([f * \phi_i](u^{-1}g))_{g,i} \\ &= \pi(u)' \Phi(f) \end{aligned} \tag{3}$$

Les non-linéarités et pooling peuvent aussi être définies de manière  $G$ -équivariante, construisant ainsi une classe de réseaux de convolution  $G$ -équivariants.

Une implémentation efficace est possible sur les images si tout  $g \in G$  peut être décomposé en translation et un nombre fini de transformations pour laquelle l'origine est stable. Cependant, le coût de ce modèle peut devenir grand si le groupe de transformation est grand.

Cette architecture en a inspiré d'autres, qui tentent de généraliser cette architecture [14], ou l'appliquer à des problèmes concrets [15, 16].

## 3.3 Réseaux neuronaux de graphe équivariants

### 3.3.1 Réseaux de neurones de graphe

Une autre famille de réseaux de neurones équivariants sont ceux qui repose sur les réseaux de neurones de graphes [17], qu'on introduit ici :  
On a un graphe  $G = (V, E)$ , avec  $V$  l'ensemble des noeuds et  $E \subset V \times V$  l'ensemble des arêtes. Le

graphe est muni d'attributs sur les arêtes notés  $(a_{ij})_{i,j \in E}$ . On note les voisins du noeud  $i$ ,  $\mathcal{N}(i)$ . On va passer des messages  $(m_{ij})_{i,j \in E} \in \mathbb{R}^a$  sur les arêtes et  $(m_i)_{i \in V} \in \mathbb{R}^a$  sur les noeuds. La représentation du noeud  $i$  à la couche  $l$  sera  $h_i^l \in \mathbb{R}^f$ . Ces variables sont calculées dans le réseau par les formules suivantes :

$$\begin{aligned} m_{ij} &= \phi_e(h_i^l, h_j^l, a_{ij}) \\ m_i &= \sum_{j \in \mathcal{N}(i)} m_{ij} \\ h_i^{l+1} &= \phi_h(h_i^l, m_i) \end{aligned} \tag{4}$$

Avec  $\phi_e$  et  $\phi_h$  des fonctions apprises par le réseau par des couches de neurones complètement connectées (aussi appelées perceptron multi-couche ou MLP en anglais).

### 3.3.2 Réseaux neuronaux de graphe $E(n)$ -équivalariants

Les réseaux de neurones de graphes  $E(n)$ -équivalariants ou réseaux de neurones de graphes équivalariants ont été décrits dans [18]. Ici,  $E(n)$  est le groupe des isométries affines de  $\mathbb{R}^n$ , aussi appelé groupe euclidien.  $SE(n)$  quand à lui est le groupe des isométries affines spécial, sans les réflexions.

Dans la couche  $l$  de réseau de graphe  $E(n)$ -équivalente, qu'on appelle CRGE on rajoute par rapport aux réseaux de neurones de graphes une représentations de coordonnées par couche  $(x_i^l)_{i,l} \in \mathbb{R}^n$ , avec  $(x_i^0)_i$  les coordonnées des noeuds du graphe.

$$\begin{aligned} m_{ij} &= \phi_e(h_i^l, h_j^l, \|x_i^l - x_j^l\|^2, a_{ij}) \\ x_i^{l+1} &= x_i^l + \frac{1}{|V| - 1} \sum_{j \neq i} (x_i^l - x_j^l) \phi_x(m_{ij}) \\ m_i &= \sum_{j \in \mathcal{N}(i)} m_{ij} \\ h_i^{l+1} &= \phi_h(h_i^l, m_i) \end{aligned} \tag{5}$$

$\phi_x$  est, comme  $\phi_e$  et  $\phi_h$ , une fonction apprises par le réseau grâce à un MLP.

On notera que beaucoup d'autres architectures  $E(3)$ -équivalentes ont pris comme point de départ les réseaux neuronaux de graphe [19, 20, 21, 8].

### 3.3.3 Propriété $E(n)$ -équivalente de ce réseau

Prouvons que cette couche est  $E(n)$ -équivalente.

Soit  $f \in E(n)$ .  $f$  peut classiquement être décomposée en une isométrie affine avec points fixes, qui est un automorphisme orthogonal  $Q \in O(n)$ , et une sans point fixe, qui est une translation d'un vecteur  $g \in \mathbb{R}^n$ .

$f$  agit alors sur les coordonnées  $(x_i^0)_i$ . On cherche alors à montrer par induction sur  $l$ , que pour toute couche  $l$  dans le réseau :

$$Qx^{l+1} + g, h^{l+1} = \text{CRGE}(Qx^l + g, h^l)$$

Tout d'abord les  $(h_i^0)_i$  sont invariants par  $f$ , car on ne mets pas les coordonnées du graphe dans ces variables, et donc on vérifie l'équivalence à la couche 0.

Supposons que pour la couche  $l$  les variables  $(x^l)$  sont équivalentes et  $(h^l)$  sont invariantes. Alors, tout d'abord, les  $(m_{ij})$  sont invariants par l'action de  $f$  car par hypothèse d'induction les  $(h_i^l)_i$  sont invariants, et puisque  $f$  est une isométrie elle préserve les distances  $\|x_i^l - x_j^l\|$ . Directement nous avons en plus l'invariance des  $m_i$  et des  $h_i^{l+1}$ .

Il reste seulement les  $x_i^{l+1}$  :

$$\begin{aligned}
 Qx_i^l + g + \frac{1}{|V|-1} \sum_{j \neq i} (Qx_i^l + g - Qx_j^l - g) \phi_x(m_{i,j}) &= Qx_i^l + g + Q \frac{1}{|V|-1} \sum_{j \neq i} (x_i^l - x_j^l) \phi_x(m_{i,j}) \\
 &= Q \left( x_i^l + \frac{1}{|V|-1} \sum_{j \neq i} (x_i^l - x_j^l) \phi_x(m_{i,j}) \right) + g \\
 &= Qx_i^{l+1} + g
 \end{aligned} \tag{6}$$

Ce qui finit la preuve de  $E(n)$ -équivariance du réseau.

## 4 Application des réseaux équivariants au couplage moléculaire

### 4.1 Présentation du problème du couplage moléculaire

Une protéine est une séquence d'acides aminés, aussi appelés résidus, encodée par l'ADN. Chaque résidu a une structure générale, avec une chaîne latérale qui varie en fonction des 22 types d'acide aminé, et la structure 3D va dépendre de l'interaction des résidus entre eux et du milieu dans lequel la protéine se trouve. La protéine joue alors une ou plusieurs fonctions biologiques qui dépendent de cette structure 3D et de sa chimie, en interagissant avec diverses autres molécules (molécule non-protéiques ou d'autres protéines) à sa surface, sur des sites donnés, aussi appelés poches.

Il est alors possible pour un pharmaco-chimiste de cibler un site d'une protéine qui intervient dans le processus d'une maladie, et ainsi espérer altérer son fonctionnement, ce qui rend la compréhension de la structure 3D de la protéine essentielle. Celle-ci peut être mesurée expérimentalement en utilisant des techniques comme la cristallographie par rayon-X, ou inférée par un modèle – qu'il soit une simulation basée sur la physique (la *dynamique moléculaire*) ou alors un algorithme d'apprentissage automatique tel qu'AlphaFold [8].

Le plus souvent, le pharmaco-chimiste cherchera à créer une petite molécule (< 1000 masses atomiques unifiées) qui va interagir avec la protéine ciblée au niveau de la poche [1]. La prédiction de l'interaction d'une molécule avec la protéine, à partir de sa structure, devient alors un problème central dans l'élaboration de médicaments.

Une difficulté particulière est la flexibilité du ligand, ainsi que celle de la protéine. Alors que la flexibilité de la protéine est un facteur connu mais souvent non pris en compte pour des raisons de rapidité de calculs [1], la flexibilité du ligand est absolument vitale pour bien comprendre l'interaction ligand-protéine. En effet, une petite molécule a souvent de nombreuses conformations 3D possibles, et peut se mettre dans une de ces possibles conformations en interagissant avec la

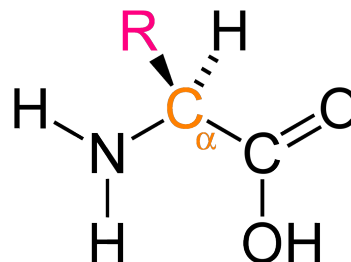


FIGURE 1 – Structure d'un acide aminé.  $R$  représente la chaîne latérale, et  $C_\alpha$  le carbone alpha. Les acides aminés dans la séquence s'attachent au niveau de l'oxygène à la place de l'hydrogène d'un côté (groupe carboxyle), et à l'azote à la place d'un des hydrogènes de l'autre (groupe amide), pour former une chaîne.

protéine, afin de faire former certaines liaisons chimique. La molécule pourra présenter des torsions autour de liaisons internes qui présentent un degré de liberté, lui permettant ainsi d’interagir de façon avantageuse avec la protéine, et donc de minimiser l’énergie du complexe ligand-protéine.

## 4.2 $L'E(n)$ -équivariance dans le problème de couplage

On remarquera que le problème de couplage, en prenant le ligand et la protéine en compte ensemble, présente une symétrie de rotation, de translation et de réflexion.  $L'E(n)$ -équivariance est alors une propriété naturelle à désirer dans tout réseau de neurone qui cherche à prédire le couplage moléculaire. Cependant, si le ligand et la protéine sont pris séparément, le problème ne présente plus de symétrie de réflexion, car généralement, ces molécules ne sont pas chirales (i.e. sont asymétriques par réflexion), et donc les réseaux devront ne pas être invariant par réflexion. En effet, les acides aminés sont asymétriques par réflexion, et sont présent sous une forme chirale (aussi appelée énantiomère), dans tout le monde biologique, phénomène appelé homochiralité [1].

Cette importance de l’asymétrie par réflexion s’illustre par le cas tristement célèbre du thalidomide [1], distribué dans les années 50 et 60 avec un mélange 50 : 50 des deux énantiomères, où un énantiomère était un excellent sédatif, mais l’autre causait des malformations fœtales, ce qui a tragiquement affecté des milliers de bébés.

## 4.3 EQUIBIND, un modèle SE(3)-équivariant de prédiction de couplage moléculaire

On présente ici EQUIBIND, un modèle de couplage moléculaire E(3)-équivariant.

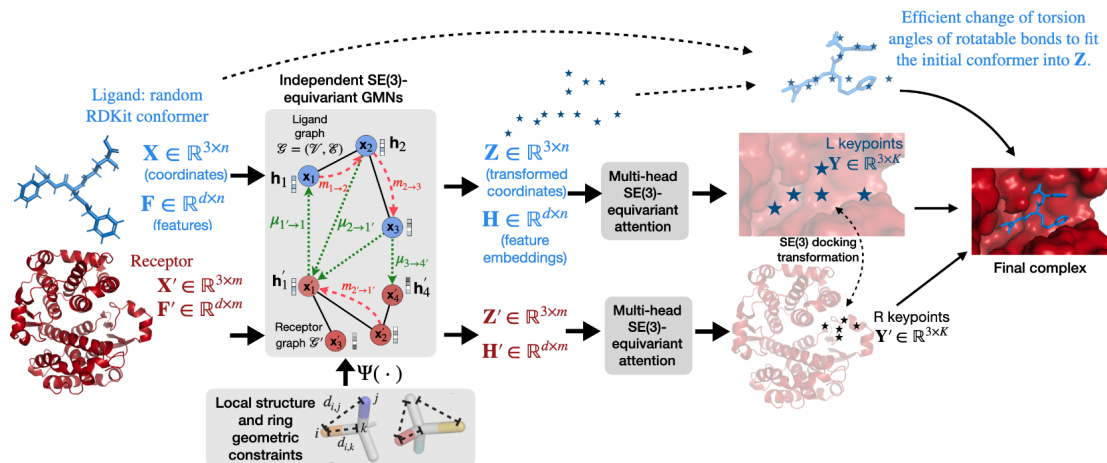


FIGURE 2 – Architecture d’EQUIBIND, Figure 2 de [9]

Le modèle prend en entrée un ligand et une protéine. Le ligand est représenté sous forme de graphe  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$   $k$ -plus proche voisins ( $k$ -NN en anglais), dont les  $n$  atomes sont les noeuds, associés à des coordonnées  $X \in \mathbb{R}^{3 \times n}$  et des descripteurs  $F \in \mathbb{R}^{d \times n}$ , et dont les arêtes sont les paires d’atomes à moins de 4Å. La protéine est aussi représentée sous forme de graphe  $k$ -NN  $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ , dont les  $m$  carbones  $\alpha$  des résidus sont les noeuds, associés à des coordonnées  $X' \in \mathbb{R}^{3 \times m}$  et des descripteurs  $F' \in \mathbb{R}^{d \times m}$ , et dont les arêtes sont les paires d’atomes à moins de 30Å, dans la limite de 10 voisins par noeud.

On définit alors une couche neuronale appelée Réseau Indépendant de Couplage de Graphe E(3)-équivariant (RICGE), pris de [22], et très proche de celui présenté en partie 3.3.2.

$$\begin{aligned}
m_{j \rightarrow i} &= \phi_e \left( h_i^{(l)}, h_j^{(l)}, \left\| x_i^{(l)} - x_j^{(l)} \right\|^2, f_{j \rightarrow i} \right), \forall (i, j) \in \mathcal{E} \cup \mathcal{E}' \\
\mu_{j' \rightarrow i} &= a_{j' \rightarrow i} W h_{j'}^{(l)}, \forall i \in \mathcal{V}, j' \in \mathcal{V}' \text{ or } i \in \mathcal{V}', j' \in \mathcal{V} \\
m_i &= \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} m_{j \rightarrow i}, \forall i \in \mathcal{V} \cup \mathcal{V}' \\
\mu_i &= \sum_{j' \in \mathcal{V}'} \mu_{j' \rightarrow i}, \forall i \in \mathcal{V}, \quad \text{and } \mu'_i = \sum_{j \in \mathcal{V}} \mu_{j \rightarrow i}, \forall i \in \mathcal{V}' \\
x_i^{(l+1)} &= \Psi \left( x_i^{(l)} + \sum_{j \in \mathcal{N}(i)} \frac{x_i^{(l)} - x_j^{(l)}}{\left\| x_i^{(l)} - x_j^{(l)} \right\|} \phi_x(m_{j \rightarrow i}) \right) \\
h_i^{(l+1)} &= (1 - \beta) \cdot h_i^{(l)} + \beta \cdot \phi_h \left( h_i^{(l)}, m_i, \mu_i, f_i \right), \forall i \in \mathcal{V} \cup \mathcal{V}'
\end{aligned} \tag{7}$$

Les coefficients d'attention  $a_{j \rightarrow i}$  sont calculés de la façon suivante :

$$a_{j \rightarrow i} = \frac{\exp \left( \left\langle \phi_q \left( \mathbf{h}_i^{(l)} \right), \phi_k \left( \mathbf{h}_j^{(l)} \right) \right\rangle \right)}{\sum_{j'} \exp \left( \left\langle \phi_q \left( \mathbf{h}_i^{(l)} \right), \phi_k \left( \mathbf{h}_{j'}^{(l)} \right) \right\rangle \right)}$$

$W$  est une matrice orthogonale paramétrisable,  $\phi_e$ ,  $\phi_x$ ,  $\phi_h$ ,  $\phi_q$  et  $\phi_k$  sont des fonctions apprises par des MLP, et  $\Psi$  une fonction équivariante choisie pour introduire de la flexibilité.

La preuve de l' $E(3)$ -équivariance de ce réseau est analogue à celle présentée en partie 3.3.3.

En sortie de ce réseau, on obtient alors  $Z \in \mathbb{R}^{3 \times n}$  et  $H \in \mathbb{R}^{d \times n}$  des nouvelles coordonnées et descripteurs pour le ligand, et  $Z' \in \mathbb{R}^{3 \times m}$  et  $H' \in \mathbb{R}^{d \times m}$  des nouvelles coordonnées et descripteurs pour la protéine.  $Z$  est alors sensé être les coordonnées de la nouvelle pose 3D du ligand.

On donne alors  $(Z, H)$  et  $(Z', H')$  à un mécanisme d'attention à multi-têtes  $SE(3)$ -équivariant, qui donne en sortie des "point-clés"  $Y \in \mathbb{R}^{3 \times K}$  et  $Y' \in \mathbb{R}^{3 \times K'}$ . L'espoir est que la transformation dans  $SE(3)$  qui prend  $Y$  et donne  $Y'$  (qui dans ce cas idéal existerait), serait la même qui mettrait  $Z$  dans la poche de la protéine.  $Y$  et  $Y'$  représenterait alors des points clés où les liens chimiques se forment, et en particulier  $Y'$  identifie la poche de la protéine. Ce mécanisme est décrit par les formules suivantes :

$$y_k := \sum_{i=1}^n \alpha_i^k z_i, \quad y'_k := \sum_{j=1}^m \beta_j^k z'_j$$

Avec  $\alpha_i^k$  et  $\beta_j^k$  des coefficients appris qui somment à 1 sur  $i$ . Ce mécanisme est clairement  $SE(3)$ -équivariant. Pour pouvoir aider le réseau à apprendre ces points clés, nous prenons comme référence les points médians des segments reliant les atomes du ligand aux atomes du récepteur qui sont plus proches que  $4\text{\AA}$ , qu'on appelle  $P$ . Il y ensuite une perte de transport optimal discret  $(Y, Y')$  à  $P$  pour aider à guider le réseau pendant l'entraînement.

Finalement, on trouve la roto-translation de  $Y$  à  $Y'$  grâce à une version différentiable de l'algorithme de Kabsch [23, 24, 25] :

Soit  $\mathbf{A} = Y'^T Y \in \mathbb{R}^{3 \times 3}$ . La décomposition en valeur singlière est  $\mathbf{A} = \mathbf{U}_2 \mathbf{S} \mathbf{U}_1^T$ , où  $\mathbf{U}_2, \mathbf{U}_1 \in O(3)$ . Finalement, on définit les fonctions différentiables :

$$\begin{aligned}
\mathbf{R} &= \mathbf{U}_2 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & d \end{pmatrix} \mathbf{U}_1^T, \quad \text{où } d = \text{sign}(\det(\mathbf{U}_2 \mathbf{U}_1^T)) \\
\mathbf{t} &= \mu(Y') - \mathbf{R} \mu(Y),
\end{aligned}$$

Avec  $\mu$  la fonction qui associe à un ensemble de vecteurs le vecteur moyen. Ceci est trivialement  $SE(3)$ -équivariant.

La pose de couplage est alors la succession de torsions des liens atomiques dans le ligand afin qu’il puisse être dans la pose la plus proche de celle décrite par  $Z$  (étape résolue analytiquement), et la roto-translation que l’on vient de trouver.

Lors de l’entraînement, on cherchera alors à minimiser la combinaison de la distance entre la pose prédite et celle expérimentalement mesurée, ainsi que de la perte du transport optimal, capturant ainsi si  $Y$  et  $Y'$  ne correspondent pas à des interactions clés.

### 4.3.1 Résultats d’EQUIBIND

Les résultats d’EQUIBIND présentés en figure 3 montrent qu’EQUIBIND trouve assez souvent une pose qui est (par rapport aux algorithmes classiques) relativement proche de la pose mesurée, en donnant des poses bien moins absurdes au 75e pourcentile, indiquant sans doute qu’EQUIBIND arrive à identifier la poche au niveau de la surface de la protéine. EQUIBIND permet aussi d’améliorer les performances des algorithmes classiques qui viennent affiner les résultats. On peut par ailleurs souligner l’efficacité remarquable de l’algorithme, qui est plus de 3000x plus rapide que les algorithmes classiques, grâce à l’architecture  $SE(3)$ -équivariante qui permet d’utiliser un petit nombre de paramètre tout en gardant une grande expressivité. Cependant, les résultats restent meilleurs avec les algorithmes classiques, et ceci ne permet pas de s’en affranchir – sûrement dû à la difficulté de prédire à la fois la poche, les points clés, et les torsions en une seule fois.

FIGURE 3 – Résultats d’Équibind lors d’une épreuve de docking sans poche identifiée, Table 1 de [9]. QVINA-W, GNINA, SMINA et GLIDE (c.) sont des algorithmes de couplage moléculaire classiques. EQUIBIND-U est EQUIBIND sans correction du ligand pour les torsions des liens atomiques (i.e. la pose  $Z$ ). EQUIBIND+Q et EQUIBIND+Q2 utilisent le modèle classique QUICKVINA2 pour affiner les poses, avec EQUIBIND+Q2 qui a des paramètres plus exhaustifs et EQUIBIND+S utilise SMINA pour affiner les poses.

METHODS	AVG. SEC.		LIGAND RMSD ↓						CENTROID DISTANCE ↓						KABSCH	
	16-CPU	GPU	PERCENTILES ↓				% BELOW THRESHOLD ↑		PERCENTILES ↓				% BELOW THRESH. ↑		RMSD ↓	
			MEAN	25TH	50TH	75TH	5 Å	2 Å	MEAN	25TH	50TH	75TH	5 Å	2 Å	MEAN	MED.
QVINA-W	49	-	13.6	2.5	7.7	23.7	40.2	20.9	11.9	0.9	3.7	22.9	54.6	41.0	<b>2.1</b>	1.9
GNINA	247	146	13.3	2.8	8.7	22.1	37.1	21.2	11.5	1.0	4.5	21.2	52.0	36.0	2.2	1.8
SMINA	146	-	12.1	3.8	8.1	17.9	33.9	13.5	9.8	1.3	3.7	16.2	55.9	38.0	2.2	1.9
GLIDE (c.)	1405*	-	16.2	2.6	9.3	28.1	33.6	21.8	14.4	<b>0.8</b>	5.6	26.9	48.7	36.1	2.2	1.9
EQUIBIND	<b>0.16</b>	<b>0.04</b>	<b>8.2</b>	3.8	6.2	<b>10.3</b>	39.1	5.5	<b>5.6</b>	1.3	2.6	7.4	67.5	40.0	2.6	2.3
EQUIBIND+Q	8	8	8.4	2.6	6.6	11.1	38.0	18.7	5.9	1.0	2.5	6.4	68.7	44.6	2.3	1.9
EQUIBIND+Q2	15	15	8.7	2.6	6.8	11.1	40.7	21.6	6.0	1.0	2.4	6.6	70.1	42.7	2.2	<b>1.6</b>
EQUIBIND+S	146	146	8.3	<b>2.1</b>	<b>5.6</b>	10.5	<b>46.4</b>	<b>24.6</b>	6.0	0.9	<b>2.0</b>	<b>6.2</b>	<b>71.0</b>	<b>50.6</b>	<b>2.1</b>	1.8
EQUIBIND-U	0.14	0.02	7.8	3.3	5.7	9.7	42.4	7.2	5.6	1.3	2.6	7.4	67.5	40.0	2.1	1.8

L’absurdité de beaucoup de poses générées par ce modèle, illustré par [26], reste un problème, et entrave sans doute la performance sans affinage.

## 4.4 E3BIND, un modèle à la pointe

Nous aimerions finalement souligner le papier de Zhang et al. [10], qui construit le modèle E3BIND, une autre architecture, comme le nom l’indique bien,  $E(3)$ -équivariante. Celle-ci surperforme les algorithmes classiques de couplage (en tout cas sur leurs tests), même sans affinage successif. Le prix est une vitesse 20 fois moindre qu’EQUIBIND.

Nous n'allons pas aller dans les détails car mathématiquement le réseau reste relativement similaire à EQUIBIND, mais nous allons la résumer rapidement afin de mettre en évidence les quelques points qui le distingue d'EQUIBIND.

- Le réseau commence par un élément appelé Trioformer, une attention multi-têtes similaire à celle en deuxième étape d'EQUIBIND, mais qui ici mélange les informations entre ligand et protéine.
- Il continue avec un réseau de graphe équivariant, aussi relativement similaire à celui présenté en partie 3.3.2. Ces deux étapes sont donc dans un ordre inverse d'EQUIBIND.
- La pose est alors donnée par les coordonnées en sortie de ce réseau neuronal de graphe, avec éventuellement un traitement de la torsion, qui est plus subtil que le calcul analytique fait en dernière étape d'EQUIBIND (mais au prix d'une vitesse 20 fois moindre, qui vaut le coût ici car le modèle est aussi plus lent), qui vient de [27].
- Finalement, il finit avec un MLP qui prend les descripteurs en sortie du réseau neuronal de graphe et calcule une confiance estimée.

La perte durant l'entraînement est une combinaison de la distance de la pose prédite à celle mesurée, ainsi que la distance entre la confiance prédite par le modèle et une "confiance réelle" qui diminue si la distance de la pose prédite à celle mesurée augmente.

La prédiction confiance est une raison primordiale dans le succès de ce modèle, car le rajout de celle-ci fait passer de 13.8% de poses prédites avec des absurdités (*conflits stériques*) à 3.3%.

## 5 Conclusion

Nous avons expliqués en quoi les réseaux de neurones équivariants sont importants dans l'étude de systèmes qui présentent des symétries cruciales, en permettant d'avoir des modèles efficaces, qui sont petits et rapides tout en étant performant.

Après avoir montré quelques exemples de réseaux de neurones équivariants pour différents groupes, nous avons introduit le cas d'étude du problème de couplage moléculaire, problème vital dans la découverte de nouveaux médicaments. Nous avons vu que les modèles de pointe du domaine adoptent ces architectures  $SE(3)$ -équivariantes, et peuvent sur-performer sur certaines métriques les modèles classiques inspirés de la physique, tout en étant beaucoup plus rapides.

La recherche d'architectures équivariantes pour différents groupes, et en particulier pour le groupe  $SE(3)$ , tout en restant efficacement implémentables reste très active. La création de modèles qui tentent de résoudre plus rapidement et avec plus de précision le problème de couplage moléculaire se nourrit de cette recherche, et use de techniques de plus en plus subtiles pour utiliser au mieux les données expérimentales disponibles.

## Références

- [1] G. L. Patrick, *Chimie pharmaceutique*. de boeck, 2 ed., Nov. 2002.
- [2] C. Yang, E. A. Chen, and Y. Zhang, “Protein–Ligand Docking in the Machine-Learning Era,” *Molecules*, vol. 27, p. 4568, Jan. 2022.
- [3] D. Santos-Martins, L. Solis-Vasquez, A. F. Tillack, M. F. Sanner, A. Koch, and S. Forli, “Accelerating AutoDock4 with GPUs and Gradient-Based Local Search,” *Journal of Chemical Theory and Computation*, Jan. 2021.
- [4] J.-L. Reymond and M. Awale, “Exploring Chemical Space for Drug Discovery Using the Chemical Universe Database,” *ACS Chemical Neuroscience*, vol. 3, pp. 649–657, Sept. 2012.
- [5] “Recursion Pharmaceuticals, Inc. - Recursion Bridges the Protein and Chemical Space with Massive Protein-Ligand Interaction Predictions Spanning 36 Billion Compounds.” <https://ir.recursion.com/news-releases/news-release-details/recursion-bridges-protein-and-chemical-space-massive-protein/>.
- [6] Z. Liu, Y. Li, L. Han, J. Li, J. Liu, Z. Zhao, W. Nie, Y. Liu, and R. Wang, “PDB-wide collection of binding data : Current status of the PDBbind database,” *Bioinformatics*, vol. 31, pp. 405–412, Feb. 2015.
- [7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet : A large-scale hierarchical image database,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, June 2009.
- [8] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Židek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis, “Highly accurate protein structure prediction with AlphaFold,” *Nature*, vol. 596, pp. 583–589, Aug. 2021.
- [9] H. Stärk, O.-E. Ganea, L. Pattanaik, R. Barzilay, and T. Jaakkola, “EquiBind : Geometric Deep Learning for Drug Binding Structure Prediction,” June 2022.
- [10] Y. Zhang, H. Cai, C. Shi, B. Zhong, and J. Tang, “E3Bind : An End-to-End Equivariant Network for Protein-Ligand Docking,” June 2023.
- [11] F. Quiroga, F. Ronchetti, L. Lanzarini, and A. F. Bariviera, “Revisiting Data Augmentation for Rotational Invariance in Convolutional Neural Networks,” in *Modelling and Simulation in Management Sciences* (J. C. Ferrer-Comalat, S. Linares-Mustarós, J. M. Merigó, and J. Kacprzyk, eds.), vol. 894, pp. 127–141, Cham : Springer International Publishing, 2020.
- [12] T. S. Cohen and M. Welling, “Group Equivariant Convolutional Networks,” June 2016.
- [13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Communications of the ACM*, vol. 60, pp. 84–90, May 2017.
- [14] T. S. Cohen and M. Welling, “Steerable CNNs,” Dec. 2016.
- [15] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow, “Harmonic Networks : Deep Translation and Rotation Equivariance,” Apr. 2017.
- [16] E. J. Bekkers, M. W. Lafarge, M. Veta, K. A. Eppenhof, J. P. Pluim, and R. Duits, “Roto-Translation Covariant Convolutional Networks for Medical Image Analysis,” June 2018.
- [17] B. Sanchez-Lengeling, E. Reif, A. Pearce, and A. B. Wiltschko, “A Gentle Introduction to Graph Neural Networks,” *Distill*, vol. 6, p. e33, Sept. 2021.
- [18] V. G. Satorras, E. Hoogeboom, and M. Welling, “E(n) Equivariant Graph Neural Networks,” Feb. 2022.
- [19] F. B. Fuchs, D. E. Worrall, V. Fischer, and M. Welling, “SE(3)-Transformers : 3D Roto-Translation Equivariant Attention Networks,” Nov. 2020.

- [20] S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt, and B. Kozinsky, “E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials,” *Nature Communications*, vol. 13, p. 2453, May 2022.
- [21] F. Sverrisson, J. Feydy, J. Southern, M. M. Bronstein, and B. E. Correia, “Physics-informed deep neural network for rigid-body protein docking,” in *MLDD 2022 - Machine Learning for Drug Discovery Workshop of ICLR 2022*, (Virtual conference, France), Apr. 2022.
- [22] O.-E. Ganea, X. Huang, C. Bunne, Y. Bian, R. Barzilay, T. Jaakkola, and A. Krause, “Independent SE(3)-Equivariant Models for End-to-End Rigid Protein Docking,” Mar. 2022.
- [23] W. Kabsch, “A solution for the best rotation to relate two sets of vectors,” *Acta Crystallographica Section A : Crystal Physics, Diffraction, Theoretical and General Crystallography*, vol. 32, pp. 922–923, Sept. 1976.
- [24] C. Ionescu, O. Vantzos, and C. Sminchisescu, “Training Deep Networks with Structured Layers by Matrix Backpropagation,” Apr. 2016.
- [25] T. Papadopoulos and M. I. A. Lourakis, “Estimating the Jacobian of the Singular Value Decomposition : Theory and Applications,” in *Computer Vision - ECCV 2000* (G. Goos, J. Hartmanis, and J. Van Leeuwen, eds.), vol. 1842, pp. 554–570, Berlin, Heidelberg : Springer Berlin Heidelberg, 2000.
- [26] M. Buttenschoen, G. M. Morris, and C. M. Deane, “PoseBusters : AI-based docking methods fail to generate physically valid poses or generalise to novel sequences,” Aug. 2023.
- [27] W. Lu, Q. Wu, J. Zhang, J. Rao, C. Li, and S. Zheng, “TANKBind : Trigonometry-Aware Neural Networks for Drug-Protein Binding Structure Prediction,” June 2022.