GENRATIVE MODELING FOR PROTEIN STRUCTURES

Authors: Namrata Anand, Po-Ssu Huang (Stanford - 2018)

Presenter: Ngoc Khoi Dang

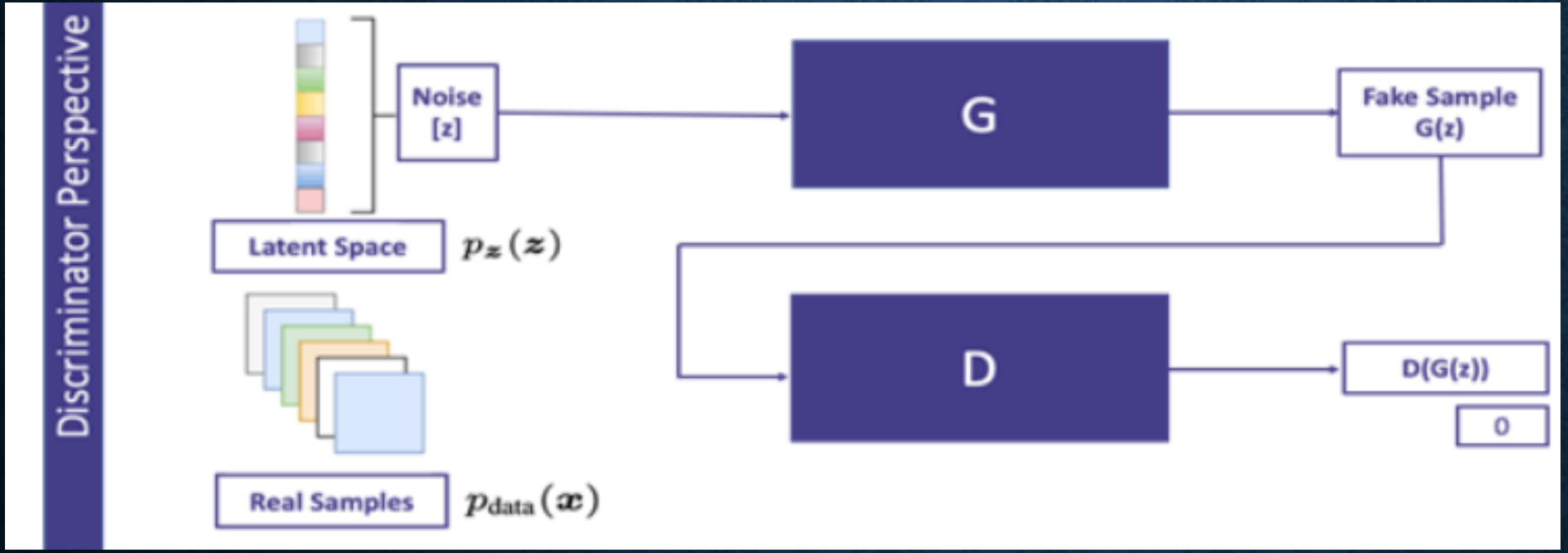
CURRENT PROBLEMS

- Analyzing the structure and function of proteins is a key part of understanding biology at molecular and cellular level.
- To fully understand the structure and function of protein, it is ideal to create *de novo* proteins.
- Current protein design process relies heavily on heuristics and imperfect scoring functions which requires expertise knowledge.
- Authors propose a new model using Generative Adversarial networks + Alternating Direction Method of Multipliers to learn the protein design and folding process.

REVIEW – PROTEIN STRUCTURE

- Interactions between the side-chains, the protein backbone and the environment defines the 3D protein structure
- Sequence-agnostic structure generation

REVIEW – GAN MODEL

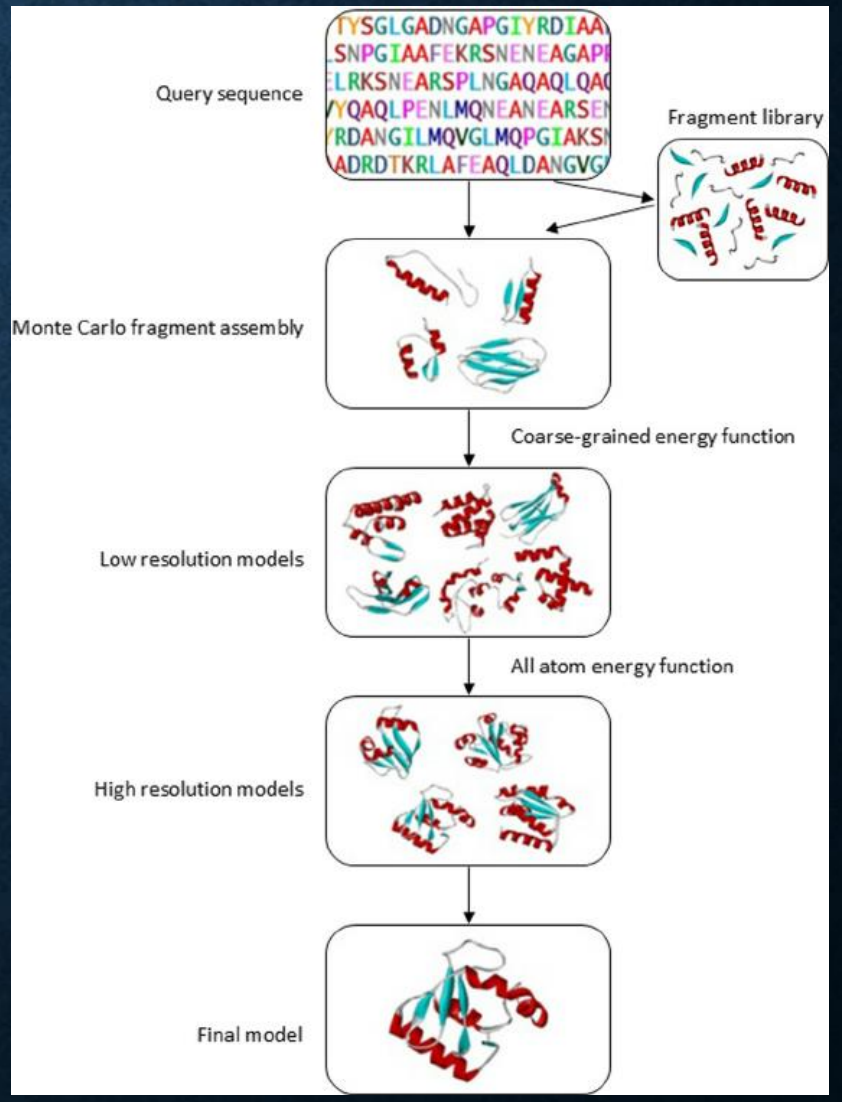


$$\max_D \; \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}[\log D(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(1-(D(G(\mathbf{z}))))]$$

$$\max_G \; \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}[\log(D(G(z)))]$$

REVIEW – ROSETTA PROCEDURE

- State-of-the-art for designing protein structures in 2018
- Heuristic energy function
- Pros: sampling is guided by a highly refined energy function, model building process are intuitive and flexible, guarantee correct local structure
- Cons: Very slow



REVIEW – ADMM

- Determining 3D cartesian coordinates given pairwise distance is a convex problem. This optimization problem can be solved quickly using SDP solvers if sample size is small.

$$[a_1, a_2, \ldots a_m] = A \in \mathbb{R}^{n \times m}$$

$$G = A^T A \in \mathcal{S}_+^m$$

$$D, \text{ with } d_{ij} = \|a_i - a_j\|_2$$

$$\min_{G,\eta} \lambda \|\eta\|_1 + \frac{1}{2} \sum_{i=1,j=1}^{m} (g_{ii} + g_{jj} - 2g_{ij} + \eta_{ij} - d_{ij}^2)^2$$

$$\text{subject to} \quad G \in \mathcal{S}_+^n \qquad \text{slack term } \eta$$

REVIEW – ADMM

- Alternating direction method of multipliers (ADMM) is an algorithm that solves convex optimization problems by breaking them into smaller pieces, each of which are then easier to handle.
- ADMM is a combination of dual ascent with decomposition and the method of multipliers.

$$\min_{G,Z,\eta} \quad \lambda \|\eta\|_1 + \frac{1}{2}\left(\sum_{i=1,j=1}^{m} (g_{ii} + g_{jj} - 2g_{ij} + \eta_{ij} - d_{ij}^2)^2\right) + \mathbb{1}\{Z \in \mathcal{S}_+^m\}$$

$$\text{subject to } G - Z = 0$$

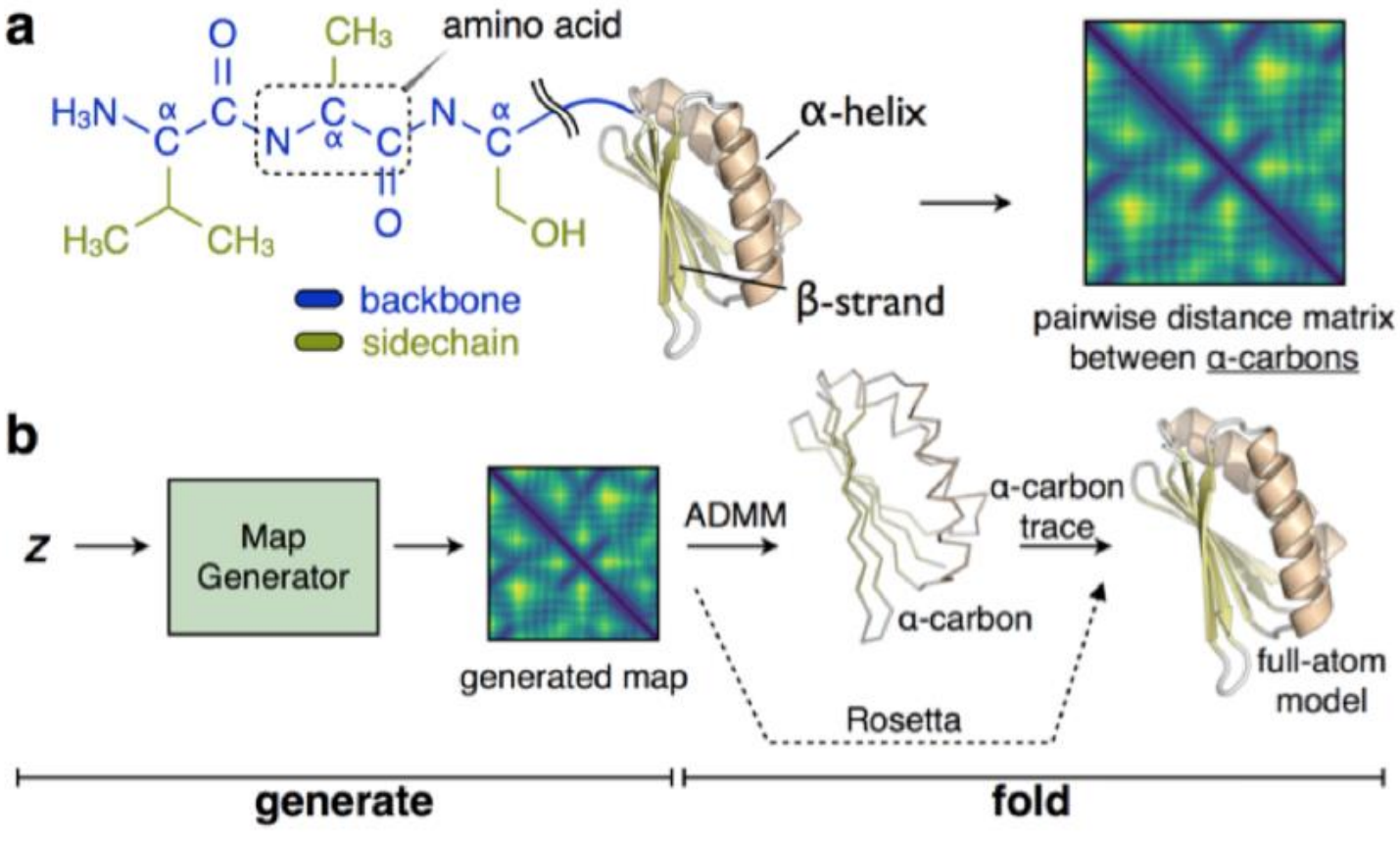
$$G_{k+1}, \eta_{k+1} = \underset{G,\eta}{\text{argmin}} \; [\, \lambda \|\eta\|_1 + \frac{1}{2}\sum_{i=1,j=1}^{m} (g_{ii} + g_{jj} - 2g_{ij} + \eta_{ij} - d_{ij}^2)^2 + \frac{\rho}{2}\|G - Z_k + U_k\|_2^2 \,]$$

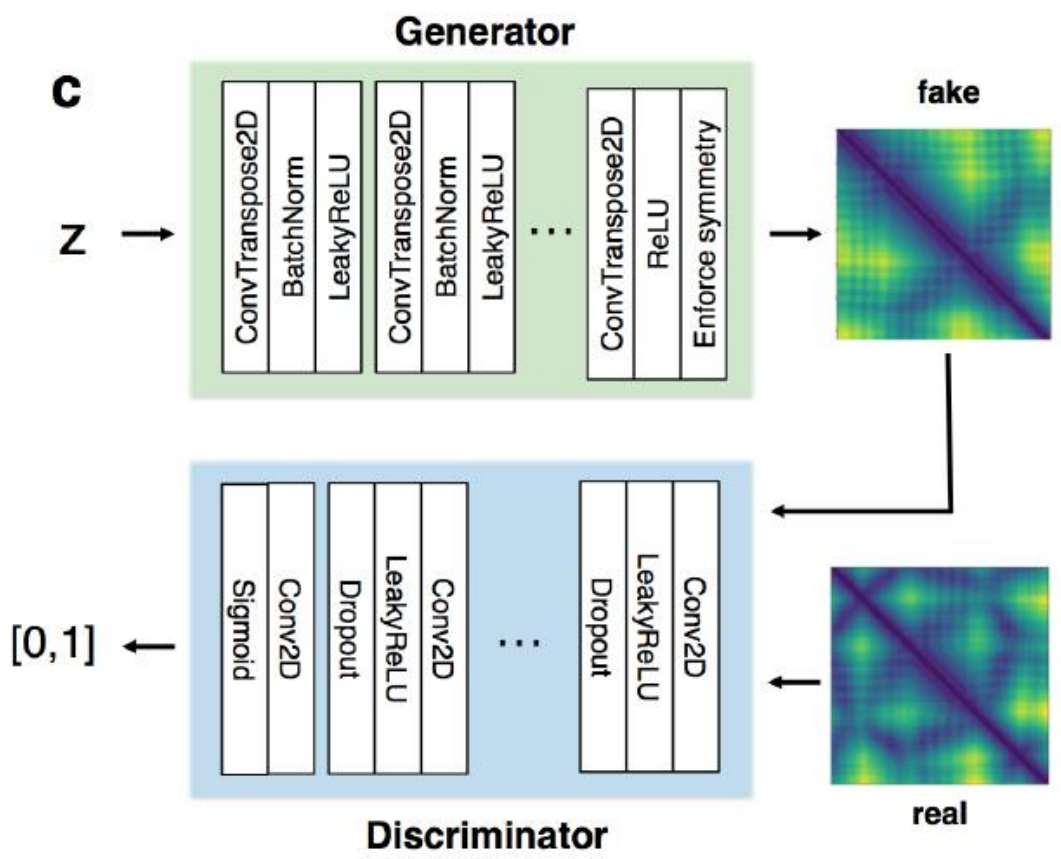
$$Z_{k+1} = \Pi_{\mathcal{S}_+^n}(G_{k+1} + U_k)$$

$$U_{k+1} = U_k + G_{k+1} - Z_{k+1}$$

Not always correct for local errors in secondary structures

PIPELINE


a
amino acid
α-helix
β-strand
backbone
sidechain
pairwise distance matrix
between α-carbons
b
z
Map
Generator
generated map
ADMM
α-carbon
trace
α-carbon
full-atom
model
Rosetta
generate
fold


c
Generator
z
ConvTranspose2D
BatchNorm
LeakyReLU
ConvTranspose2D
BatchNorm
LeakyReLU
ConvTranspose2D
ReLU
Enforce symmetry
fake
real
[0,1]
Conv2D
Sigmoid
Conv2D
Dropout
LeakyReLU
Conv2D
Dropout
LeakyReLU
Conv2D
Discriminator

METHOD – DATASET & MAP GENERATION

- Dataset: Protein Data Bank
- Generating maps:
  - Encode 3D structure in 2D pairwise distances between α-carbon on the protein backbone
  - 16-, 64-, 128- residue maps are generated
- Folding:
  - Rosetta's procedure: generate a-C distance constraints by fragment sampling
  - ADMM algorithm: find 3D a-C placement that satisfies the generated constraints
  - Then the a-C trace script is used to trace an idealized peptide backbone geometry

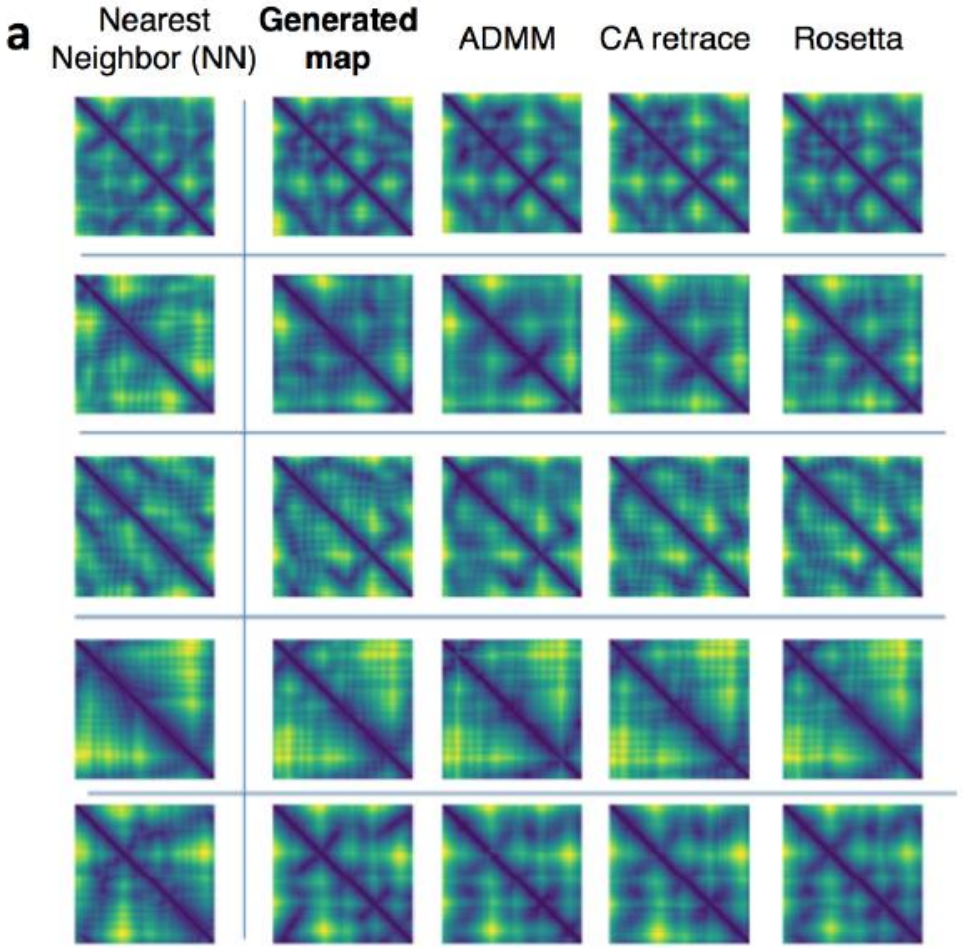
EXPERIMENTS

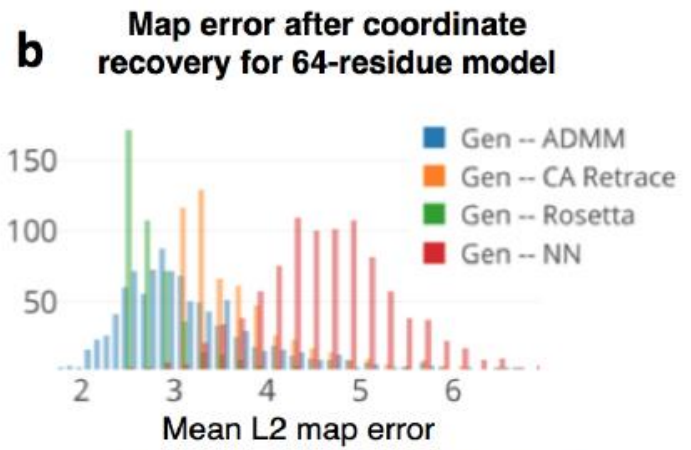
- The results are compared to
  - TorusDBN (HMM)
  - FB5-HMM (HMM)
  - Multi-scale torsion angle GAN
  - 3DGAN
  - Full-atom GAN

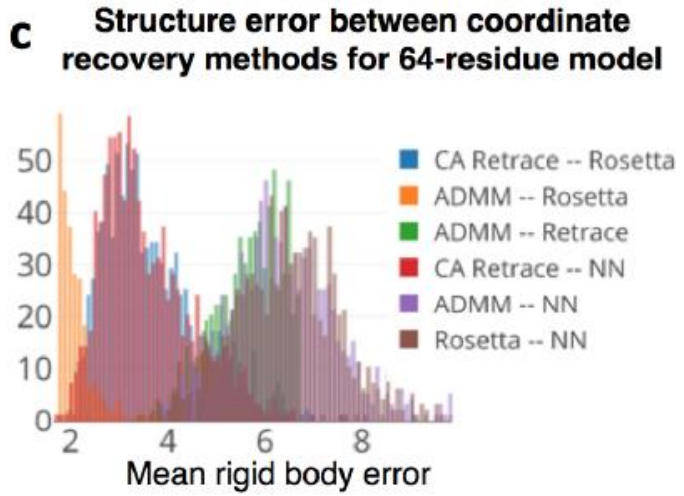
EXPERIMENTS

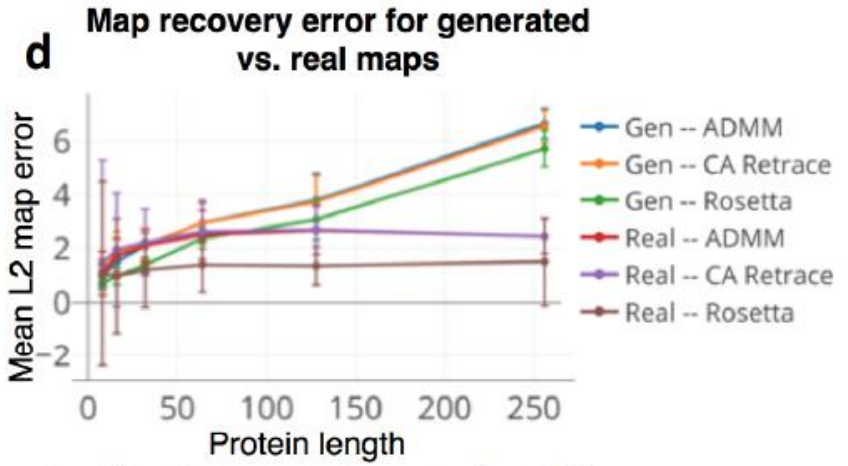
- **Inpainting for protein design:** testing how to use the trained generative models to infer contextually correct missing portions of protein structures
  - 10-residue supervised autoencoder: AE is trained to reconstruct completed 64-residue pairwise distance maps given input maps with random 10-residue corruptions
  - Random corruption supervised autoencoder: same AE is trained to reconstruct completed 64-residue pairwise distance maps given input maps with random 10-residue corruptions residues in length.
  - Rosetta remodel: uses fragment sampling to do loop closure, followed by a sequence design process, guided by a heuristic energy score

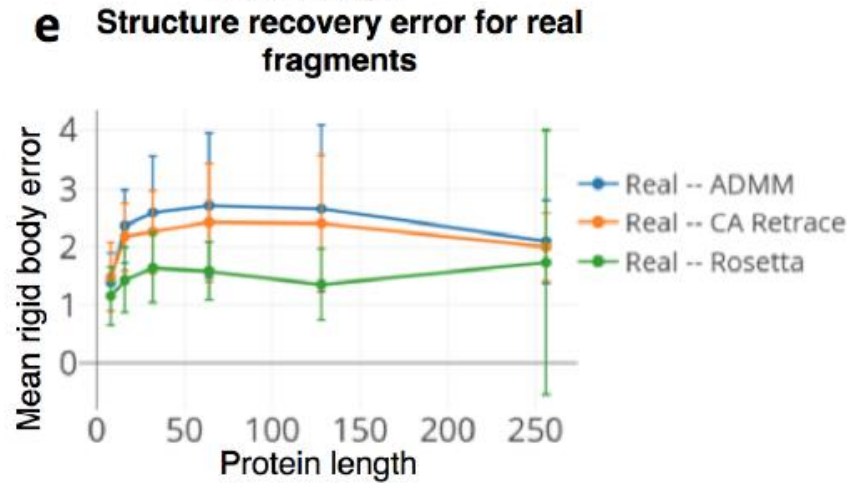
RESULTS


a
Nearest Neighbor (NN)
Generated map
ADMM
CA retrace
Rosetta

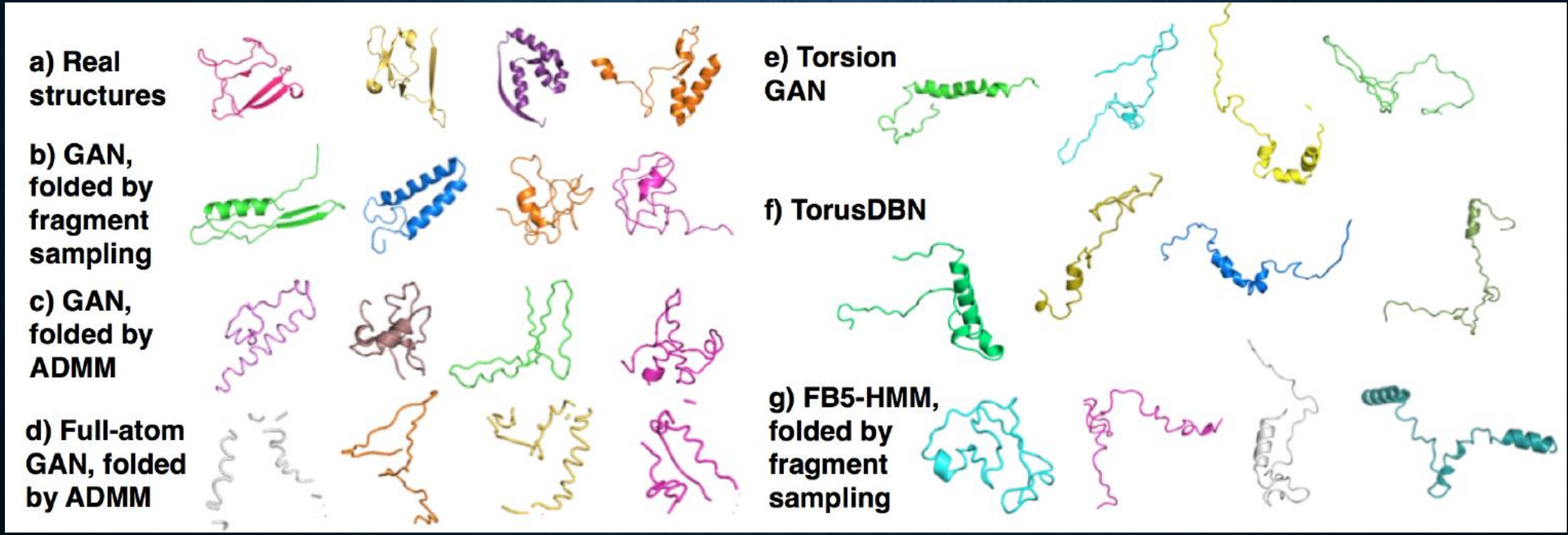

b
Map error after coordinate recovery for 64-residue model
Gen -- ADMM
Gen -- CA Retrace
Gen -- Rosetta
Gen -- NN
150
100
50
2
3
4
5
6
Mean L2 map error


c
Structure error between coordinate recovery methods for 64-residue model
CA Retrace -- Rosetta
ADMM -- Rosetta
ADMM -- Retrace
CA Retrace -- NN
ADMM -- NN
Rosetta -- NN
50
40
30
20
10
0
2
4
6
8
Mean rigid body error

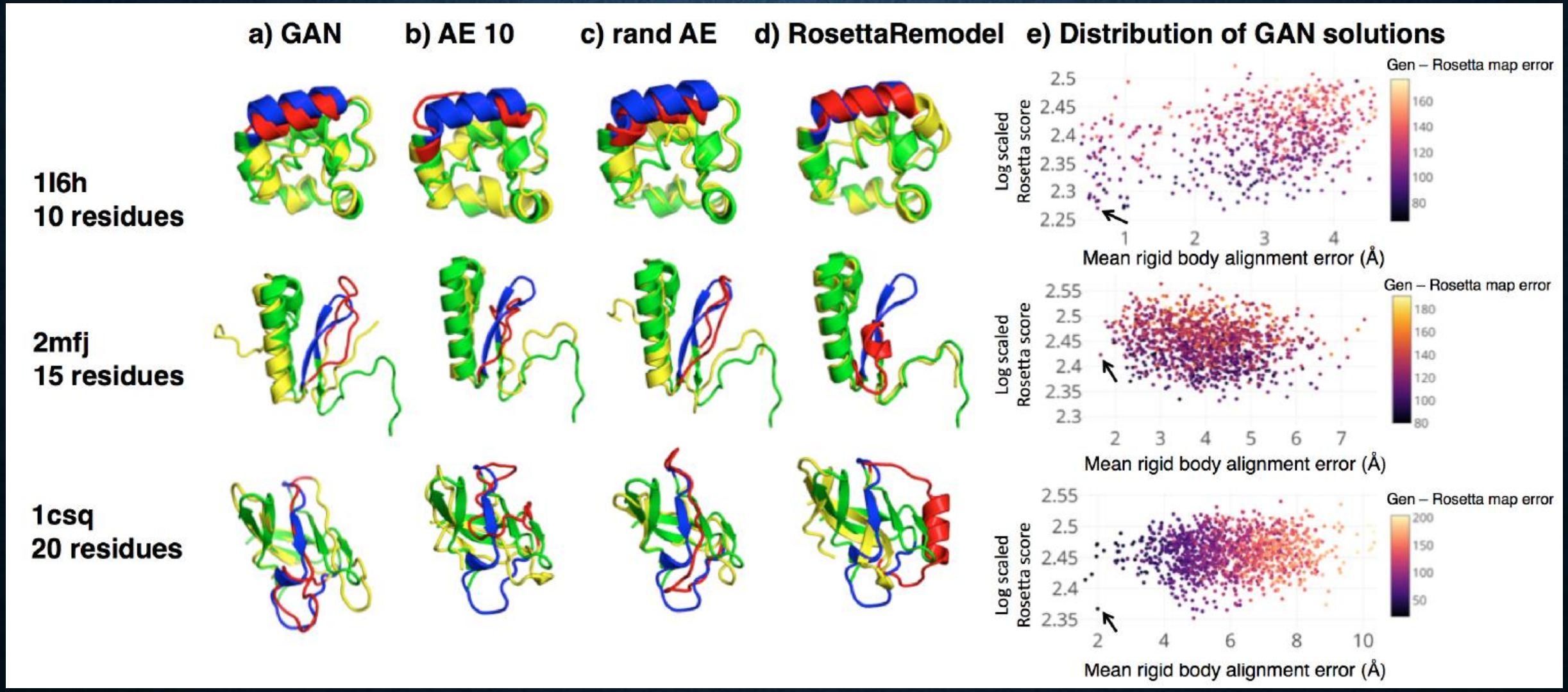

d
Map recovery error for generated vs. real maps
Gen -- ADMM
Gen -- CA Retrace
Gen -- Rosetta
Real -- ADMM
Real -- CA Retrace
Real -- Rosetta
Mean L2 map error
6
4
2
0
−2
0
50
100
150
200
250
Protein length


e
Structure recovery error for real fragments
Real -- ADMM
Real -- CA Retrace
Real -- Rosetta
Mean rigid body error
4
3
2
1
0
0
50
100
150
200
250
Protein length

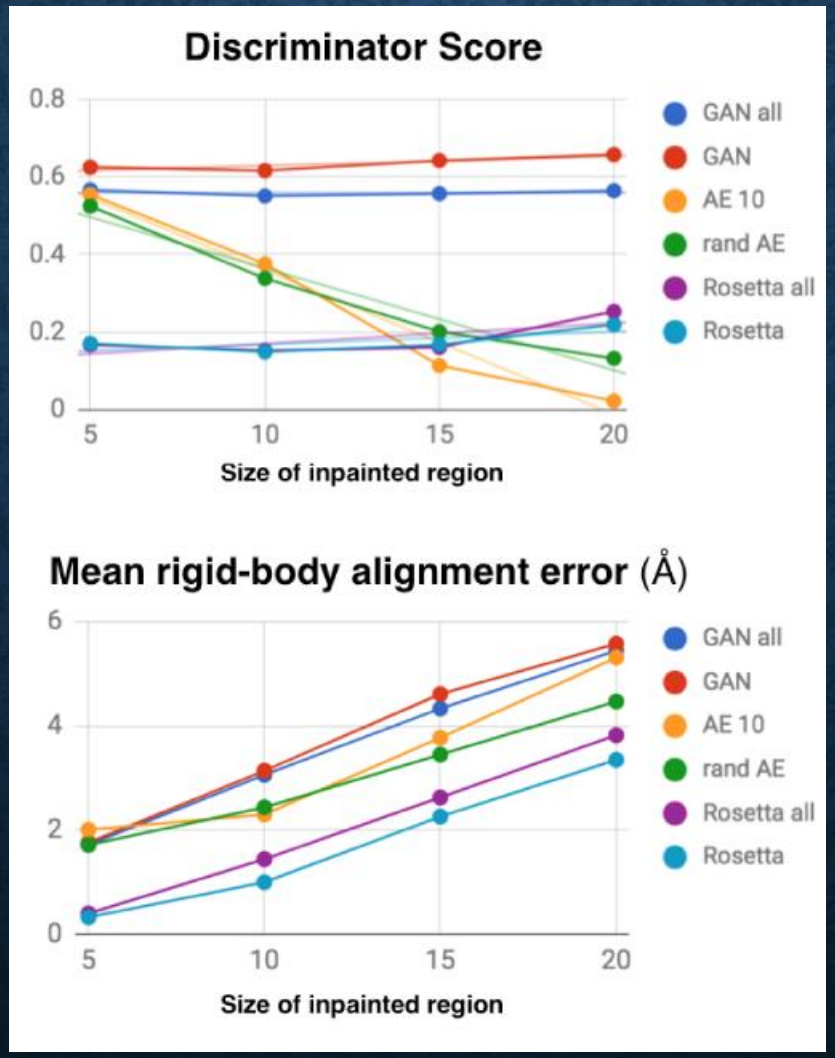
RESULTS


a) Real structures
b) GAN, folded by fragment sampling
c) GAN, folded by ADMM
d) Full-atom GAN, folded by ADMM
e) Torsion GAN
f) TorusDBN
g) FB5-HMM, folded by fragment sampling

RESULTS


a) GAN
b) AE 10
c) rand AE
d) RosettaRemodel
e) Distribution of GAN solutions
1l6h
10 residues
2mfj
15 residues
1csq
20 residues
Log scaled Rosetta score
Mean rigid body alignment error (Å)
Gen – Rosetta map error

RESULTS


Discriminator Score
0.8
0.6
0.4
0.2
0
5
10
15
20
Size of inpainted region
GAN all
GAN
AE 10
rand AE
Rosetta all
Rosetta
Mean rigid-body alignment error (Å)
6
4
2
0
5
10
15
20
Size of inpainted region
GAN all
GAN
AE 10
rand AE
Rosetta all
Rosetta