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

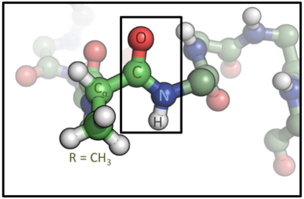
IRISA - INRIA Rennes - September 2015

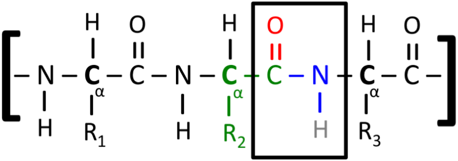
What is a protein ?

(Kudos to wikipedia)

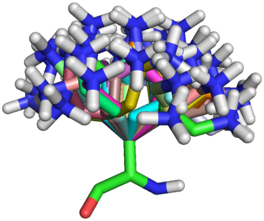
Amino acids, proteins

- 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)



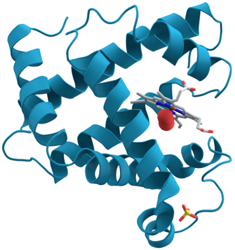
Why ?

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

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

Protein design's aim

Identify sequences that have a suitable function (shape).



Protein Design

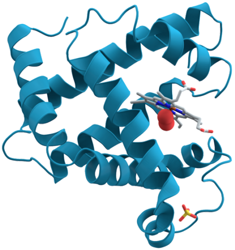
Protein function linked to its 3D shape through its amino acid composition.

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.

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.

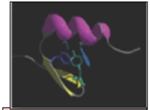
Issues

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

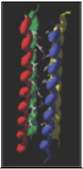
Successes of Protein Design

INRA SCIENCE & IMPACT

1997



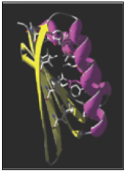
- Zinc-finger
  Dahiyat et al
  (Science 1997)

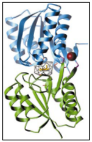


- Coiled-coils
  Harbury et al
  (Science 1998)

2003

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

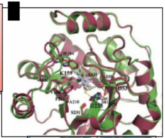




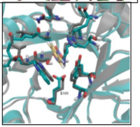
- Sensors & receptors toward non native ligands *Hellinga et al (Nature, 2003)*

2008

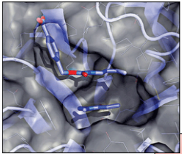
- New catalyst for a multi-step-reaction (retro-aldolase) *Jiang et al (Science 2008)*



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



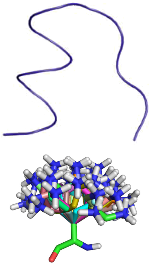
2010



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

Rigid backbone variant

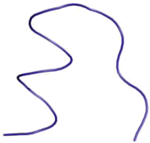
1. Assume a rigid protein backbone.
2. Choose 1 AA among possible ones at each mutable position.
3. Spatial conformation discretized in rotamers.
4. Statistically frequent orientations.
5. Several 100's rotamers per position.

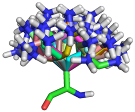


The (basic) CPD problem: search space

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

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)

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 / Sustitutability / Dead End Elimination [Des+92]

$$E(i_a) + \sum_{j \neq i}^{n} \min_{c} E(i_a, j_c) > E(i_b) + \sum_{j \neq i}^{n} E \max_{b} E(i_b, j_c)$$

Dominance / Sustitutability / Dead End Elimination [Des+92]

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

Strenghtened by [Gol94]

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

Dedicated CPD Methods

Dominance / Sustitutability / 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.

Dedicated CPD Methods

Dominance / Sustitutability / 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.

"(Soft) substitutability" [Coo97; LRD12]
Dominating 1-clause rule in MaxSAT [NR00].

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]:

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

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].
- Obsoleted by local consistencies.

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 [All+14]

- Identify a most efficient model/solving technique for the rigid backbone/rotamer based/pairwise energy CPD problem.
- Do one of the first large spectrum comparison of NP-complete optimization techniques (AI: CFN, CP, SAT, MRF and OR: ILP, QP, QPBO) on one well defined, 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. $c_S : D^S = \prod_{D^i, i \in S} \to \{0, \ldots, k\}$

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

Cost Function Networks (aka WCSP or MRF)

Cost Function Network (X, D, E)

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

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

- 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.

Pause pub

1. black box solver (à la SAT/01LP)

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular...)

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular...)
4. (treewidth aware) DFBB and Hybrid BFS [All+15]

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular...)
4. (treewidth aware) DFBB and Hybrid BFS [All+15]
5. Updated Optimality gap (HBFS), anytime behavior

mulcyber.toulouse.inra.fr/projects/toulbar2

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular. . .)
4. (treewidth aware) DFBB and Hybrid BFS [All+15]
5. Updated Optimality gap (HBFS), anytime behavior
6. Default clever horizontal (value) ordering

mulcyber.toulouse.inra.fr/projects/toulbar2

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular...)
4. (treewidth aware) DFBB and Hybrid BFS [All+15]
5. Updated Optimality gap (HBFS), anytime behavior
6. Default clever horizontal (value) ordering
7. Weighted degree + last conflict vertical ordering heuristics

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular...)
4. (treewidth aware) DFBB and Hybrid BFS [All+15]
5. Updated Optimality gap (HBFS), anytime behavior
6. Default clever horizontal (value) ordering
7. Weighted degree + last conflict vertical ordering heuristics
8. Maintains NC, AC, DAC, FDAC, EDAC and VAC.

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular...)
4. (treewidth aware) DFBB and Hybrid BFS [All+15]
5. Updated Optimality gap (HBFS), anytime behavior
6. Default clever horizontal (value) ordering
7. Weighted degree + last conflict vertical ordering heuristics
8. Maintains NC, AC, DAC, FDAC, EDAC and VAC.
9. Maintains non-dominance (aka substitutability aka DEE)

mulcyber.toulouse.inra.fr/projects/toulbar2

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular...)
4. (treewidth aware) DFBB and Hybrid BFS [All+15]
5. Updated Optimality gap (HBFS), anytime behavior
6. Default clever horizontal (value) ordering
7. Weighted degree + last conflict vertical ordering heuristics
8. Maintains NC, AC, DAC, FDAC, EDAC and VAC.
9. Maintains non-dominance (aka substitutability aka DEE)
10. (On the fly) Variable elimination (degree ≤ 3)

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular...)
4. (treewidth aware) DFBB and Hybrid BFS [All+15]
5. Updated Optimality gap (HBFS), anytime behavior
6. Default clever horizontal (value) ordering
7. Weighted degree + last conflict vertical ordering heuristics
8. Maintains NC, AC, DAC, FDAC, EDAC and VAC.
9. Maintains non-dominance (aka substitutability aka DEE)
10. (On the fly) Variable elimination (degree ≤ 3)
11. Local search upper bounding (INCOP [NT03])

mulcyber.toulouse.inra.fr/projects/toulbar2

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular...)
4. (treewidth aware) DFBB and Hybrid BFS [All+15]
5. Updated Optimality gap (HBFS), anytime behavior
6. Default clever horizontal (value) ordering
7. Weighted degree + last conflict vertical ordering heuristics
8. Maintains NC, AC, DAC, FDAC, EDAC and VAC.
9. Maintains non-dominance (aka substitutability aka DEE)
10. (On the fly) Variable elimination (degree ≤ 3)
11. Local search upper bounding (INCOP [NT03])
12. Table cost function decomposition

1. black box solver (à la SAT/01LP)
2. table cost functions (tables, lists)
3. global cost functions (Weighted AllDiff, GCC, Regular...)
4. (treewidth aware) DFBB and Hybrid BFS [All+15]
5. Updated Optimality gap (HBFS), anytime behavior
6. Default clever horizontal (value) ordering
7. Weighted degree + last conflict vertical ordering heuristics
8. Maintains NC, AC, DAC, FDAC, EDAC and VAC.
9. Maintains non-dominance (aka substitutability aka DEE)
10. (On the fly) Variable elimination (degree ≤ 3)
11. Local search upper bounding (INCOP [NT03])
12. Table cost function decomposition
13. Parallel VNS search [Oua+14]

Past successes...

1. First/second in approximate graphical model MRF/MAP challenges (2010, 2012, 2014).
2. Bioinformatics: pedigree debugging [SGS08], Haplotyping (QTLMap), structured RNA gene finding [ZGS08], Computational Protein Design [Tra+13] (now in OSPREY)
3. RLFAP: closed all CELAR min-interference RLFAP instances fap.zib.de/problems/CALMA
4. Inductive Logic Programming [AR07], Natural Langage Processing (in hltdi-l3), Multi-agent and cost-based planning [KZ10; CRR11], Model Abstraction [SFN11], diagnostic [MJS11b], Music processing and Markov Logic [PT12; PT13], Data mining [MLC13], Partially observable Markov Decision Processes [Dib+13], Probabilistic counting [Erm+13] and inference [MJS11a], ...

Arc EPT

- A cost function c_S, here c_{ij}.
- EPT $\texttt{Project}$ $(\{ij\},\{i\},a,\alpha)$ 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 $\texttt{Project}(\{i,j\},\{i\},a,\alpha)$

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

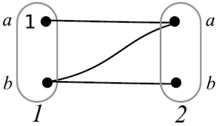
  foreach *($t' \in D^{ij}$ such that $t'[i] = i_a$)* **do**

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

  end

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

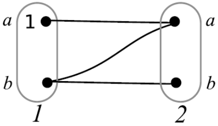
Example

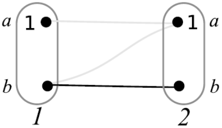


Example

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

$\rightarrow$

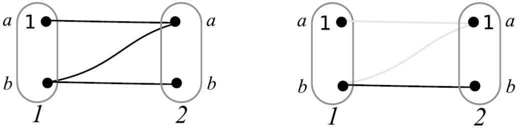




Example

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

$\rightarrow$



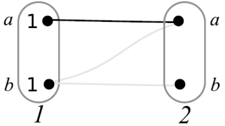
$\leftarrow$

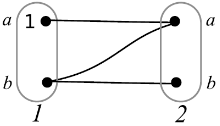
$\texttt{Project}(\{1,2\},\{2\},a,-1)$

Example

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

$\leftarrow$

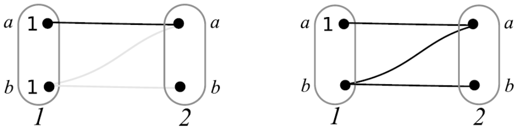




Example

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

$\leftarrow$



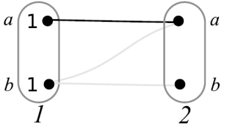
$\rightarrow$

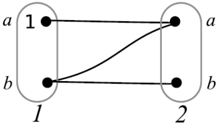
$\texttt{Project}(\{1,2\},\{1\},b,-1)$

Example

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

$\leftarrow$

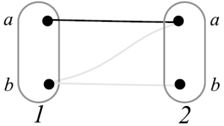




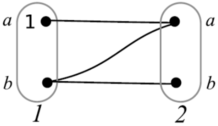
$\Downarrow \qquad \texttt{Project}\left(\{1\},\varnothing,[],1\right)$

Example

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



$\leftarrow$



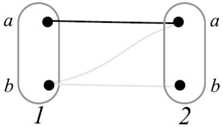
$\Downarrow \qquad \texttt{Project}\,(\{1\},\varnothing,[],1)$

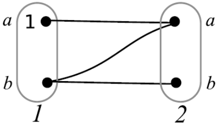
$c_{\varnothing} = 1$

Example

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

$\leftarrow$





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

$c_\varnothing = 1$

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

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.

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

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 $c_\varnothing$
- $p^S_{i_a}$: amount of cost projected from c_S to i_a

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

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

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).

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

Relaxation = dual of OSAC LP

1. Arc consistencies: limited Block Coordinate Descent algorithms for the dual of this specific LP.

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).

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

Relaxation = dual of OSAC LP

1. Arc consistencies: limited Block Coordinate Descent algorithms for the dual of this specific LP.
2. Not so specific: any LP can be reduced to it in linear time [PW15].

As quadratic 0/1 programs

QP - Cplex

$$\begin{aligned}
\min \sum_{i,r} E(i_r).d_{ir} + \sum_{\substack{i,r,j,s \\ j>i}} E(i_r, j_s).d_{ir}.d_{js} \\
\text{s.t.} \quad \sum_{r} d_{ir} = 1 \quad (\forall i) \\
d_{ir} \in \{0, 1\} \quad (\forall i, r)
\end{aligned}$$

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 [Son+12]

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]

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 $(\neg d_{i_r} \vee \neg d_{j_s}, E(i_r, j_s))$

Property [Bac07]

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

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.

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]

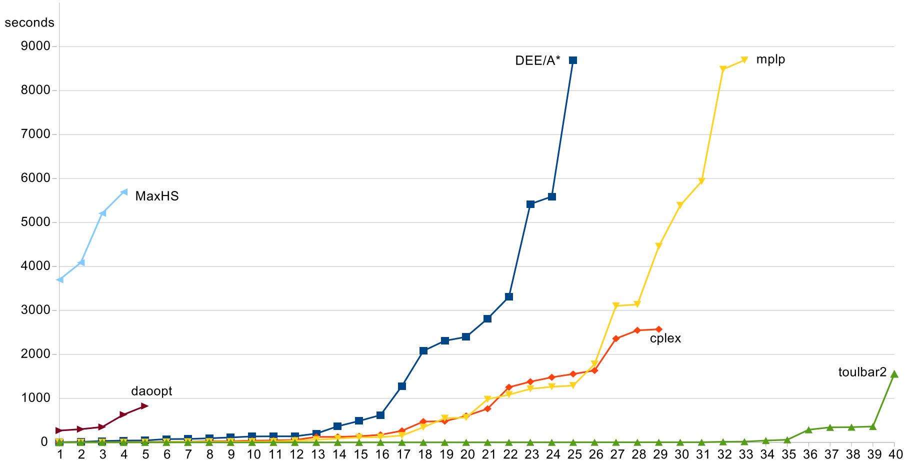
1. GeCode (`http://www.gecode.org/`),
2. Mistral (Python `numberjack` interface, `http://numberjack.ucc.ie/`),
3. Opturion/CPX `http://www.opturion.com/cpx.html`

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}
7. Largest solved has size 10^{98}

Results - 9000 seconds



Analysis

1. **QP by Cplex**: dense model, but weak and somewhat expensive lb (very large node file, large gaps).

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.

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%).

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).

Analysis

1. **Daoopt**: almost complete graphs. Not ideal for tree decomposition based methods.

… 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.

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).

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.

... 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, AFAiK, 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 ?

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.

David Allouche et al. "Anytime Hybrid Best-First Search with Tree Decomposition for Weighted CSP". In: *Principles and Practice of Constraint Programming*. Springer. 2015, pp. 12–29.

Érick Alphonse and Céline Rouveirol. "Extension of the top-down data-driven strategy to ILP". In: *Inductive Logic Programming*. Springer, 2007, pp. 49–63.

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

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.

Martin C Cooper, Marie de Roquemaurel, and Pierre Régnier. "A weighted CSP approach to cost-optimal planning". In: *Ai Communications* 24.1 (2011), pp. 1–29.

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.

Jilles Steeve Dibangoye et al. "Optimally solving Dec-POMDPs as continuous-state MDPs". In: *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*. AAAI Press. 2013, pp. 90–96.

Stefano Ermon et al. "Embed and project: Discrete sampling with universal hashing". In: *Advances in Neural Information Processing Systems*. 2013, pp. 2085–2093.

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.

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.

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.

References V

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

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.

Akshat Kumar and Shlomo Zilberstein. "Point-based backup for decentralized POMDPs: Complexity and new algorithms". In: *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems. 2010, pp. 1315–1322.

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

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.

Paul Maier, Dominik Jain, and Martin Sachenbacher. “Compiling AI engineering models for probabilistic inference”. In: *KI 2011: Advances in Artificial Intelligence*. Springer, 2011, pp. 191–203.

Paul Maier, Dominik Jain, and Martin Sachenbacher. “Diagnostic hypothesis enumeration vs. probabilistic inference for hierarchical automata models”. In: *the International Workshop on Principles of Diagnosis (DX), Murnau, Germany*. 2011.

Jean-Philippe Métivier, Samir Loudni, and Thierry Charnois. “A constraint programming approach for mining sequential patterns in a sequence database”. In: *Proceedings of the ECML/PKDD Workshop on Languages for Data Mining and Machine Learning*. arXiv preprint arXiv:1311.6907. Praha, Czech republic, 2013.

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

Bertrand Neveu and Gilles Trombettoni. “Incop: An open library for incomplete combinatorial optimization”. In: *Principles and Practice of Constraint Programming–CP 2003*. Springer. 2003, pp. 909–913.

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

Abdelkader Ouali et al. "Cooperative parallel decomposition guided VNS for solving weighted CSP". In: *Hybrid Metaheuristics*. Springer, 2014, pp. 100–114.

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.

Hélène Papadopoulos and George Tzanetakis. "Modeling Chord and Key Structure with Markov Logic." In: *Proc. Int. Conf. of the Society for Music Information Retrieval (ISMIR)*. 2012, pp. 121–126.

Hélene Papadopoulos and George Tzanetakis. "Exploiting structural relationships in audio music signals using Markov Logic Networks". In: *ICASSP 2013-38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Canada (2013)*. 2013, pp. 4493–4497.

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 Prusa and Tomas Werner. “Universality of the local marginal polytope”. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 37.4 (2015), pp. 898–904.

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.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.

Peter Struss, Alessandro Fraracci, and D Nyga. “An Automated Model Abstraction Operator Implemented in the Multiple Modeling Environment MOM”. In: *25th International Workshop on Qualitative Reasoning, Barcelona, Spain*. 2011.

Martí Sánchez, Simon de Givry, and Thomas Schiex. “Mendelian Error Detection in Complex Pedigrees Using Weighted Constraint Satisfaction Techniques”. In: *Constraints* 13.1-2 (2008), pp. 130–154.

David Sontag et al. “Tightening LP relaxations for MAP using message passing”. In: *arXiv preprint arXiv:1206.3288* (2012).

References IX

Seydou Traoré et al. "A new framework for computational protein design through cost function network optimizatio n". In: *Bioinformatics* 29.17 (2013), pp. 2129–2136.

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.

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.

Matthias Zytnicki, Christine Gaspin, and Thomas Schiex. "DARN! A weighted constraint solver for RNA motif localization". In: *Constraints* 13.1-2 (2008), pp. 91–109.