

# Independent $SE(3)$ -Equivariant Models for End-To-End Rigid Protein Docking

ICLR 2022

---

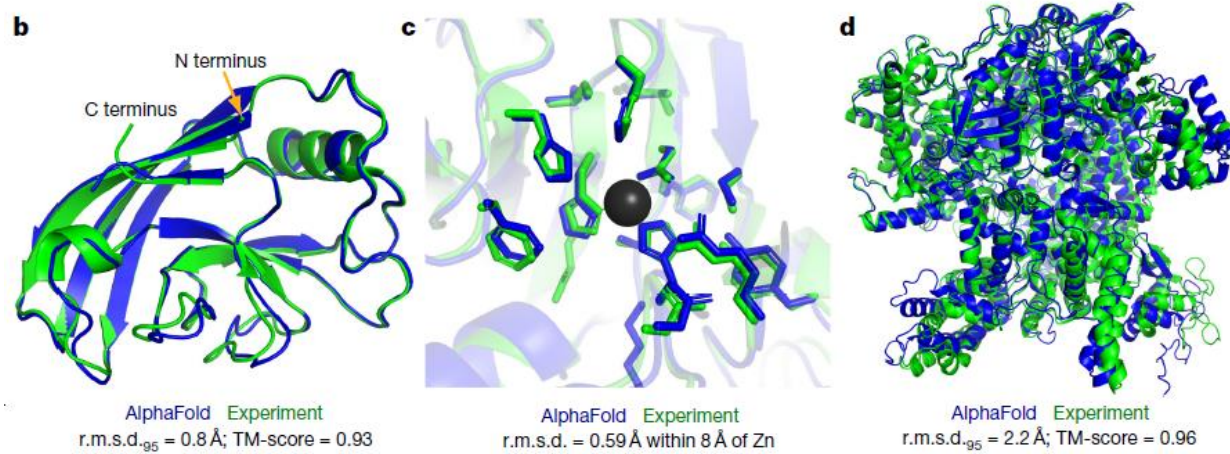
Department of Artificial Intelligence Korea University

Younghan Son

July 2022

# AlphaFold

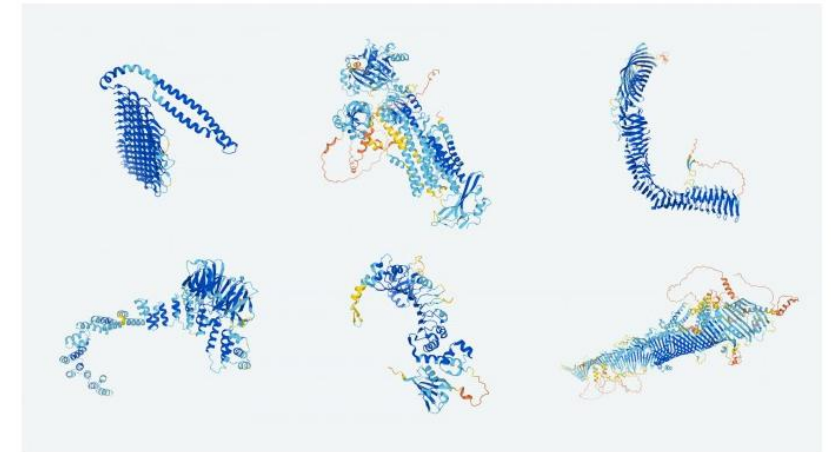
- Predicting 3D structure of protein
- Knowing structure is very expensive → Solved by Deep learning



10년 걸린 연구 3개월에 끝낸다...구글 AI '알파폴드'가 주도하는 생명과학 혁명

2022.04.22 07:00

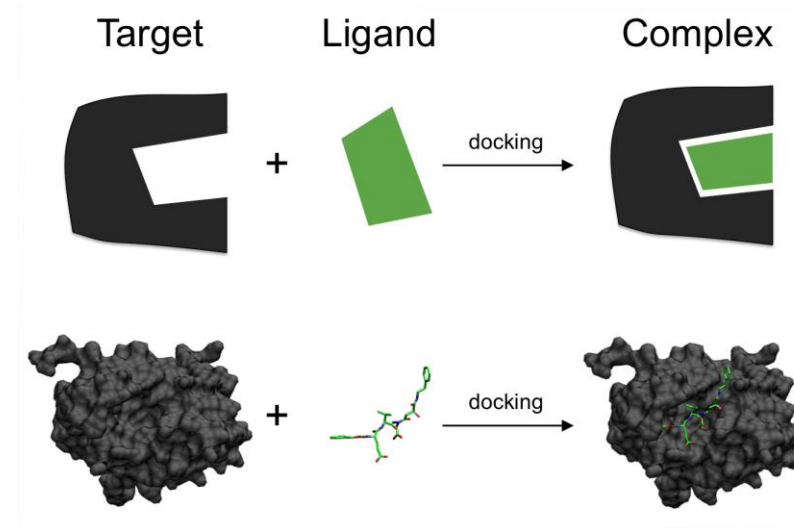
| 플라스틱 먹는 효소 연구, 진화생물학까지 확장 중



알파폴드가 해독한 여러 단백질의 3D 구조. 다양한 단백질 구조를 통해 생명 현상에 관여하는 단백질의 기능이 구현된다. 답마인드 제공.

# Protein docking

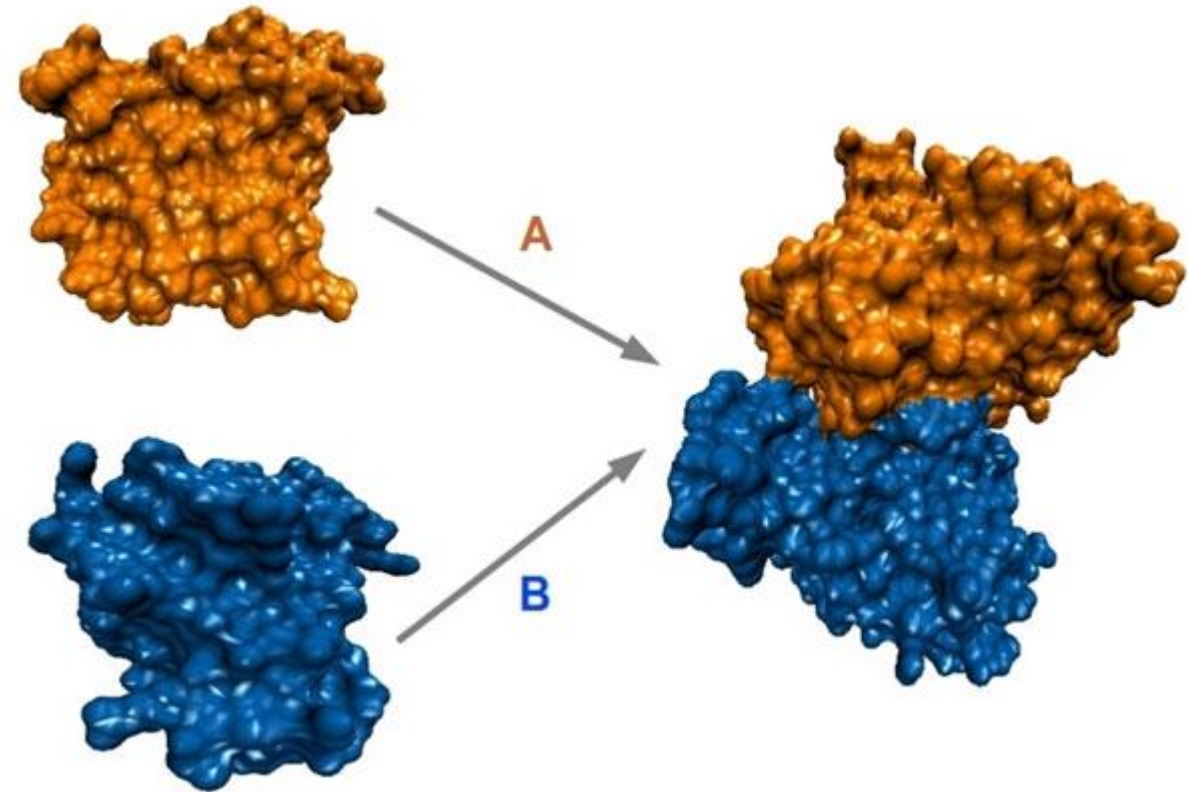
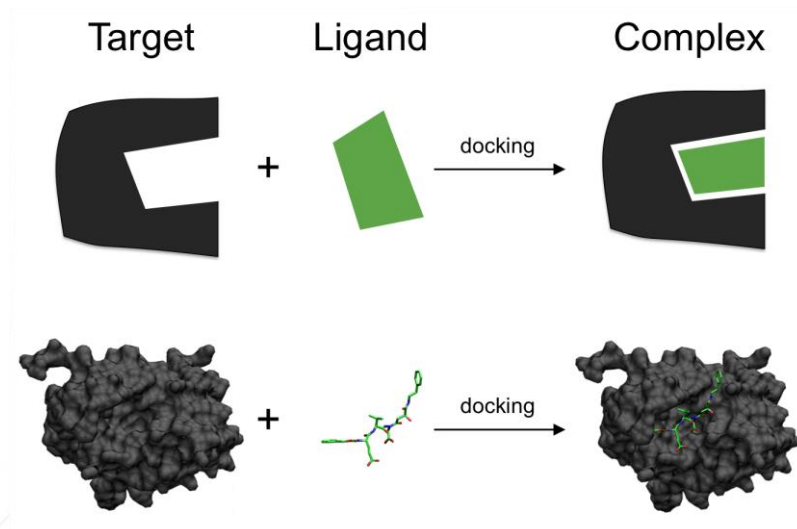
- Knowing not only single structure, predicts molecular docking





# Protein docking

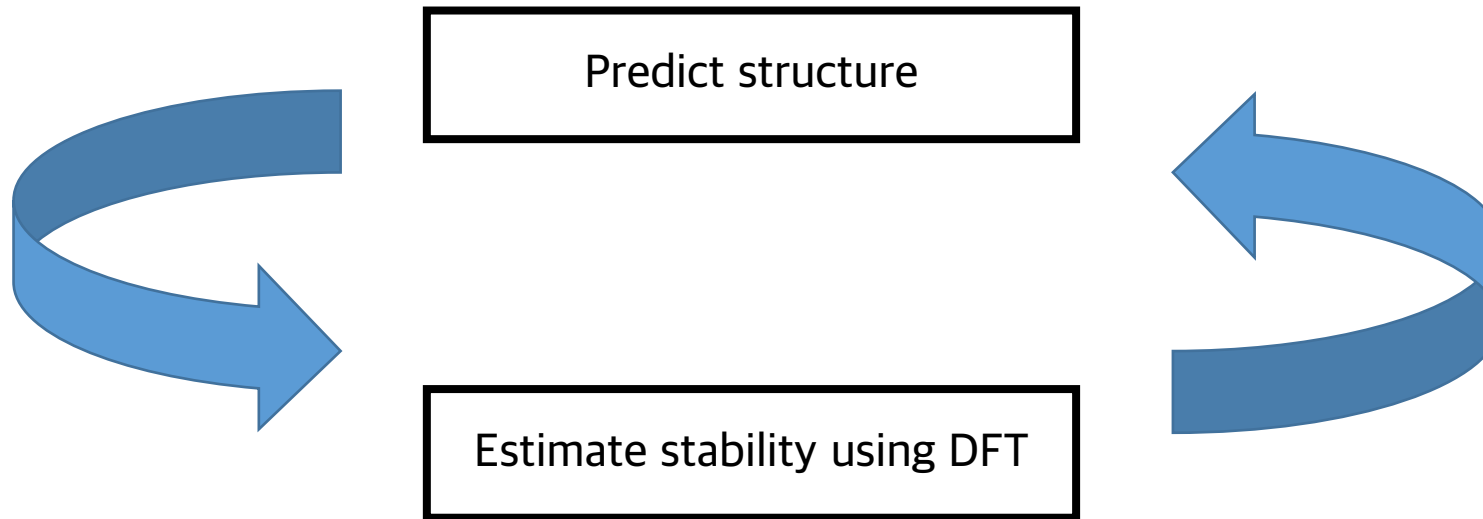
- Protein docking relates to protein synthesis, DNA replication...
- Key to understanding protein interactions' mechanism, effects and developing therapeutic interventions
- We are going to dock ligand to receptor



# Conventional methods

---

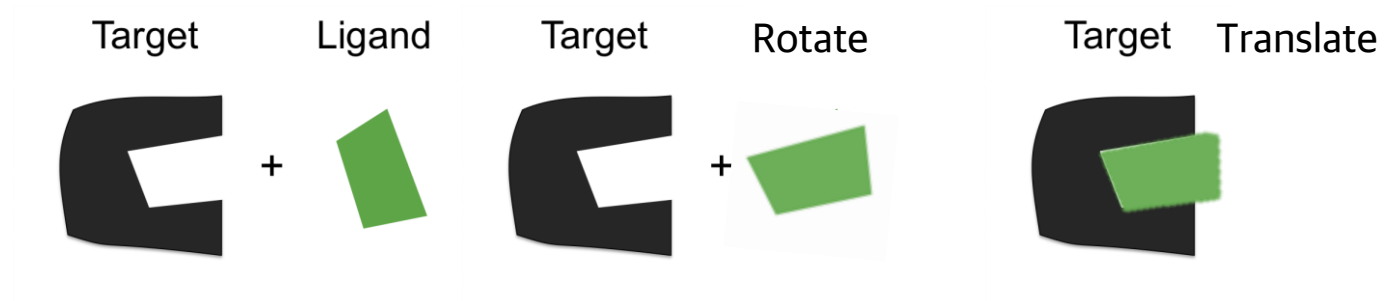
- Trial and Error
- DFT(Density Functional Theory) calculates quantum mechanical equations → takes long time



# Docking process

---

- 1) Rotate ligand to appropriate position
- 2) Translate(move) ligand to docked position



# Recap - Rotation matrix

- Rotation matrix: 3x3 matrix

$$\text{2D} \quad R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

$$\begin{aligned} \text{3D} \quad R &= R_z(\alpha) R_y(\beta) R_x(\gamma) = \begin{bmatrix} \cos \alpha & \overset{\text{yaw}}{-\sin \alpha} & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \beta & \overset{\text{pitch}}{0} & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix} \begin{bmatrix} 1 & \overset{\text{roll}}{0} & 0 \\ 0 & \cos \gamma & -\sin \gamma \\ 0 & \sin \gamma & \cos \gamma \end{bmatrix} \\ &= \begin{bmatrix} \cos \alpha \cos \beta & \cos \alpha \sin \beta \sin \gamma - \sin \alpha \cos \gamma & \cos \alpha \sin \beta \cos \gamma + \sin \alpha \sin \gamma \\ \sin \alpha \cos \beta & \sin \alpha \sin \beta \sin \gamma + \cos \alpha \cos \gamma & \sin \alpha \sin \beta \cos \gamma - \cos \alpha \sin \gamma \\ -\sin \beta & \cos \beta \sin \gamma & \cos \beta \cos \gamma \end{bmatrix} \end{aligned}$$

# SO(3)

---

- Rotation matrix는 한 개가 아니라 여러 개
- 여기서 특히 역행렬이 가능한 R의 집합을 general linear group(GL)  $GL(3, \mathbb{R})$
- GL 에서 determinant 가 +1/-1 인matrix들을 Orthogonal(O)  $O(3) \subset GL(3, \mathbb{R})$
- 이 중 변환 후 점 사이의 거리가 변하지 않는 Special Orthogonal(SO)  $(SO(3) \subset O(3))$
- 이렇게 SO(3)는 translation 을 고려하지 않는 Rotation 만을 고려하는 조건이다.

$$(SO(3) \subset O(3) \subset GL(3, \mathbb{R}))$$



# SE(3)

- SO(3)는 rotation 3X3 만을 고려한 조건
- Translation을 위해서는 4X4 matrix가 필요
- Rotation + Translation Equivariance 가 만족하면 SE(3) Equivariance라고 부름

## 2.3. 3D+YPR to matrix

### 2.3.1 Transformation

The transformation matrix associated to a 6D pose given in yaw-pitch-roll form has this structure:

$$\mathbf{P}(x, y, z, \phi, \chi, \psi) = \left( \begin{array}{ccc|c} \mathbf{R}(\phi, \chi, \psi) & x \\ & y \\ & z \\ \hline 0 & 0 & 0 & 1 \end{array} \right) \quad (2.14)$$

where the  $3 \times 3$  rotation matrix  $\mathbf{R}$  can be easily derived from the fact that each of the three individual rotations (yaw, pitch and roll) operate consecutively one after the other, i.e. over the already modified axis. This can be achieved by right-side multiplication of the individual rotation matrices:

# Invariant and Equivariance

## Definition

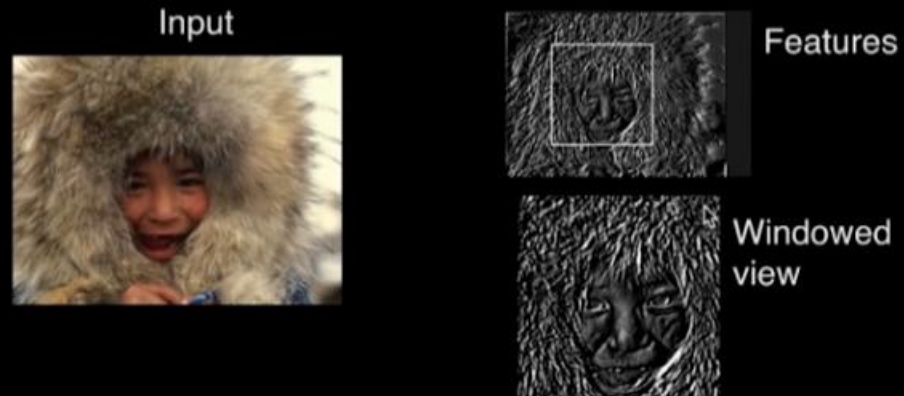
$f$  is **invariant** if  $f(h \cdot x) = f(x)$  for all  $h \in H$ .

$f$  is **equivariant** if  $f(h \cdot x) = h \cdot f(x)$  for all  $h \in H$ .

- ex) CNN is equivariant to translation

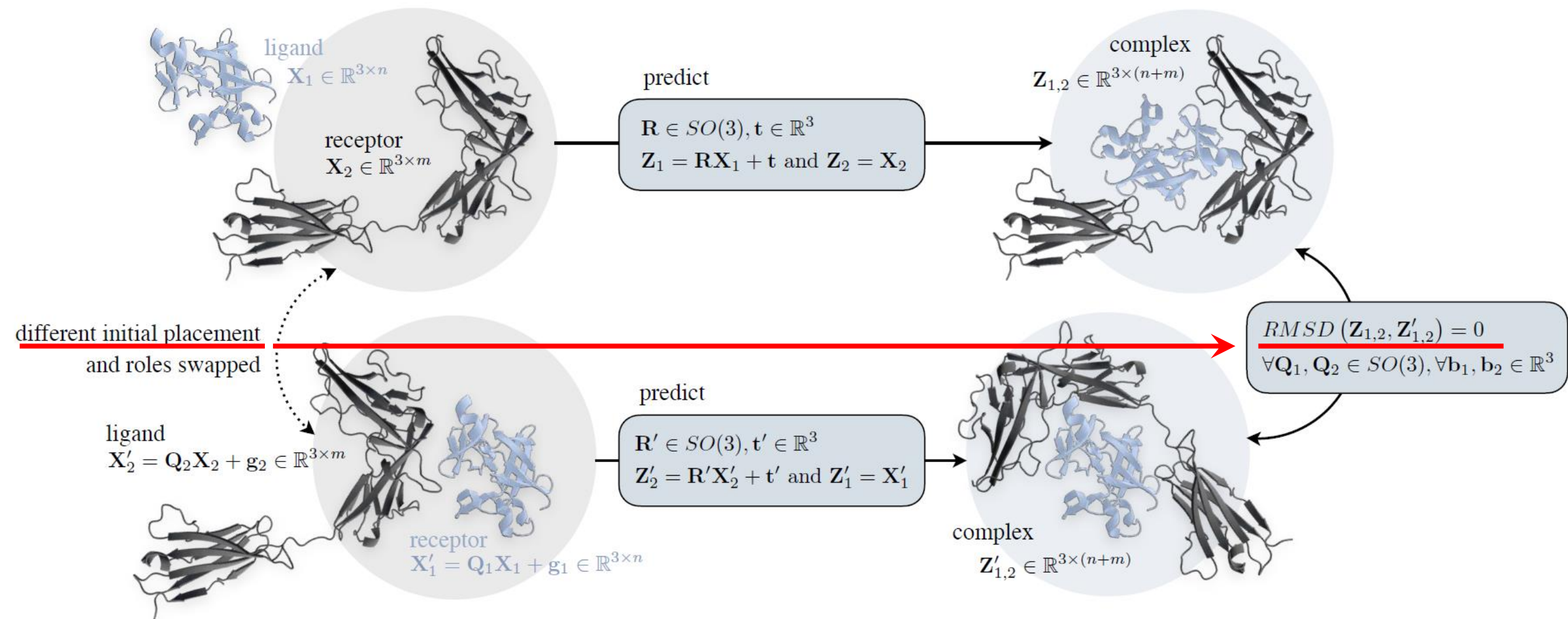
## Translation Equivariance in CNNs

### Existing CNNs: Translation Equivariance



# Invariant to Initial position

- Ligand X1 and Receptor X2's initial positions may differ, but docking results should be the same (assuming X2 is at the correct position)



## SE(3) Equivariance

$$(\mathbf{R}(\mathbf{Z}_1|\mathbf{Z}_2)\mathbf{Z}_1 + \mathbf{t}(\mathbf{Z}_1|\mathbf{Z}_2)) \oplus \mathbf{Z}_2 \equiv (\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{X}_1 + \mathbf{t}(\mathbf{X}_1|\mathbf{X}_2)) \oplus \mathbf{X}_2, \quad (\text{SE(3)-invariance})$$

$$(\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{X}_1 + \mathbf{t}(\mathbf{X}_1|\mathbf{X}_2)) \oplus \mathbf{X}_2 \equiv \mathbf{X}_1 \oplus (\mathbf{R}(\mathbf{X}_2|\mathbf{X}_1)\mathbf{X}_2 + \mathbf{t}(\mathbf{X}_2|\mathbf{X}_1)), \quad (\text{commutativity})$$

$$\forall \mathbf{Q}_1, \mathbf{Q}_2 \in SO(3), \forall \mathbf{g}_1, \mathbf{g}_2 \in \mathbb{R}^3, \forall \mathbf{X}_1 \in \mathbb{R}^{3 \times n}, \mathbf{X}_2 \in \mathbb{R}^{3 \times m}, \text{ and } \mathbf{Z}_l = \mathbf{Q}_l \mathbf{X}_l + \mathbf{g}_l, l \in \{1, 2\}.$$

- $\mathbf{Z}$ 는 임의의 initialization

$$\mathbf{Z}_l = \mathbf{Q}_l \mathbf{X}_l + \mathbf{g}_l, l \in \{1, 2\}$$

- 1) 초기 위치가 달라도 (rotate, translate) 결과가 같아야 함 (SE(3) invariance)
- 2)  $\mathbf{X}_1, \mathbf{X}_2$  위치가 반대라도 결과가 같아야 함 (Commutativity)  $\rightarrow$  생각보다 쉽지 않다고 함



## SE(3) 증명

*Proof.* Denote the predicted ligand position by  $\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{X}_1 + \mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) = \tilde{\mathbf{X}}_1$ .

Assume first that SE(3)-invariance of the predicted docked complex defined by Eq. (1) is satisfied. Then the transformation to dock  $\mathbf{Q}_1\mathbf{X}_1 + \mathbf{g}_1$  with respect to  $\mathbf{Q}_2\mathbf{X}_2 + \mathbf{g}_2$  is the same as the transformation to change  $\mathbf{Q}_1\mathbf{X}_1 + \mathbf{g}_1$  into  $\mathbf{Q}_2\tilde{\mathbf{X}}_1 + \mathbf{g}_2$ . We use the notation:  $\mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2) = (\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2))^\top$ . Then, we have the following derivation steps:

$$\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{X}_1 + \mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) = \tilde{\mathbf{X}}_1$$

$$\mathbf{X}_1 + \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) = \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\tilde{\mathbf{X}}_1$$

$$\mathbf{X}_1 + \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) = \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_2^\top(\mathbf{Q}_2\tilde{\mathbf{X}}_1 + \mathbf{g}_2 - \mathbf{g}_2)$$

$$\mathbf{X}_1 + \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) = \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_2^\top(\mathbf{Q}_2\tilde{\mathbf{X}}_1 + \mathbf{g}_2) - \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_2^\top\mathbf{g}_2$$

$$\mathbf{X}_1 + \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) + \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_2^\top\mathbf{g}_2 = \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_2^\top(\mathbf{Q}_2\tilde{\mathbf{X}}_1 + \mathbf{g}_2)$$

$$\mathbf{Q}_1^\top(\mathbf{Q}_1\mathbf{X}_1 + \mathbf{g}_1 - \mathbf{g}_1) + \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)(\mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) + \mathbf{Q}_2^\top\mathbf{g}_2) = \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_2^\top(\mathbf{Q}_2\tilde{\mathbf{X}}_1 + \mathbf{g}_2)$$

$$\mathbf{Q}_1^\top(\mathbf{Q}_1\mathbf{X}_1 + \mathbf{g}_1) - \mathbf{Q}_1^\top\mathbf{g}_1 + \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)(\mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) + \mathbf{Q}_2^\top\mathbf{g}_2) = \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_2^\top(\mathbf{Q}_2\tilde{\mathbf{X}}_1 + \mathbf{g}_2)$$

$$\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_1^\top(\mathbf{Q}_1\mathbf{X}_1 + \mathbf{g}_1) - \mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_1^\top\mathbf{g}_1 + \mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) + \mathbf{Q}_2^\top\mathbf{g}_2 = \mathbf{Q}_2^\top(\mathbf{Q}_2\tilde{\mathbf{X}}_1 + \mathbf{g}_2)$$

$$\mathbf{Q}_2\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_1^\top(\mathbf{Q}_1\mathbf{X}_1 + \mathbf{g}_1) - \mathbf{Q}_2\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_1^\top\mathbf{g}_1 + \mathbf{Q}_2\mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) + \mathbf{g}_2 = \mathbf{Q}_2\tilde{\mathbf{X}}_1 + \mathbf{g}_2$$

$$\mathbf{R}(\mathbf{Q}_1\mathbf{X}_1 + \mathbf{g}_1|\mathbf{Q}_2\mathbf{X}_2 + \mathbf{g}_2) = \mathbf{Q}_2\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_1^\top$$

$$\mathbf{t}(\mathbf{Q}_1\mathbf{X}_1 + \mathbf{g}_1|\mathbf{Q}_2\mathbf{X}_2 + \mathbf{g}_2) = \mathbf{Q}_2\mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) - \mathbf{Q}_2\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{Q}_1^\top\mathbf{g}_1 + \mathbf{g}_2$$



## 교환법칙 증명

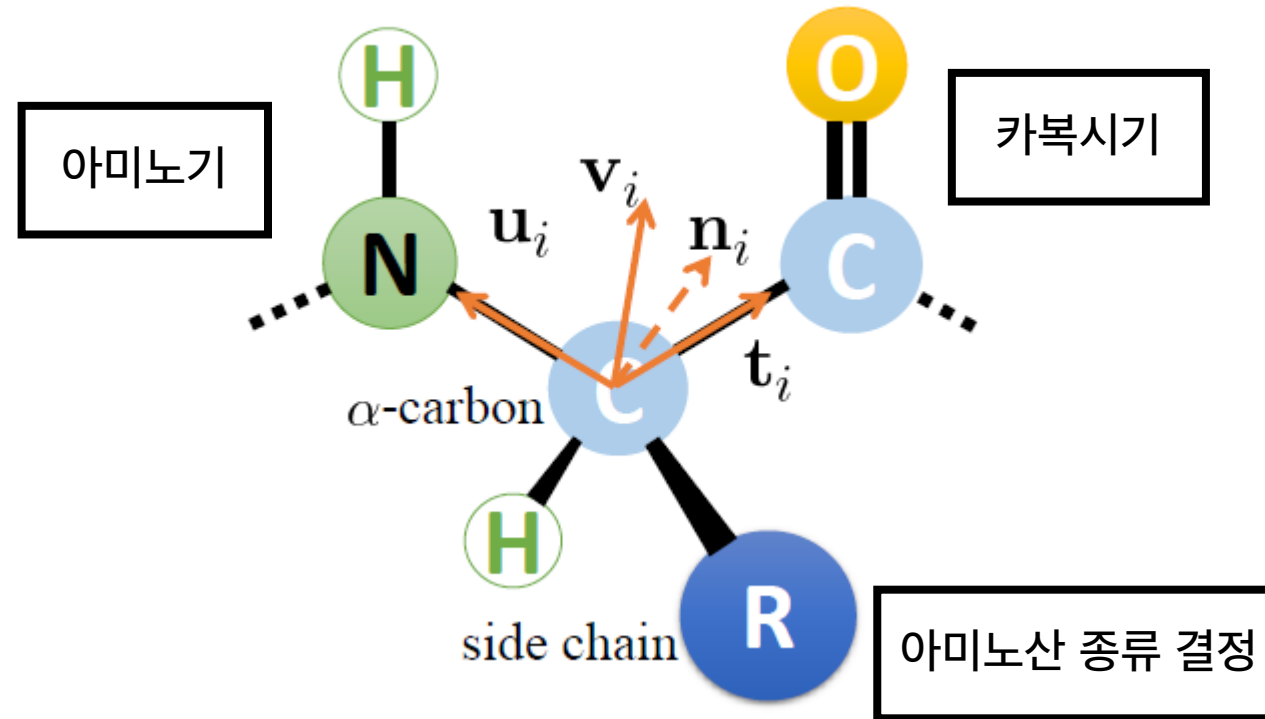
*Proof.* We use the notation  $\mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2) := (\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2))^\top$ . As in Appendix B.1, we denote  $\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{X}_1 + \mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) = \tilde{\mathbf{X}}_1$ . Then the transformation to dock  $\mathbf{X}_2$  with respect to  $\mathbf{X}_1$  is the same as the transformation to change  $\tilde{\mathbf{X}}_1$  back to  $\mathbf{X}_1$ , which is

$$\begin{aligned}\mathbf{R}(\mathbf{X}_1|\mathbf{X}_2)\mathbf{X}_1 + \mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) &= \tilde{\mathbf{X}}_1 \\ \mathbf{X}_1 + \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{t}(\mathbf{X}_1|\mathbf{X}_2) &= \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\tilde{\mathbf{X}}_1 \\ \mathbf{X}_1 &= \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\tilde{\mathbf{X}}_1 - \mathbf{R}^\top(\mathbf{X}_1|\mathbf{X}_2)\mathbf{t}(\mathbf{X}_1|\mathbf{X}_2)\end{aligned}$$

From the last equation above, we derive the transformation to change  $\tilde{\mathbf{X}}_1$  back to  $\mathbf{X}_1$ , which is the same as the transformation to dock  $\mathbf{X}_2$  with respect to  $\mathbf{X}_1$ .  $\square$

# 그래프 표현

- 노드피쳐: Alpha carbon (x,y,z) coordinate + 아미노산 종류 + Surface aware node feature(이후 설명)
- 엣지피쳐: Relative position + Orientation + RBF distance
- 엣지는 가까운 knn 10개



# IEGMN - Independent E(3)-Equivariant Graph Matching Networks

- X: coordinate F: node feature, Z: transformed coordinate, H: node feature embedding
- H는 equivariant 일 필요 없음
- 아래 노드피쳐 msg와 어텐션, coordinate 모두 SE(3) Invariant를 만족함(증명 생략) → IEGMN은 SE(3) 조건을 만족

노드 피쳐 msg  $\underline{\mathbf{m}_{j \rightarrow i}} = \varphi^e(\mathbf{h}_i^{(l)}, \mathbf{h}_j^{(l)}, \exp(-\|\mathbf{x}_i^{(l)} - \mathbf{x}_j^{(l)}\|^2 / \sigma), \mathbf{f}_{j \rightarrow i}), \forall e_{j \rightarrow i} \in \mathcal{E}_1 \cup \mathcal{E}_2$

어텐션 msg  $\underline{\mu_{j \rightarrow i}} = a_{j \rightarrow i} \mathbf{W} \mathbf{h}_j^{(l)}, \forall i \in \mathcal{V}_1, j \in \mathcal{V}_2 \text{ or } i \in \mathcal{V}_2, j \in \mathcal{V}_1$

$$a_{j \rightarrow i} = \frac{\exp(\langle \psi^q(\mathbf{h}_i^{(l)}), \psi^k(\mathbf{h}_j^{(l)}) \rangle)}{\sum_{j'} \exp(\langle \psi^q(\mathbf{h}_i^{(l)}), \psi^k(\mathbf{h}_{j'}^{(l)}) \rangle)}$$

노드 피쳐 aggregate  $\mathbf{m}_i = \frac{1}{|\mathcal{N}(i)|} \sum_{j \in \mathcal{N}(i)} \mathbf{m}_{j \rightarrow i}, \forall i \in \mathcal{V}_1 \cup \mathcal{V}_2$

어텐션 aggregate  $\mu_i = \sum_{j \in \mathcal{V}_2} \mu_{j \rightarrow i}, \forall i \in \mathcal{V}_1, \text{ and } \mu_i = \sum_{j \in \mathcal{V}_1} \mu_{j \rightarrow i}, \forall i \in \mathcal{V}_2$

coordinate update  $\mathbf{x}_i^{(l+1)} = \underline{\eta \mathbf{x}_i^{(0)}} + (1 - \eta) \mathbf{x}_i^{(l)} + \sum_{j \in \mathcal{N}(i)} \underline{(\mathbf{x}_i^{(l)} - \mathbf{x}_j^{(l)}) \varphi^x(\mathbf{m}_{j \rightarrow i})}, \forall i \in \mathcal{V}_1 \cup \mathcal{V}_2$

Edge weight 를 거리에 따라       $\mathbf{x}_1, \mathbf{x}_2$ 가 붙었을 때 사용됨

노드 피쳐 update  $\mathbf{h}_i^{(l+1)} = (1 - \beta) \cdot \mathbf{h}_i^{(l)} + \beta \cdot \varphi^h(\mathbf{h}_i^{(l)}, \mathbf{m}_i, \mu_i, \mathbf{f}_i), \forall i \in \mathcal{V}_1 \cup \mathcal{V}_2,$

# IEGMN 이 SE(3)를 만족함을 증명

## B.3 PROOF OF PROPOSITION 4.

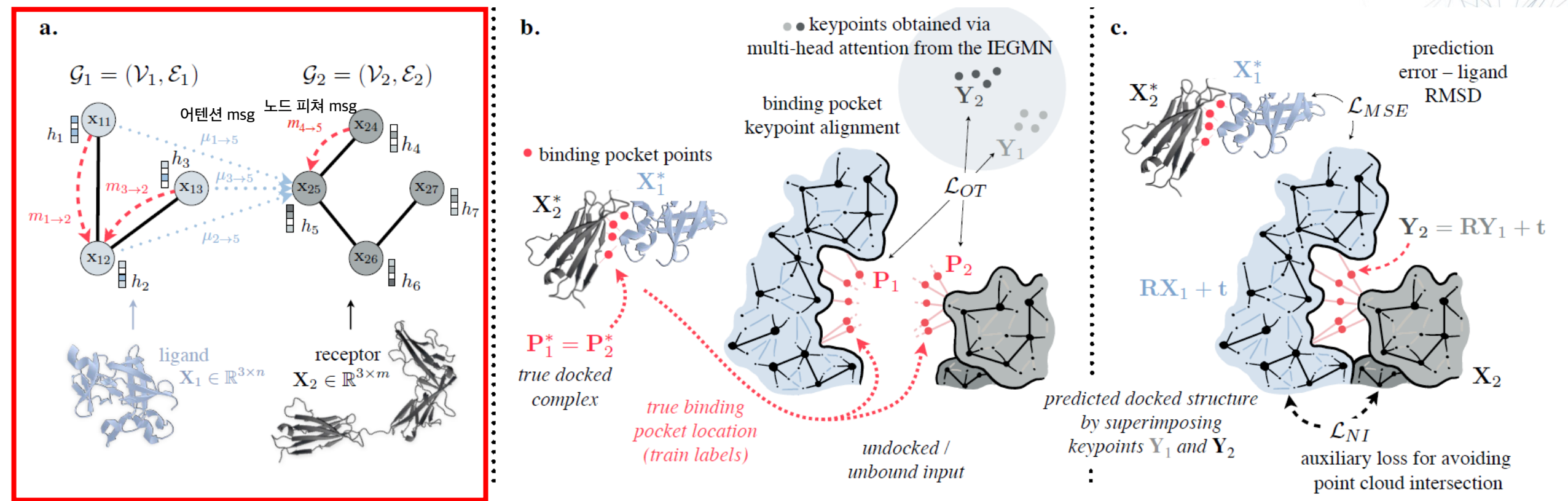
*Proof.* Let  $\mathbf{X}_1^{(l+1)}, \mathbf{H}_1^{(l+1)}, \mathbf{X}_2^{(l+1)}, \mathbf{H}_2^{(l+1)} = \text{IEGMN}(\mathbf{X}_1^{(l)}, \mathbf{H}_1^{(l)}, \mathbf{X}_2^{(l)}, \mathbf{H}_2^{(l)})$  be the output of an IEGMN layer. Then, for any matrices  $\mathbf{Q}_1, \mathbf{Q}_2 \in SO(3)$  and any translation vectors  $\mathbf{g}_1, \mathbf{g}_2 \in \mathbb{R}^3$ , we want to prove that IEGMN satisfy the pairwise independent SE(3)-equivariance property:

$$\mathbf{Q}_1 \mathbf{X}_1^{(l+1)} + \mathbf{g}_1, \mathbf{H}_1^{(l+1)}, \mathbf{Q}_2 \mathbf{X}_2^{(l+1)} + \mathbf{g}_2, \mathbf{H}_2^{(l+1)} = \text{IEGMN}(\mathbf{Q}_1 \mathbf{X}_1^{(l)} + \mathbf{g}_1, \mathbf{H}_1^{(l)}, \mathbf{Q}_2 \mathbf{X}_2^{(l)} + \mathbf{g}_2, \mathbf{H}_2^{(l)})$$

where each column of  $\mathbf{X}_1^{(l)} \in \mathbb{R}^{3 \times n}, \mathbf{H}_1^{(l)} \in \mathbb{R}^{d \times n}, \mathbf{X}_2^{(l)} \in \mathbb{R}^{3 \times m}$  and  $\mathbf{H}_2^{(l)} \in \mathbb{R}^{d \times m}$  represent an individual node's coordinate embedding or feature embedding.

We first note that the equations of our proposed IEGMN layer that compute messages  $\mathbf{m}_{j \rightarrow i}, \mu_{j \rightarrow i}, \mathbf{m}_i$  and  $\mu_i$  are SE(3)-invariant. Indeed, they depend on the initial features which are SE(3)-invariant by design, the current latent node embeddings  $\{\mathbf{h}_i^{(l)}\}_{i \in \mathcal{V}_1 \cup \mathcal{V}_2}$ , as well as the Euclidean distances on the current node coordinates  $\{\mathbf{x}_i^{(l)}\}_{i \in \mathcal{V}_1 \cup \mathcal{V}_2}$ . Thus, we also derive that the equation that computes the new latent node embeddings  $\mathbf{h}_i^{(l+1)}$  is SE(3)-invariant. Last, the equation that updates the coordinates  $\mathbf{x}_i^{(l+1)}$  is SE(3)-equivariant with respect to the 3D coordinates of nodes from the same graph as  $i$ , but SE(3)-invariant with respect to the 3D coordinates of nodes from the other graph since it only uses invariant transformations of the latter.





**Figure 3: Details on EQUIDOCK's Architecture and Losses.** **a.** The message passing operations in IEGMN guarantee pairwise independent SE(3)-equivariance as in Eq. (4), **b.** We predict keypoints for each protein that are aligned with the binding pocket location using an additional optimal transport (OT) loss, **c.** After predicting the docked position, we compute an MSE loss on the ligand, as well as a loss to discourage body intersections.



# Keypoints by attention score

- IEGMN이후 Binding point가 될 point를 뽑는 과정
  - Attention 이용 K head 가 있을 때 Y는 K개 뽑힘
- Y1, Y2 두 단백질의 binding point가 겹쳐야 함(Y1 = Y2)

$$y_{1k} := \sum_{i=1}^n \alpha_i^k \mathbf{z}_{1i}; \quad y_{2k} := \sum_{j=1}^m \beta_j^k \mathbf{z}_{2j},$$

$$\alpha_i^k = \text{softmax}_i \left( \frac{1}{\sqrt{d}} \mathbf{h}_{1i}^\top \mathbf{W}'_k \mu(\varphi(\mathbf{H}_2)) \right)$$

head

# Optimal Transport

- 예측된 binding site Y1, Y2 (3XK)
- 실제 binding site (label) P1, P2 (P1==P2) (3XS)
- 예측은 K개, 실제 label은 S개(S is variable across dataset)
- Optimal transport를 통해 둘 간의 loss를 최소화함

$$\mathcal{L}_{OT} = \min_{\mathbf{T} \in \mathcal{U}(S, K)} \langle \mathbf{T}, \mathbf{C} \rangle, \quad \text{where } \mathbf{C}_{s,k} = \|\mathbf{y}_{1k} - \mathbf{p}_{1s}\|^2 + \|\mathbf{y}_{2k} - \mathbf{p}_{2s}\|^2,$$

Optimal Transport plan

# Kabsch 알고리즘

- Kabsch 알고리즘 은 RMSD (root mean squared )을 최소화하는 최적의 회전 행렬 을 계산하는 방법입니다. 편차) 두 쌍의 포인트 세트 사이. 그래픽, 화학 포식 학 분자 구조를 비교하고 생물 정보학 단백질 구조를 비교하는 데 유용합니다  
by Wikipedia site:ko.tr2tr.wiki
- Kabsch 알고리즘을 사용하여  $Y_2 = Y_1$  를 완성합니다.

$\mathbf{A} = \overline{\mathbf{Y}}_2 \overline{\mathbf{Y}}_1^\top \in \mathbb{R}^{3 \times 3}$  computed using zero-mean keypoints. The singular value decomposition (SVD) is  $\mathbf{A} = \mathbf{U}_2 \mathbf{S} \mathbf{U}_1^\top$ , where  $\mathbf{U}_2, \mathbf{U}_1 \in O(3)$ . Finally, we define the differentiable functions

$$\mathbf{R}(\mathbf{X}_1 | \mathbf{X}_2; \theta) = \mathbf{U}_2 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & d \end{pmatrix} \mathbf{U}_1^\top, \quad \text{where } d = \text{sign}(\det(\mathbf{U}_2 \mathbf{U}_1^\top)) \quad (13)$$

$$\mathbf{t}(\mathbf{X}_1 | \mathbf{X}_2; \theta) = \mu(\mathbf{Y}_2) - \mathbf{R}(\mathbf{X}_1 | \mathbf{X}_2; \theta) \mu(\mathbf{Y}_1),$$

# Avoiding Point cloud Intersection

- 실제 protein에서 intersection이 일어 날리 없음.
  - Point cloud가 안겹치도록 해야함
- Surface of point cloud 구해야함

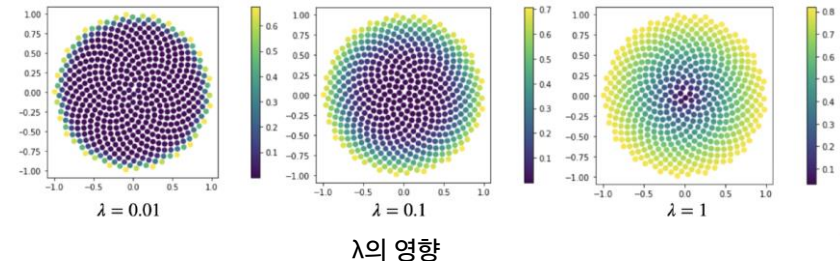
$$G(\mathbf{x}) = -\sigma \ln\left(\sum_{i=1}^n \exp(-\|\mathbf{x} - \mathbf{x}_i\|^2 / \sigma)\right)$$

- $G(x)$ 가  $\gamma$  보다 크면 외부, 작으면 내부

$$\mathcal{L}_{\text{NI}} = \frac{1}{n} \sum_{i=1}^n \max(0, \gamma - \underbrace{G_2(\mathbf{x}_{1i})}_{\text{X1이 X2 기준 밖이어야함}}) + \frac{1}{m} \sum_{j=1}^m \max(0, \gamma - \underbrace{G_1(\mathbf{x}_{2j})}_{\text{X2이 X1 기준 밖이어야함}}).$$

# Surface node feature

- 노드피쳐 중 하나
- 노드에다가 surface 정보를 넣어 줌
- 만약 내부에 있는 점이라면, 주변 노드들의 벡터의 합이 0에 가까워야함



$$\rho_i(\mathbf{x}_i; \lambda) = \frac{\left\| \sum_{i' \in \mathcal{N}_i} w_{i,i',\lambda} (\mathbf{x}_i - \mathbf{x}_{i'}) \right\|}{\sum_{i' \in \mathcal{N}_i} w_{i,i',\lambda} \|\mathbf{x}_i - \mathbf{x}_{i'}\|}, \quad \text{where } w_{i,i',\lambda} = \frac{\exp(-\|\mathbf{x}_i - \mathbf{x}_{i'}\|^2 / \lambda)}{\sum_{j \in \mathcal{N}_i} \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / \lambda)}. \quad (16)$$

P가 크면 표면

Normalized weighted force mean vector of length  $\rho_i(\mathbf{x}_i; \lambda)$

Inter residue force vector  $w_{i,i',\lambda}(\mathbf{x}_i - \mathbf{x}_{i'})$

Protein surface

a.

b.



# Results

- RMSD(Root Means Square Deviation), Geometric task에 많이 쓰임

**Table 1: Complex Prediction Results.** As in the main text, the proprietary baselines might internally use parts of the test sets (e.g. to extract templates or features), thus their numbers might be optimistic.

Methods	DIPS Test Set						DB5.5 Test Set					
	Complex RMSD			Interface RMSD			Complex RMSD			Interface RMSD		
	Median	Mean	Std	Median	Mean	Std	Median	Mean	Std	Median	Mean	Std
ATTRACT	17.17	14.93	10.39	12.41	14.02	11.81	9.55	10.09	9.88	7.48	10.69	10.90
HDock	6.23	10.77	11.39	3.90	8.88	10.95	0.30	5.34	12.04	0.24	4.76	10.83
CLUSPRO	15.76	14.47	10.24	12.54	13.62	11.11	3.38	8.25	7.92	2.31	8.71	9.89
PATCHDOCK	15.24	13.58	10.30	11.44	12.15	10.50	18.26	18.00	10.12	18.88	18.75	10.06
EQUIDock	13.29	14.52	7.13	10.18	11.92	7.01	14.13	14.72	5.31	11.97	13.23	4.93

# Results

- Inference time이 굉장히 낮음

Table 4: Inference time comparison (in seconds). Note: ClusPro and PatchDock were run manually using the respective public webserver, thus their runtimes are influenced by their cluster load.

Methods	Runtime on DIPS Test Set					Runtime on DB5.5 Test Set				
	Mean	Median	Min	Max	Std	Mean	Median	Min	Max	Std
ATTRACT (LOCAL)	1285	793	62	8192	793	570	524	180	1708	373
HDOCK (LOCAL)	778	635	145	3177	570	615	461	210	2593	459
CLUSPRO (WEB)	10475	9831	2632	22654	4512	15507	14393	9207	28528	4126
PATCHDOCK (WEB)	7378	6900	600	16560	3979	3290	2820	1080	14520	2459
<b>EQUIDOCK (LOCAL)</b>	<b>5</b>	<b>3</b>	<b>1</b>	<b>22</b>	<b>5</b>	<b>5</b>	<b>3</b>	<b>1</b>	<b>53</b>	<b>10</b>



---

**Thank you**

---