Computer-aided Protein Design

by combining automated reasoning and learning

Thomas Schiex

July 6-9 2021

JOBIM 2021

Why should we want to design proteins?

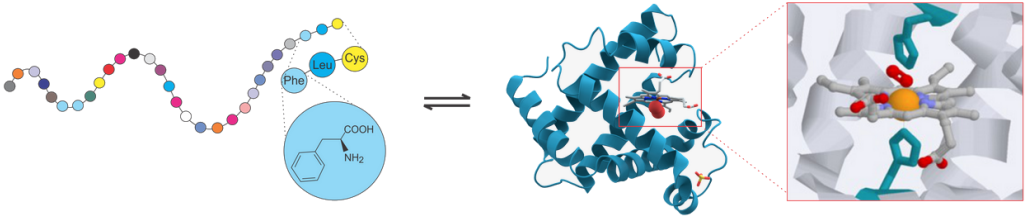
Eco-friendly chemical/structural nano-agents

- New drugs for health (human, animals, plants)
- New catalysts (environment, recycling, biofuels, food and feed, cosmetics…),
- New components for nanotechnologies
- Relying on inexpensive atomic level 3D-printers (bacterias, yeast, …)

Why should we want to design proteins?

Eco-friendly chemical/structural nano-agents

- New drugs for health (human, animals, plants)
- New catalysts (environment, recycling, biofuels, food and feed, cosmetics…),
- New components for nanotechnologies
- Relying on inexpensive atomic level 3D-printers (bacterias, yeast, …)

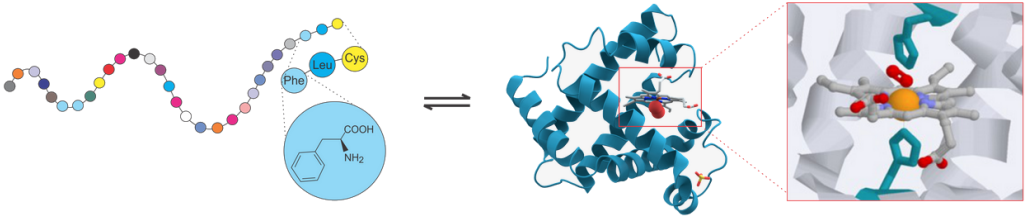


[0]Thanks to the Zhang lab. for this image.

Why should we want to design proteins?

Eco-friendly chemical/structural nano-agents

- New drugs for health (human, animals, plants)
- New catalysts (environment, recycling, biofuels, food and feed, cosmetics…),
- New components for nanotechnologies
- Relying on inexpensive atomic level 3D-printers (bacterias, yeast, …)



20^n sequences! Experimental techniques can only explore a very tiny fraction of it.

[0]Thanks to the Zhang lab. for this image.

The Computational Design Problem

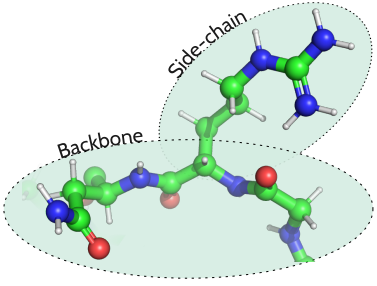
Informal definition (globular proteins)

Produce a sequence s of amino-acids that *spontaneously adopts* a conformation X that *performs some function*.

What defines a conformation ?

Conformation

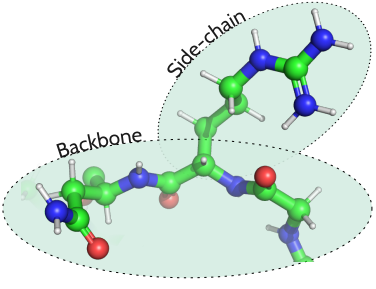
- backbone: dihedral angles ϕ_i, ψ_i
- sequence: amino-acid choice s_i
- side-chains: torsion angles χ_{ij}

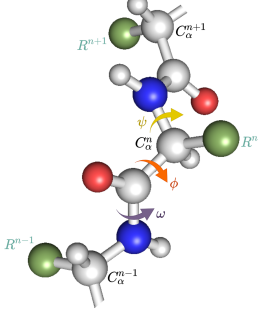


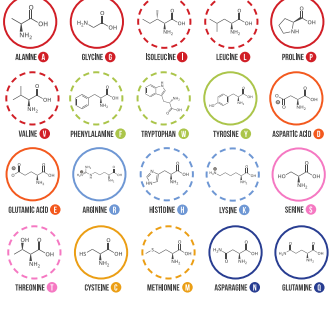
What defines a conformation ?

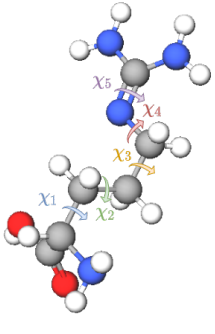
Conformation

- backbone: dihedral angles ϕ_i, ψ_i
- sequence: amino-acid choice s_i
- side-chains: torsion angles χ_{ij}





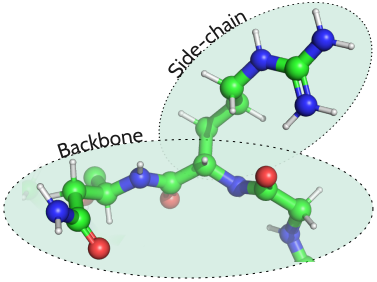




What defines a conformation ?

Conformation

- backbone: dihedral angles ϕ_i, ψ_i
- sequence: amino-acid choice s_i
- side-chains: torsion angles χ_{ij}



Challenging space to explore

- very high dimensionality, continuous variables ($\phi_i, \psi_i, \chi_{ij}$)
- discrete set of possible sequences s (size 20^n)

Atomic forces and entropic effects

- Chemical bonds geometries
- Inter atomic forces (electrostatics, polar, van der Waals…)
- Solvent effects

Atomic forces and entropic effects

- Chemical bonds geometries
- Inter atomic forces (electrostatics, polar, van der Waals…)
- Solvent effects

Thermodynamics[2]

- The stability of a sequence s in a given conformation X can be estimated through a real valued energy function $E(\mathrm{s}, X)$.

$$p_{\mathrm{s}}(X) \propto e^{-\frac{E(\mathrm{s},X)}{k_B T}}$$

- intractable non convex $E(\mathrm{s}, X)$ (free energy, quantum mechanics)
- Plus **extra requirements** for the function itself (sequence, geometry, flexibility…).

The "rigid backbone, discrete rotamers, pairwise decomposable energy" problem

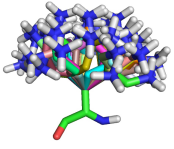
1. (ϕ_i, ψ_i) are given (rigid backbone).

The "rigid backbone, discrete rotamers, pairwise decomposable energy" problem

1. (ϕ_i, ψ_i) are given (rigid backbone).
2. sequence s is discrete, so χ_{ij} is discretized too.

Rotamer libraries: Tuffery,[28] Penultimate,[15] Dunbrack[25] …

Catalog of (amino acid, side-chain conformations) pairs build from the PDB (typically 400 or more rotamers)

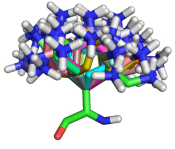


The "rigid backbone, discrete rotamers, pairwise decomposable energy" problem

1. (ϕ_i, ψ_i) are given (rigid backbone).
2. sequence $\mathbf{s}$ is discrete, so χ_{ij} is discretized too.
3. a pairwise decomposable energy function $E(\mathbf{s}, X)$

Rotamer libraries: Tuffery,[28] Penultimate,[15] Dunbrack[25] …

Catalog of (amino acid, side-chain conformations) pairs build from the PDB (typically 400 or more rotamers)



We need to minimize (+ fitness) Precomputed tables

$$E(\mathbf{s}, X) = E_{\varnothing} + \sum_{i=1}^{n} E_i(i_r) + \sum_{(i,j)\in I} E_{ij}(i_r, j_s)$$

It's an ill-posed problem

Forgetting all approximations

Even if $(\mathbf{s}, \chi)$ minimizes E on (ϕ, ψ), a better backbone configuration for $\mathbf{s}$ may exist.

It's an ill-posed problem

Forgetting all approximations

Even if $(\mathbf{s}, \chi)$ minimizes E on (ϕ, ψ), a better backbone configuration for $\mathbf{s}$ may exist.

Extra checks

1. Post-hoc continuous minimization of ϕ, ψ, χ (nicely dealt with by OSPREY[7,10])
2. Molecular dynamics simulations (expensive).
3. Forward folding: predict the structure from $\mathbf{s}$.

It's an ill-posed problem

Forgetting all approximations

Even if $(\mathbf{s}, \chi)$ minimizes E on (ϕ, ψ), a better backbone configuration for $\mathbf{s}$ may exist.

Extra checks

1. Post-hoc continuous minimization of ϕ, ψ, χ (nicely dealt with by OSPREY[7,10])
2. Molecular dynamics simulations (expensive).
3. Forward folding: predict the structure from $\mathbf{s}$.

It works[22]

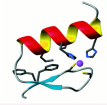
- There are less than 2,000 known folds for many more sequences.
- Secondary structure elements and hydrophobic packing constrain the space.
- We are in control and can make designs very predictable (forward folding).

1985

Calmodulin-binding peptide

[DeGrado et al. 1985]

1997



Zinc Finger

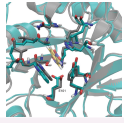
[Dehiyat & Mayo 1997]

2003

Novel Topology (top7)

[Kuhlman et al. 2003]

2008



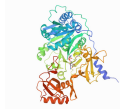
Functional Enzyme

[Rothlisberger et al. 2008]

Enzyme for Multi-Step Reaction

[Jiang et al. 2008]

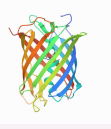
2009



Enzyme activity

[Chen et al. 2009]

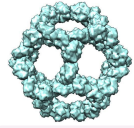
2011



Longer Emission Wave Length Fluorescence

[Chica et al. 2011]

2016



Self-Assembling Nanocage

[Hsia et al. 2016]

2019

Auto-Assembling Symmetrical Protein

[Niguchi et al. 2019]

NP-hard[19] (intractable?) — **Precomputed tables**

$$E(\mathbf{s}, X) = E_{\varnothing} + \sum_{i=1}^{n} E_i(i_r) + \sum_{(i,j)\in I} E_{ij}(i_r, j_s)$$

- mostly solved by Monte-Carlo algorithms (Rosetta)[14]
- NP-hard: standard excuse for approximate modeling or (meta)-heuristics.

Minimizing E

NP-hard[19] (intractable?) — **Precomputed tables**

$$E(\mathbf{s}, X) = E_{\varnothing} + \sum_{i=1}^{n} E_i(i_r) + \sum_{(i,j)\in I} E_{ij}(i_r, j_s)$$

- mostly solved by Monte-Carlo algorithms (Rosetta)[14]
- NP-hard: standard excuse for approximate modeling or (meta)-heuristics.

Using Cost Function Network (CFN) algorithms — **github.com/toulbar2**

- Intense progress in AI on logical/Boolean reasoning (200TB theorem proof[13])
- CFN use automated reasoning algorithms extended to numerical functions.[11]
- Can still handle logical information (constraints)

Cost Function Network (a type of Graphical model)

Cost function network (X, E)

- a sequence X of discrete variables x_i, domain D_i

Cost Function Network (a type of Graphical model)

Cost function network (X, E)

- a sequence X of discrete variables x_i, domain D_i
- a set E of cost functions expressed as tables (or dedicated functions)

Cost Function Network (a type of Graphical model)

Cost function network (X, E)

- a sequence X of discrete variables x_i, domain D_i
- a set E of cost functions expressed as tables (or dedicated functions)
- $e_S \in E$ is a numerical function $\prod_{i \in S} D_i$ (possibly infinite costs)

Cost Function Network (a type of Graphical model)

Cost function network (X, E)

- a sequence X of discrete variables x_i, domain D_i
- a set E of cost functions expressed as tables (or dedicated functions)
- $e_S \in E$ is a numerical function $\prod_{i \in S} D_i$ (possibly infinite costs)
- a solution minimizes the joint cost $E = \sum_{e_S \in E} e_S$ (WCSP, NP-complete)

Cost Function Network (a type of Graphical model)

Cost function network (X, E)

- a sequence X of discrete variables x_i, domain D_i
- a set E of cost functions expressed as tables (or dedicated functions)
- $e_S \in E$ is a numerical function $\prod_{i \in S} D_i$ (possibly infinite costs)
- a solution minimizes the joint cost $E = \sum_{e_S \in E} e_S$ (WCSP, NP-complete)

Graphical models?

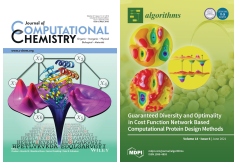
- The interactions captured by the model can be represented as a graph
- Variables are vertices
- They are connected by an edge if they interact (participate together in a function)
- Cost Function Networks are closely related to Markov Random Fields

Exact vs. Stochastic search (See JFPC'21 slides)

Large input (> 1GB) **NP-hard problem**

Toulbar2 is able to…

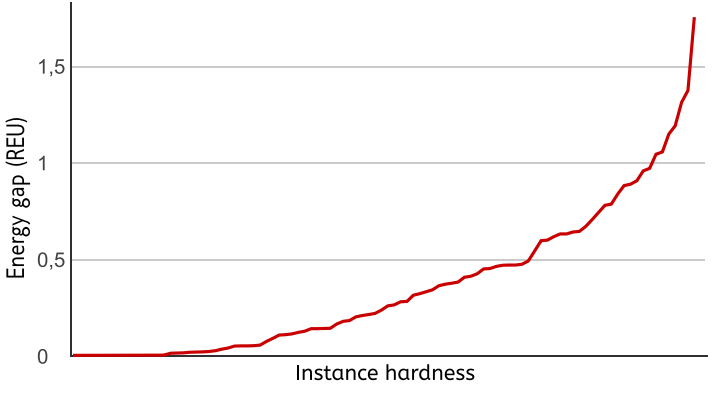
- provide a proven zero/bounded gap minimum energy solution[27]
- exhaustively enumerate sequences close to it
- provide sequence libraries with guaranteed diversity.[20]
- in sequence-conformation spaces of size $> 10^{400}$



Rosetta's Monte Carlo Simulated Annealer increasingly fails to find the optimal sequence[a]

[a] David Simoncini et al. "Guaranteed Discrete Energy Optimization on Large Protein Design Problems". In: *Journal of Chemical Theory and Computation* 11.12 (2015), pp. 5980–5989. DOI: 10.1021/acs.jctc.5b00594.

Unbounded error



Asymptote: **Size matters!**

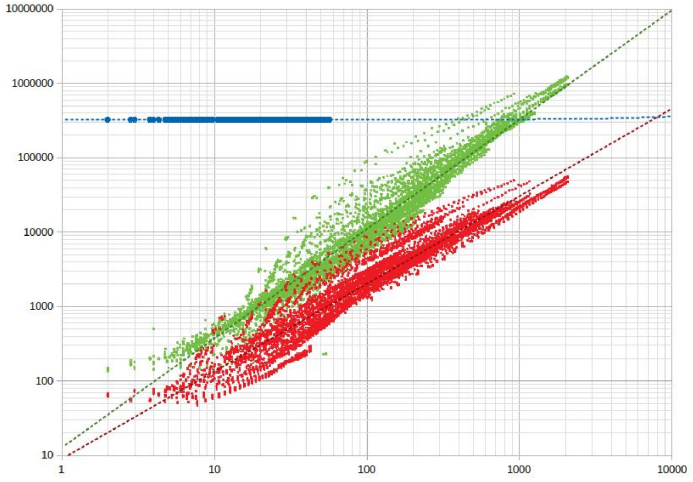
Asymptotic convergence can be arbitrarily slow…

Guaranteed Discrete Energy Optimization on Large Protein Design Problems

David Simoncini[†], David Allouche[†], Simon de Givry[†], Céline Delmas[†], Sophie Barbe[‡§⊥], and Thomas Schiex[*†]

JCTC Journal of Chemical Theory and Computation

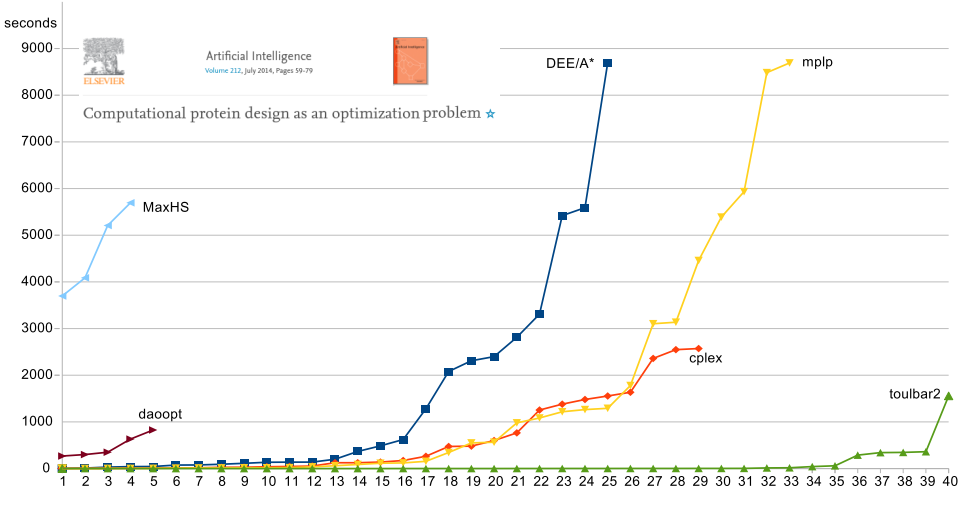
Quantum computing (DWave),Toulbar2 & SA[1]



DWave approximations		*kcal/mol*
gap > 1.16, 90% of the time	> 4.35, 50% of the time	> 8.45, 10% of the time

[1] Vikram Khipple Mulligan et al. "Designing Peptides on a Quantum Computer". In: *bioRxiv* (2019), p. 752485.

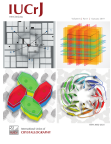
Toulbar2 vs. CPLEX, MaxHS…(real instances)



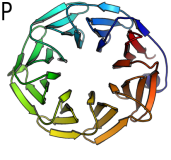
# of instances solved (*X*) within a per instance cpu-time limit (*Y*)

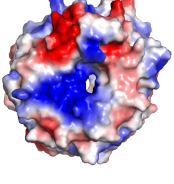
Designing a self-assembling β-propeller

Coll. A. Voet (KU Leuven), D. Simoncini[17]



2OVP

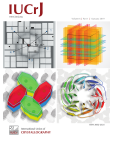




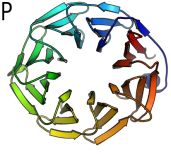
Designing a self-assembling β-propeller

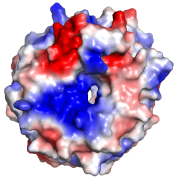
Coll. A. Voet (KU Leuven), D. Simoncini[17]

-  Tako: (R)evolution + Rosetta/talaris14 8 fold

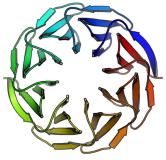


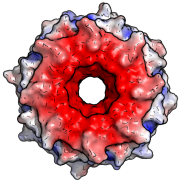
2OVP





Tako

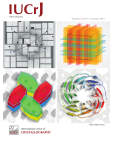


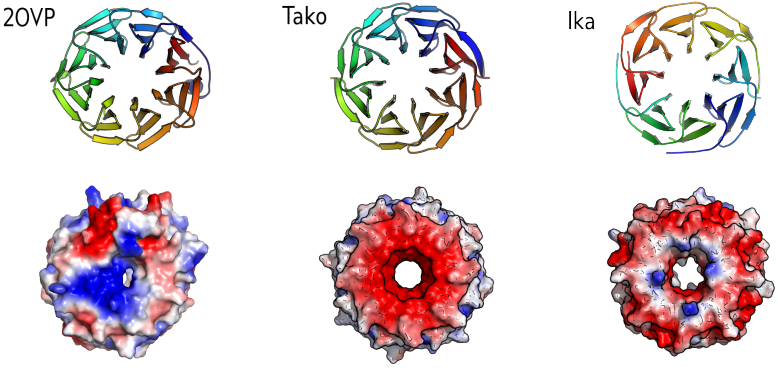


Designing a self-assembling β-propeller

Coll. A. Voet (KU Leuven), D. Simoncini[17]

-  Tako: (R)evolution + Rosetta/talaris14 — 8 fold
-  Ika: toulbar2 + talaris14 — 4 fold

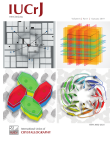




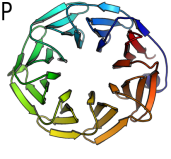
Designing a self-assembling β-propeller

Coll. A. Voet (KU Leuven), D. Simoncini[17]

-  Tako: (R)evolution + Rosetta/talaris14 — 8 fold
-  Ika: toulbar2 + talaris14 — 4 fold

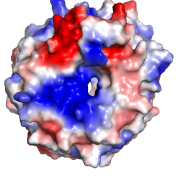


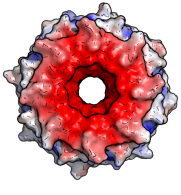
2OVP

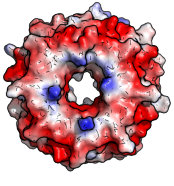


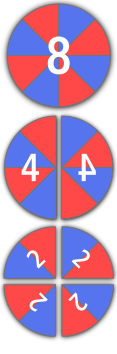
Tako

Ika









Using this new computational capacity for better modeling[2]

Capturing protein flexibility[4] through Multi-state design

Find a sequence that stabilizes multiple structures at the same time

What for?

- Bound and unbound conformations for enzymes, or binders
- Conformational switches
- All proteins are flexible!
- Can be achieved using just constraints (no new algorithm)

[2] Jelena Vucinic et al. "Positive multistate protein design". In: *Bioinformatics* 36.1 (2020), pp. 122–130.

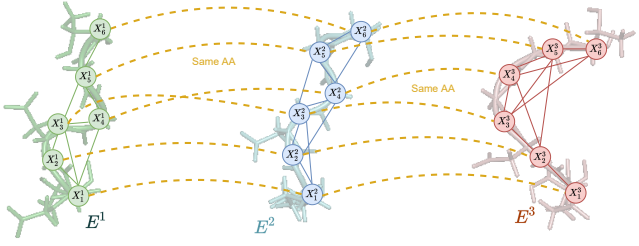
Using this new computational capacity for better modeling[2]

Capturing protein flexibility[4] through Multi-state design

Find a sequence that stabilizes multiple structures at the same time

What for?

- Bound and unbound conformations for enzymes, or binders
- Conformational switches
- All proteins are flexible!
- Can be achieved using just constraints (no new algorithm)



[2] Jelena Vucinic et al. "Positive multistate protein design". In: *Bioinformatics* 36.1 (2020), pp. 122–130.

Improves design quality at reasonable computational costs[3]

How correctly does it reconstruct natural proteins?

- Native sequence recovery (NSR)

Improvement over traditional Single State Design

	NMR	X-ray
NSR	+ 15,6 %	+8 %

[3] Jelena Vucinic et al. "Positive multistate protein design". In: *Bioinformatics* 36.1 (2020), pp. 122–130.

Injecting Machine Learned information

Energy is imperfect

- Approximations: solvent effect…
- Ignored: polarisability, expressability…
- Needs more information, extracted from *data*

Injecting Machine Learned information

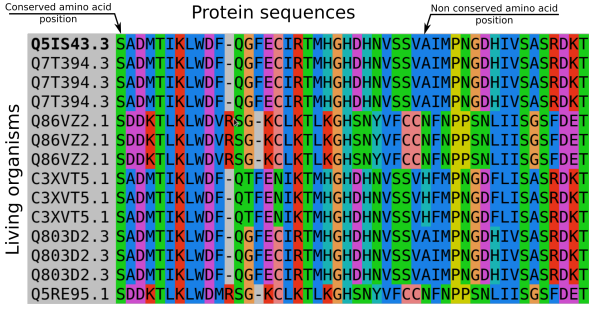
Energy is imperfect

- Approximations: solvent effect…
- Ignored: polarisability, expressability…
- Needs more information, extracted from *data*

Evolutionary information

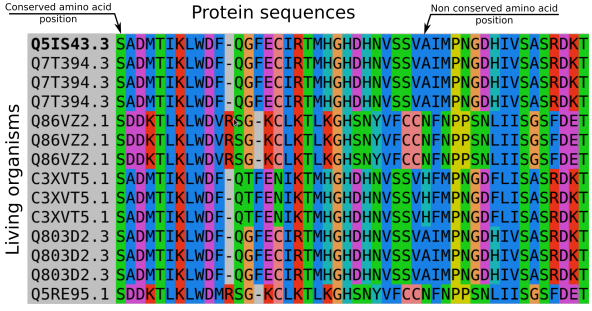
- Use a Multiple alignment of similar proteins (homologs)

A multiple alignment with conserved positions



- Used to force amino acid choice (constraint) at conserved positions.
- or bias by "Position Specific Score Matrices" (frequency / position)

A multiple alignment with conserved positions



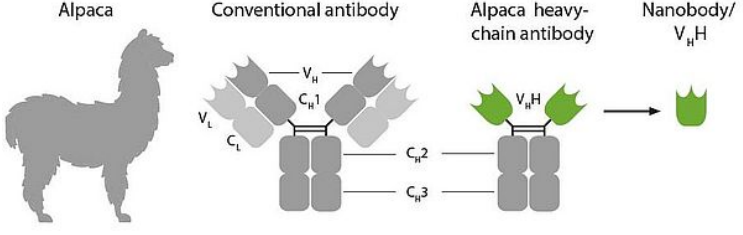
- Used to force amino acid choice (constraint) at conserved positions.
- or bias by "Position Specific Score Matrices" (frequency / position)

Algorithms for contact-map predictions[24] (MRF estimation)

- Identifies how close residues prefer to co-vary
- Combine this information with energy (linear combination)

Designing a new nanobody scaffold (coll. TBI, INSERM-CRCT)

- Using Rosetta score function and rotamer library
- Trying to satisfy several constraints (originality, composition…)
- Multi-state design:[29] multi-CDR loops compatible
- MSA-extracted evolutionary preferences



Designing a new nanobody scaffold (coll. TBI, INSERM-CRCT)

- Using Rosetta score function and rotamer library
- Trying to satisfy several constraints (originality, composition…)
- Multi-state design:[29] multi-CDR loops compatible
- MSA-extracted evolutionary preferences

Limited experimental power

- Over 6 sequences designed without evolutionary information: 3 expressed
- Over 3 sequences designed with evolutionary information: 3 expressed
- Much more power in recent Science/PLOS papers[21,23]
- Positive results on three environmental-friendly enzymes (coll. S. Barbe, TBI).

CFNs bring ML and Automated reasoning together

With Cost Function Network algorithms, one can…

- Model the problem as a CFN: knowledge (logic, energy)

CFNs bring ML and Automated reasoning together

With Cost Function Network algorithms, one can…

- Model the problem as a CFN: knowledge (logic, energy)
- Learn more CFNs from data (e.g. from natural sequences)[5]

CFNs bring ML and Automated reasoning together

With Cost Function Network algorithms, one can…

- Model the problem as a CFN: knowledge (logic, energy)
- Learn more CFNs from data (e.g. from natural sequences)[5]
- Combine the models by scaling/adding/connecting them together

CFNs bring ML and Automated reasoning together

With Cost Function Network algorithms, one can…

- Model the problem as a CFN: knowledge (logic, energy)
- Learn more CFNs from data (e.g. from natural sequences)[5]
- Combine the models by scaling/adding/connecting them together
- Add further design constraints/preferences: desired properties

CFNs bring ML and Automated reasoning together

With Cost Function Network algorithms, one can…

- Model the problem as a CFN: knowledge (logic, energy)
- Learn more CFNs from data (e.g. from natural sequences)[5]
- Combine the models by scaling/adding/connecting them together
- Add further design constraints/preferences: desired properties
- Solve them with toulbar2 to get your new design (NP-hard)

Open source

https://github.com/toulbar2/toulbar2
https://github.com/toulbar2/CFN-learn

Exciting DL revolution in Protein Design too

Rigid body DL design approaches[8]

- Strongly inspired from NLP approaches (sequence, translation: transformers,...)
- Enriched by 3D geometry: SE(3) equivariance
- Coarse grained approches (backbone atoms only) mapping a backbone to a sequence
- Learning $P(s_i = AA|\mathrm{environment})$ for design[1,12]

Exciting DL revolution in Protein Design too

Rigid body DL design approaches[8]

- Strongly inspired from NLP approaches (sequence, translation: transformers,...)
- Enriched by 3D geometry: SE(3) equivariance
- Coarse grained approches (backbone atoms only) mapping a backbone to a sequence
- Learning $P(s_i = AA|\text{environment})$ for design[1,12]

Design requires to impose 'fitness' constraints on the output

- Non trivial for Deep Learning
- Variational auto-encoders latent space interpolation[6]
- Driven generative adversarial networks[9]

Unleashing the power of DL-based structure prediction for design

Inverting DL Structure predictors (TrRosetta/RosettaFold[3]) α-Fold 2

- These networks somehow capture the sequence/structure relationship
- Back-propagation from a (sequence(s), structure) pair: symbolic sequence gradients
- Seems to be able to fight the "ill-posed problem" issue[18]
- Best performance obtained by injecting DL prediction as energy bias terms
- These terms can also be swallowed by Cost Function Networks.

Conclusion

- Designing new proteins with new functions can have strong real-world impact
- Design requires to assemble knowledge, experience (data), and constraints on the output
- Cost Function networks algorithms offer new capacities for CPD (NP-hard$\neq$ intractable)
- They rigourously combine physical energy with design constraints
- And can also swallow Machine/Deep Learned information
- Deep Learning may contribute to solve the long standing issue of 'alternative structures"
- But still needs to improve its capacities to satisfy output constraints

We still need to get rid of plenty of assumptions: come and dive in the amazing world of molecular design!

Thanks

AI/toulbar2

S. de Givry (INRA)
G. Katsirelos (INRA)
M. Zytnicki (PhD, INRA)
D. Allouche (INRA)
M. Ruffini (INRA)
H. Nguyen (PhD, INRA)
C. Brouard (ML, INRA)
M. Cooper (IRIT, Toulouse)
J. Larrosa (UPC, Spain)
F. Heras (UPC, Spain)
M. Sanchez (Spain)
E. Rollon (UPC, Spain)
P. Meseguer (CSIC, Spain)
G. Verfaillie (ONERA, ret.)
JH. Lee (CU. Hong Kong)
C. Bessiere (LIMM, Montpellier)
JP. Métivier (GREYC, Caen)
S. Loudni (GREYC, Caen)
M. Fontaine (GREYC, Caen),...

Protein Design

A. Voet (KU Leuven)
A. Olichon (INSERM)
D. Simoncini (UFT, Toulouse)
S. Barbe (INSA, Toulouse)
C. Dumont (INSA, Toulouse)
J. Vucinic (INRA/INSA)
S. Traoré (PhD, CEA)
C. Viricel (PhD)
K. Zhang (Riken, CBDR)
S. Tagami (Riken, CBDR)
RosettaCommons (U. Washington)
W. Sheffler (U. Washington)
V. Mulligan (Flatiron Institute)
PyRosetta (U. John Hopkins)
B. Donald (U. North Carolina)
K. Roberts (U. North Carolina)
T. Simonson (Polytechnique)
J. Cortes (LAAS/CNRS),…

My apologies to those missing in these lists. Even imperfect lists seem better than no list

References I

[1] Namrata Anand et al. "Protein sequence design with a learned potential". In: *bioRxiv* (2020).

[2] C. Anfinsen. "Principles that govern the folding of protein chains". In: *Science* 181.4096 (1973), pp. 223–253.

[3] Minkyung Baek et al. "Accurate prediction of protein structures and interactions using a 3-track network". In: *bioRxiv* (2021).

[4] Younes Bouchiba et al. "Molecular flexibility in computational protein design: an algorithmic perspective". In: *Protein Engineering, Design and Selection* 34 (2021).

[5] Céline Brouard, Simon de Givry, and Thomas Schiex. "Pushing data into CP models using Graphical Model Learning and Solving". In: *Principles and Practice of Constraint Programming–CP 2020*. Springer, 2020.

[6] Payel Das et al. "PepCVAE: Semi-Supervised Targeted Design of Antimicrobial Peptide Sequences". In: *arXiv:1810.07743 [cs, q-bio, stat]* (Nov. 2018). arXiv: 1810.07743. URL: `http://arxiv.org/abs/1810.07743` (visited on 12/15/2020).

[7] Pablo Gainza, Kyle E Roberts, and Bruce R Donald. "Protein design using continuous rotamers". In: *PLoS Comput Biol* 8.1 (2012), e1002335.

[8] Wenhao Gao et al. "Deep learning in protein structural modeling and design". In: *Patterns* (2020), p. 100142.

[9] Anvita Gupta and James Zou. "Feedback GAN for DNA optimizes protein functions". en. In: *Nature Machine Intelligence* 1.2 (Feb. 2019). Number: 2 Publisher: Nature Publishing Group, pp. 105–111. ISSN: 2522-5839. DOI: `10.1038/s42256-019-0017-4`. URL: `https://www.nature.com/articles/s42256-019-0017-4` (visited on 12/15/2020).

[10] Mark A Hallen and Bruce R Donald. "Protein design by provable algorithms". In: *Communications of the ACM* 62.10 (2019), pp. 76–84.

[11] Barry Hurley et al. "Multi-language evaluation of exact solvers in graphical model discrete optimization". In: *Constraints* (2016), pp. 1–22.

References II

[12] Bowen Jing et al. "Learning from Protein Structure with Geometric Vector Perceptrons". In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL: https://openreview.net/forum?id=1YLJDvSx6J4.

[13] Oliver Kullmann. "The Science of Brute Force". In: *Communications of the ACM* (2017).

[14] Andrew Leaver-Fay et al. "ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules.". In: *Methods Enzymol.* 487 (2011), pp. 545–574. ISSN: 1557-7988.

[15] S C Lovell et al. "The penultimate rotamer library.". In: *Proteins* 40.3 (Aug. 2000), pp. 389–408. ISSN: 0887-3585. URL: http://www.ncbi.nlm.nih.gov/pubmed/10861930.

[16] Vikram Khipple Mulligan et al. "Designing Peptides on a Quantum Computer". In: *bioRxiv* (2019), p. 752485.

[17] Hiroki Noguchi et al. "Computational design of symmetrical eight-bladed β-propeller proteins". In: *IUCrJ* 6.1 (2019).

[18] Christoffer Norn et al. "Protein sequence design by explicit energy landscape optimization". In: *bioRxiv* (2020).

[19] N. Pierce et al. "Conformational splitting: A more powerful criterion for dead-end elimination". In: *Journal of computational chemistry* 21.11 (2000), pp. 999–1009.

[20] Manon Ruffini et al. "Guaranteed Diversity and Optimality in Cost Function Network Based Computational Protein Design Methods". In: *Algorithms* 14.6 (2021), p. 168.

[21] William P Russ et al. "An evolution-based model for designing chorismate mutase enzymes". In: *Science* 369.6502 (2020), pp. 440–445.

[22] Ilan Samish. *Computational protein design*. Springer, 2017.

[23] Samuel Schmitz et al. "Rosetta design with co-evolutionary information retains protein function". In: *PLOS Computational Biology* 17.1 (2021), e1008568.

References III

[24] Stefan Seemayer, Markus Gruber, and Johannes Söding. "CCMpred—fast and precise prediction of protein residue–residue contacts from correlated mutations". In: *Bioinformatics* 30.21 (2014), pp. 3128–3130.

[25] Maxim V Shapovalov and Roland L Dunbrack. "A smoothed backbone-dependent rotamer library for proteins derived from adaptive kernel density estimates and regressions". In: *Structure* 19.6 (2011), pp. 844–858.

[26] David Simoncini et al. "Guaranteed Discrete Energy Optimization on Large Protein Design Problems". In: *Journal of Chemical Theory and Computation* 11.12 (2015), pp. 5980–5989. DOI: 10.1021/acs.jctc.5b00594.

[27] Seydou Traoré et al. "A New Framework for Computational Protein Design through Cost Function Network Optimization". In: *Bioinformatics* 29.17 (2013), pp. 2129–2136.

[28] P Tuffery et al. "A new approach to the rapid determination of protein side chain conformations". In: *Journal of Biomolecular structure and dynamics* 8.6 (1991), pp. 1267–1289.

[29] Jelena Vucinic et al. "Positive multistate protein design". In: *Bioinformatics* 36.1 (2020), pp. 122–130.