Computational Protein Design as an Optimization Problem

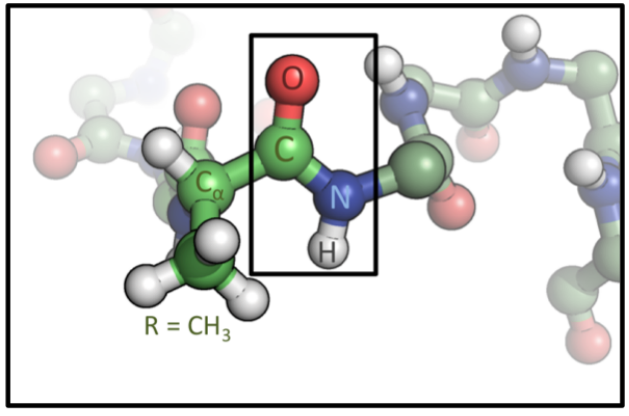
T. Schiex
D. Allouche, Isabelle André, Sophie Barbe, Jessica Davies, Simon de Givry, George Katsirelos, Barry O'Sullivan, Steve Prestwich, David Simoncini, Seydou Traoré

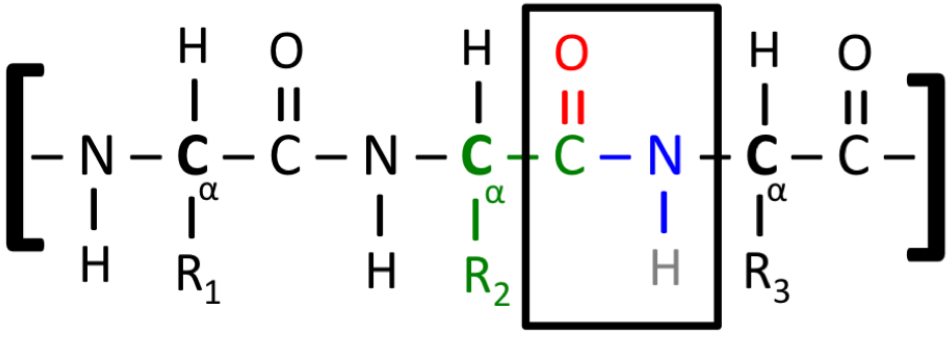
IJCAI'2015
Advances in Bioinformatics and Artificial Intelligence
Bridging the Gap

What is a protein ? (Kudos to wikipedia)

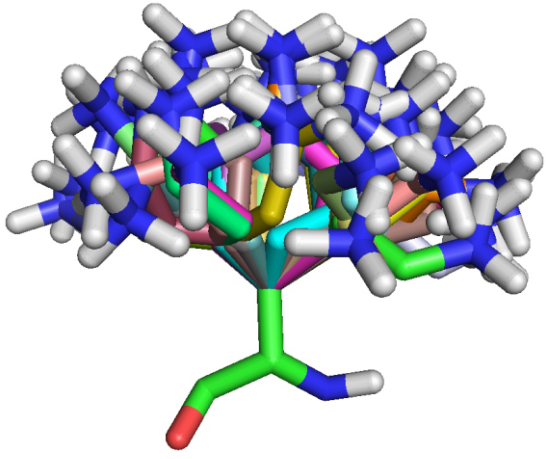
Amino acids, proteins

- Proteins are linear chains of amino-acids (20 natural AAs).
- All AAs share a common "core" and have a variable side-chain.





Side-chains are flexible (ARG)



Protein Design

Why ?

- Proteins have various functions in the cell: catalysis, signaling, recognition, regulation...
- Efficient, biodegrable, 10^6 to 10^{20} speedups
- Some reactions / ligands miss enzymes / partners.
- Nano-technologies (shape more than function).
- Medecine, cosmetics, food, bio-energies...

Protein Design

Protein function linked to its 3D shape through its amino acid composition.

Protein design's aim

Identify sequences that have a suitable function (shape).

Issue

There are 20^n proteins of length n.
Impossible to synthesize and test all of them.

The CPD problem - stability variant

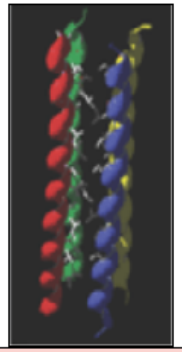
Preparation

- A backbone is chosen/built from a known protein/structure (or *de novo*).
- Positions are set as mutable, flexible or rigid
- The aim is to find an AA sequence that folds, stably, in the backbone.

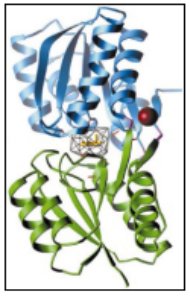
Issues

- CPD is a sort of inverse of folding.
- But folding is far from being a solved problem

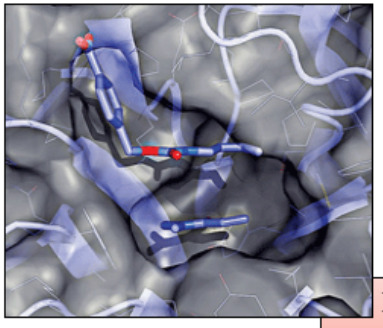
Successes of Protein Design



- Coiled-coils *Harbury et al (Science 1998)*

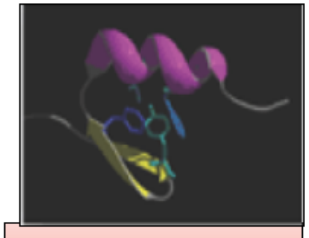


- Sensors & receptors toward non native ligands *Hellinga et al (Nature, 2003)*



2010

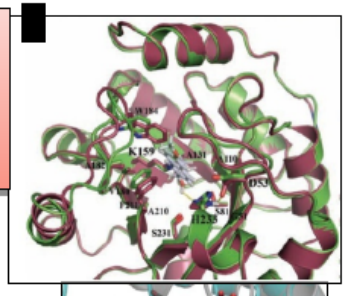
- Design of Biocatalyst for a stereoselective & bimolecular reaction (Diels-alder) *Siegel et al (Science 2010)*

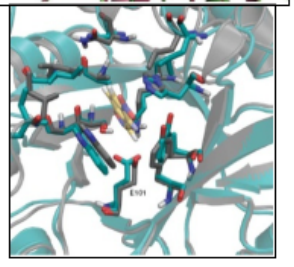


- Zinc-finger *Dahiyat et al (Science 1997)*

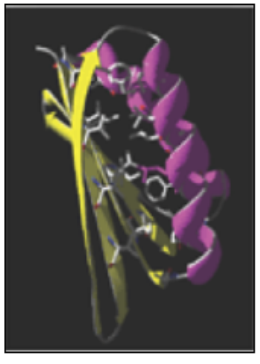
2008

- New catalyst for a multi-step-reaction (retro-aldolase) *Jiang et al (Science 2008)*





2003



- A novel topology (Top7) *Khulman et al (Science 2003)*

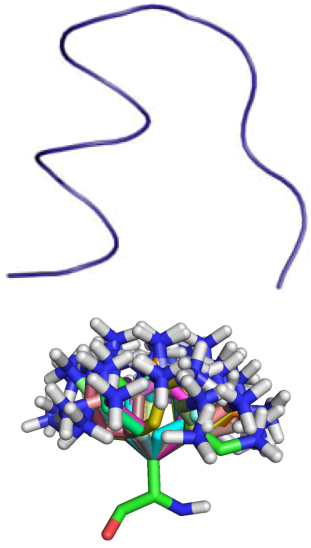
- De novo design of new functional enzyme (Kemp elimination) *Rothlisberger et al (Nature 2008)*

1997

The (basic) CPD problem: search space

Rigid backbone variant

1. Assume a rigid protein backbone.
2. Choose 1 AA among possible ones at each mutable position.
3. Spatial conformation discretized in rotamers.
4. Statistically frequent orientations.
5. Several 100's rotamers per position.



Search Space

1. Fully discrete description, defined by a choice of rotamer (AA $\times$ conformation) for each position.
2. Search space can be $\approx 250^n$

Stable = minimum energy (GMEC, NP-hard [PW02])

Energy: interactions between atoms.

- Electrostatic, van der Waals (Amber)
- Dihedral torsion angles, Implicit Solvation (EEF1)
- "Statistical terms" (Talaris)
- Cutoff functions

Pairwise decomposable energy

- backbone/backbone (constant)
- backbone/rotamer (depends on rotamer)
- rotamer/rotamer (depends on pairs of rotamers)

$$E(c) = E_{\varnothing} + \sum_{i=1}^{n} E(i_r) + \sum_{i<j} E(i_r, j_s)$$

Dedicated CPD Methods

Dominance / Dead End Elimination [Des+92]

$$E(i_a) + \sum_{j \neq i}^{n} \min_c E(i_a, j_c) > E(i_b) + \sum_{j \neq i}^{n} E \max_b E(i_b, j_c)$$

Strengthened by [Gol94]

$$E(i_a) - E(i_b) + \sum_{j \neq i}^{n} \min_c \left[E(i_a, j_c) - E(i_b, j_c) \right] > 0$$

Many further enhancements (splitting, pairs...). Polynomial time pre-processing.

In CSP/SAT [Coo97; NR00; LRD12]

Known as "(soft) substituability" in CSP and Dominating 1-clause rule in MaxSAT.

DEE + A^*

polytime DEE, GMEC NP-hard

- DEE cannot reduce all domains to singletons
- Followed by A^* best-first search using the following lower bound (admissible heuristics) [GLD08]:

$$\underbrace{\sum_{i=1}^{d} E(i_r) + \sum_{j=i+1}^{d} E(i_r, j_s)}_{\text{Assigned}} + \sum_{j=d+1}^{n} \Big[\underbrace{\min_s (E(j_s) + \sum_{i=1}^{d} E(i_r, j_s)}_{\text{Forward checking}} + \underbrace{\sum_{k=j+1}^{n} \min_u E(j_s, k_u))}_{\text{DAC counts}} \Big]$$

Lower bound

- Same as a lower bound introduced in AI (WCSP) in 1994 [Wal95].
- Obsolete.

T. Schiex. "Arc consistency for soft constraints". In: *Principles and Practice of Constraint Programming - CP 2000*. Vol. 1894. LNCS. Singapore, Sept. 2000, pp. 411–424

Solving the Fixed Backbone CPD problem

Our targets

- Identify a most efficient model/solving technique for solving the rigid backbone/rotamer based/pairwise energy CPD problem.
- Do one of the first large spectrum comparison of NP-complete optimization techniques (AI: CSP, CP, SAT, MRF and OR: ILP, QP, QPBO) on one well defined and important optimization problem.
- Learn from it.

Cost Function Networks (aka WCSP or MRF)

Cost Function Network (X, D, E)

1. $X = (1, \ldots, n)$, n variables (indices).
2. $D = (D_1, \ldots, D_n)$, n domains
3. C set of non negative integer cost functions c_S.
4. $c_S : D^S = \prod_{D_i, i \in S} \rightarrow \{0, \ldots, k\}$

$$\min_{t \in D^X} E(t) = \sum_{c_S \in C} c_S(t[S])$$

- k is an intolerable cost. May be finite or not.
- Cost functions defined as tables, analytic formulas or predicates (global cost functions).
- Bounded addition, subtraction. $c_\varnothing$ is a lower bound.

Solving techniques (CFN solver: toulbar2)

Inspired by Constraint Satisfaction

1. Backtrack becomes Branch and Bound (Depth First)
2. Local consistency reformulates the problem in a more explicit equivalent problem (Equivalence Preserving Transformation).
3. Provides non naive $c_\varnothing$ (lb), incremental.

Many additional techniques

1. Dynamic variable/value ordering, learning heuristics
2. On the fly variable elimination,
3. Tree-decomposition for Branch and Bound (BTD)...

Won the UAI 2010 and 2014 *approximate* inference challenges (2nd in 2012).

Equivalence Preserving Transformation

Arc EPT

- A cost function c_S, here c_{ij}.
- EPT `Project` $(\{ij\}, \{i\}, a, \alpha)$ shifts cost α between $c_i(i_a)$ and the cost function c_{ij}.
- projection $(\alpha \geq 0)$, extension $(\alpha < 0)$.

Precondition: $-c_i(i_a) \leq \alpha \leq \min_{t' \in D^{ij}, t'[i]=i_a} c_{ij}(t')$;

Procedure `Project`$(\{i,j\}, \{i\}, a, \alpha)$

$c_i(i_a) \leftarrow c_i(i_a) \oplus \alpha$;

foreach *($t' \in D^{ij}$ such that $t'[i] = i_a$)* **do**

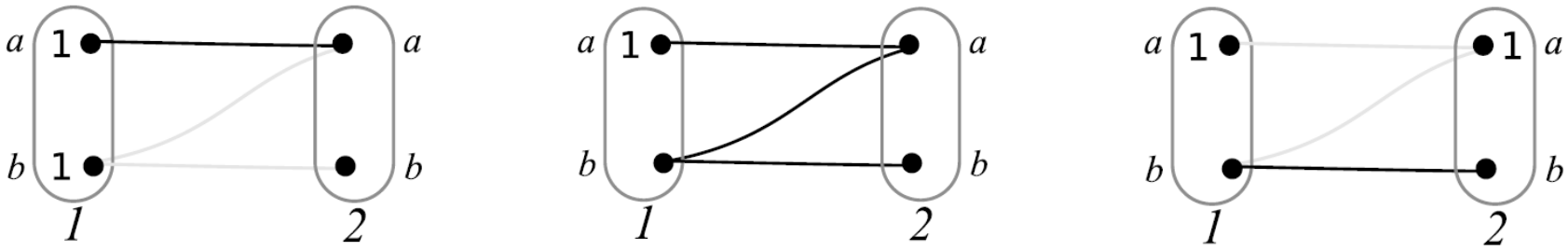
$c_{ij}(t') \leftarrow c_{ij}(t') \ominus \alpha$;

end

$\oplus$ is $m-$bounded addition. Pseudo-inverse $\ominus$ (you can take whatever you want from k).

Example

$\texttt{Project}(\{1,2\},\{1\},b,1)$ $\leftarrow$ $\texttt{Project}(\{1,2\},\{2\},a,1)$ $\rightarrow$



$\rightarrow$ $\texttt{Project}(\{1,2\},\{1\},b,-1)$ $\leftarrow$ $\texttt{Project}(\{1,2\},\{2\},a,-1)$ $\leftarrow$

$\Downarrow$ $\texttt{Project}\ (\{1\},\varnothing,[],1)$

$c_{\varnothing} = 1$

Non confluent (multi fix-point). Not all as good in term of lb.
With integer costs, finding the best fix-point is NP-hard [CS04].

Local consistencies

Polynomial time filtering

- Node consistency: at the variable level. Moves cost to $c_\varnothing$, upper bounding ($c_i(a) + c_\varnothing = k$).
- Arc consistency, directional AC, Full directional AC, EDAC, VAC, OSAC (Optimal Soft Arc Consistency).
- VAC and OSAC solve submodular subproblems.

T. Schiex. "Arc consistency for soft constraints". In: *Principles and Practice of Constraint Programming - CP 2000*. Vol. 1894. LNCS. Singapore, Sept. 2000, pp. 411–424

M. Cooper et al. "Soft arc consistency revisited". In: *Artificial Intelligence* 174 (2010), pp. 449–478

Universally used principle

Also underlies Weighted MaxSAT "resolution" and Markov Random Field "reparametrization by Message Passing" (TRW-S, MPLP, SRMP,...).

J. Larrosa and F. Heras. "Resolution in Max-SAT and its relation to local consistency in weighted CSPs". In: *Proc. of the 19th IJCAI*. Edinburgh, Scotland, 2005, pp. 193–198

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009

Optimal Soft Arc Consistency

OSAC

An LP that identifies a set of EPTs (rational costs) that maximizes the lower bound. After propagation of hard (k) costs using Arc Consistency.

maximize $\sum_i u_i$ where

- u_i: amount of cost projected from c_i to $c_\varnothing$
- $p_{i_a}^S$: amount of cost projected from c_S to i_a

$$\forall i \in X, \forall a \in d_i, \quad c_i(a) - u_i + \sum_{(c_S \in C),(i \in S)} p_{i,a}^S \geq 0$$

$$\forall c_S \in C, |S| > 1, \forall t \in \ell(S) \quad c_S(t) - \sum_{i \in S} p_{i,t[\{i\}]}^S \geq 0$$

M C. Cooper, S. de Givry, and T. Schiex. "Optimal soft arc consistency". In: *Proc. of IJCAI'2007*. Hyderabad, India, Jan. 2007, pp. 68–73

M.I. Schlesinger. "Sintaksicheskiy analiz dvumernykh zritelnikh signalov v usloviyakh pomekh (Syntactic analysis of two-dimensional visual signals in noisy conditions)". In: *Kibernetika* 4 (1976), pp. 113–130

M. Cooper et al. "Soft arc consistency revisited". In: *Artificial Intelligence* 174 (2010), pp. 449–478

ILP models

ILP for WCSP/CPD/MRF

1. Koster's ILP model for WCSP [KHK99]. Used for CPD in [KCS05]. Is the "local polytope" of MRF [Wer07]
2. One 0/1 variable per value and per pair (relaxable for pairs).

$$\begin{array}{lll} \min & \sum_{i,r} E(i_r).d_{i,r} + \sum_{i,r,j,s} E(i_r,j_s).p_{i,r,j,s} & \\ \text{s.t.} & \sum_r d_{i,r} = 1 & (\forall i) \\ & \sum_s p_{i,r,j,s} = d_{i,r} & (\forall i,r,j) \end{array}$$

Relaxation = dual of OSAC LP

1. Arc consistencies: limited Block Coordinate Descent algorithms for the dual of this specific (?) LP
2. Any LP can be reduced to it in linear time [PW15].

As quadratic 0/1 programs

QP - Cplex

$$\min \sum_{i,r} E(i_r).d_{ir} + \sum_{\substack{i,r,j,s \\ j>i}} E(i_r, j_s).d_{ir}.d_{js}$$

$$\text{s.t.} \quad \sum_{r} d_{ir} = 1 \quad (\forall i)$$

$$d_{ir} \in \{0,1\} \quad (\forall i, r)$$

QPBO - MaxCut (BiqMac/SDP bound): Big M

$$\min \sum_{i,r} (E(i_r) - N).d_{ir} + \sum_{\substack{i,r,j,s \\ j>i}} (E(i_r, j_s) - N).d_{ir}.d_{js} + \sum_{\substack{i,r,s \\ s>r}} M.d_{ir}.d_{is}$$

MRF methods

daoopt [OD12]

1. won the UAI (PIC) approximate inference challenge in 2012.
2. lower bound based on "Mini-buckets" (dynamic programming with bounded width).
3. tree-decomposition used in AND/OR search

MPLP [SCL12]

1. Dual relaxed solution (lower bound) provided by BCD optimization.
2. Strengthens the Dual by including empty ternary cost functions.
3. Heuristics for Primal.
4. Iterative, no search.

Partial Weighted maxSAT

PW MaxSAT

- Boolean variables, litteral: variable or its negation
- Weighted clauses: disjunction of litterals.
- criteria: sum of weight of violated clauses.
- B&B - Core solvers: MiniMaxSat [HLO08],akMaxSat [Kue10] - bincd [HMM11],wpm1/2 [ABL09; ABL10],MaxHS [DB13]

Direct encoding

- d_{i_a}: use i_a
- $\forall i_r, i_s, i_r \neq i_s, (\neg d_{i_r} \vee \neg d_{i_s})$ (AMO)
- $\forall i, (\bigvee_r d_{i_r})$ (ALO)
- $(\neg d_{i_r}, E(i_r)$ and $(\neg d_{i_r} \vee \neg d_{j_s}, E(i_r, j_s))$

Tuple encoding

Property [Bac07]

In CSP, Unit Propagation on this encoding enforces AC on the CSP. Close to the ILP model.

Direct encoding

- d_{i_a} + AMO + ALO.
- $p_{i_r j_s}$: pair i_a, j_s is used.
- $\forall i_r, j_s : (d_{i_r} \vee \neg p_{i_r j_s})$ and $(d_{j_s} \vee \neg p_{i_r j_s})$.
- $\forall i_r, j (\neg d_{i_r} \vee \bigvee_s p_{i_r j_s})$
- idem for $E(i_r)$, $\forall i_r, j_s (\neg p_{i_r j_s}, E(i_r, j_s))$

Pure CP - Soft as Hard model [PRB00]

General idea

1. add one "cost" variable to every cost function to make a ternary constraint.
2. use a global "Sum" constraint on these new cost variables.

Can be expressed in MiniZinc [Mar+08]

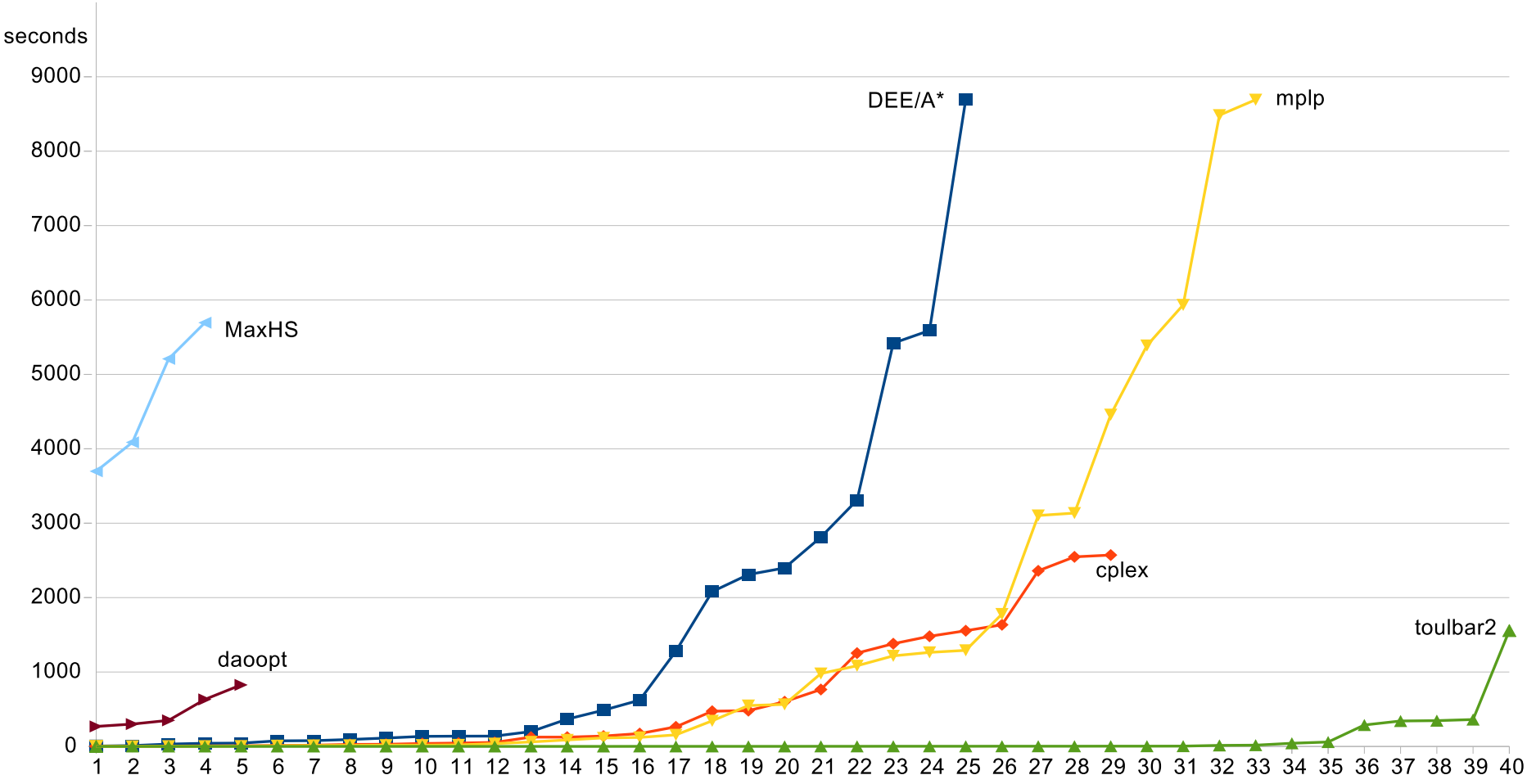
1. GeCode (`http://www.gecode.org/`),
2. Mistral (Python `numberjack` interface, `http://numberjack.ucc.ie/`),
3. Opturion/CPX `http://www.opturion.com/cpx.html`

A realistic benchmark: 35+12 designs tested

The designs

1. Extracted from the litterature,
2. Good resolution of the PDB structures,
3. Structure preparation,
4. Domains assigned based on accessibility,
5. Amber + EEF1 + No cutoff (almost complete graphs)
6. Variable search space size, from 10^{26} to 10^{249}

Results - 9000 seconds



From failures...

Analysis

1. **QP by Cplex**: dense model, but weak and somewhat expensive lb (very large node file, large gaps).
2. **SDP based QPO**: probably tight lower bound, but far too expensive (few nodes explored after several hours). biqmac library of MaxCut beasley instances size 100: solved in 1" by tb2, 1' by biqmac.
3. **MaxSAT, direct**: branch and bound solvers very fast (36k nodes/sec, 100 times faster than tb2). found incumbent solutions but never started the optimality proof. Weak lb (root $= 25\%$ of optimum, tb2 always $> 97\%$).
4. **MaxSAT, tuple**: b&b,strong lower bound (should be similar to VAC for core based solvers). Still weaker than tb2 and very slow (2 nodes before timeout at best for akmaxsat). No incumbent. Core based better (maxHS, good lb).

... to Successes

Analysis

1. **Daoopt**: almost complete graphs. Not ideal for tree decomposition based methods.
2. **DEE/A***: surprisingly good given the lower bound used. Very strong preprocessing.
3. **ILP - Cplex**: LP bound similar to OSAC (dual). tb2 has upper bounding. Similar number of nodes but tb2 much faster (ILP: 1 to 40 nodes / minutes, tb2: 1 to 40 thousand).
4. **MPLP**: no branching but able to solve few more problems than CPLEX.

A Lesson for (AI) Optimization

The lower bounding/search efforts compromise is not understood, nor exploited. But may be crucial.

Enumerating all suboptimal solutions on 35 designs

All within 2 kcal/mol of GMEC, 100 h, tb2 and DEE/A*

- Enumeration feasible for 1 design only (DEE/A*)
- Enumeration finished for all solved designs (CFN).
- More than 1 billion sequence-conformations for one design.

May be useful for partition function estimation [Vir+15].
Additional progresses since.

Final note and Acknowledgments

This is all for a rigid backbone. Modern CPD increasingly uses "flexible" representations (eg. with a backbone ensemble).

Thanks to...

- Bruce Donald and Kyle Roberts (Duke Univ.) for the open source software Osprey and helping us with it.
- Hugo Bazille (ENS/INRIA): for testing ASP on the CP2012 instances.

Questions ?

References I

Carlos Ansótegui, María Luisa Bonet, and Jordi Levy. "Solving (weighted) partial MaxSAT through satisfiability testing". In: *Theory and Applications of Satisfiability Testing-SAT 2009*. Springer, 2009, pp. 427–440.

Carlos Ansótegui, Maria Luisa Bonet, and Jordi Levy. "A New Algorithm for Weighted Partial MaxSAT." In: *Proceedings of 20th National Conference on Artificial Intelligence (AAAI'10)*. 2010.

David Allouche et al. "Computational protein design as an optimization problem". In: *Artificial Intelligence* 212 (2014), pp. 59–79.

Fahiem Bacchus. "GAC via unit propagation". In: *Principles and Practice of Constraint Programming–CP 2007*. Springer, 2007, pp. 133–147.

References II

M C. Cooper, S. de Givry, and T. Schiex. "Optimal soft arc consistency". In: *Proc. of IJCAI'2007*. Hyderabad, India, Jan. 2007, pp. 68–73.

M. Cooper et al. "Soft arc consistency revisited". In: *Artificial Intelligence* 174 (2010), pp. 449–478.

M.C. Cooper. "Fundamental properties of neighbourhood substitution in constraint satisfaction problems". In: *Artificial Intelligence* 90.1-2 (1997), pp. 1–24.

M C. Cooper and T. Schiex. "Arc consistency for soft constraints". In: *Artificial Intelligence* 154.1-2 (2004), pp. 199–227.

Jessica Davies and Fahiem Bacchus. "Exploiting the Power of MIP Solvers in MaxSAT". In: *Theory and Applications of Satisfiability Testing–SAT 2013*. Springer, 2013, pp. 166–181.

References III

J Desmet et al. "The dead-end elimination theorem and its use in protein side-chain positioning." In: *Nature* 356.6369 (Apr. 1992), pp. 539–42. ISSN: 0028-0836. URL: http://www.ncbi.nlm.nih.gov/pubmed/21488406.

Ivelin Georgiev, Ryan H Lilien, and Bruce R Donald. "The minimized dead-end elimination criterion and its application to protein redesign in a hybrid scoring and search algorithm for computing partition functions over molecular ensembles." In: *Journal of computational chemistry* 29.10 (July 2008), pp. 1527–42. ISSN: 1096-987X. DOI: 10.1002/jcc.20909. URL: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3263346%5C&tool=pmcentrez%5C&rendertype=abstract.

References IV

R F Goldstein. "Efficient rotamer elimination applied to protein side-chains and related spin glasses." In: *Biophysical journal* 66.5 (May 1994), pp. 1335–40. ISSN: 0006-3495. DOI: 10.1016/S0006-3495(94)80923-3. URL: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1275854%5C&tool=pmcentrez%5C&rendertype=abstract.

Federico Heras, Javier Larrosa, and Albert Oliveras. "MiniMaxSAT: An Efficient Weighted Max-SAT solver." In: *J. Artif. Intell. Res.(JAIR)* 31 (2008), pp. 1–32.

References V

Federico Heras, Antonio Morgado, and Joao Marques-Silva. "Core-Guided Binary Search Algorithms for Maximum Satisfiability." In: *Proceedings of 21th National Conference on Artificial Intelligence (AAAI'11)*. 2011.

Carleton L Kingsford, Bernard Chazelle, and Mona Singh. "Solving and analyzing side-chain positioning problems using linear and integer programming." In: *Bioinformatics (Oxford, England)* 21.7 (Apr. 2005), pp. 1028–36. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/bti144. URL: http://www.ncbi.nlm.nih.gov/pubmed/15546935.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

References VI

A.M.C.A Koster, S.P.M van Hoesel, and A.W.J. Kolen. *Solving Frequency Assignment Problems via Tree-Decomposition*. Tech. rep. RM/99/011. Maastricht, The Netherlands: Universiteit Maastricht, 1999.

Adrian Kuegel. "Improved exact solver for the weighted Max-SAT problem". In: *Workshop Pragmatics of SAT*. 2010.

J. Larrosa and F. Heras. "Resolution in Max-SAT and its relation to local consistency in weighted CSPs". In: *Proc. of the 19th IJCAI*. Edinburgh, Scotland, 2005, pp. 193–198.

References VII

Christophe Lecoutre, Olivier Roussel, and Djamel E Dehani. "WCSP integration of soft neighborhood substitutability". In: *Principles and Practice of Constraint Programming*. Springer. 2012, pp. 406–421.

Kim Marriott et al. "The design of the Zinc modelling language". In: *Constraints* 13.3 (2008), pp. 229–267.

Rolf Niedermeier and Peter Rossmanith. "New Upper Bounds for Maximum Satisfiability". In: *J. Algorithms* 36.1 (2000), pp. 63–88.

Lars Otten and Rina Dechter. "Anytime AND/OR depth-first search for combinatorial optimization". In: *AI Communications* 25.3 (2012), pp. 211–227.

References VIII

T. Petit, J.C. Régin, and C. Bessière. "Meta constraints on violations for over constrained problems". In: *Proceedings of IEEE ICTAI'2000*. Vancouver, BC, Canada, 2000, pp. 358–365.

Niles A Pierce and Erik Winfree. "Protein design is NP-hard." In: *Protein engineering* 15.10 (Oct. 2002), pp. 779–82. ISSN: 0269-2139. URL: http://www.ncbi.nlm.nih.gov/pubmed/12468711.

Daniel Pruša and Tomáš Werner. "How Hard Is the LP Relaxation of the Potts Min-Sum Labeling Problem?" In: *Energy Minimization Methods in Computer Vision and Pattern Recognition*. Springer. 2015, pp. 57–70.

T. Schiex. "Arc consistency for soft constraints". In: *Principles and Practice of Constraint Programming - CP 2000*. Vol. 1894. LNCS. Singapore, Sept. 2000, pp. 411–424.

References IX

M.I. Schlesinger. "Sintaksicheskiy analiz dvumernykh zritelnikh signalov v usloviyakh pomekh (Syntactic analysis of two-dimensional visual signals in noisy conditions)". In: *Kibernetika* 4 (1976), pp. 113–130.

David Sontag, Do Kook Choe, and Yitao Li. "Efficiently Searching for Frustrated Cycles in MAP Inference". In: *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence (UAI-12)*. Corvallis, Oregon: AUAI Press, 2012, pp. 795–804.

C. Viricel et al. "Approximate Counting with Deterministic Guarantees for Affinity Computations". In: *Proc. of Modeling, Computation and Optimization in Information Systems and Management Sciences - MCO'15*. Metz, France, May 2015.

References X

R. Wallace. "Directed Arc Consistency Preprocessing". In: *Selected papers from the ECAI-94 Workshop on Constraint Processing*. Ed. by M. Meyer. LNCS 923. Berlin: Springer, 1995, pp. 121–137.

T. Werner. "A Linear Programming Approach to Max-sum Problem: A Review." In: *IEEE Trans. on Pattern Recognition and Machine Intelligence* 29.7 (July 2007), pp. 1165–1179. URL: http://dx.doi.org/10.1109/TPAMI.2007.1036.