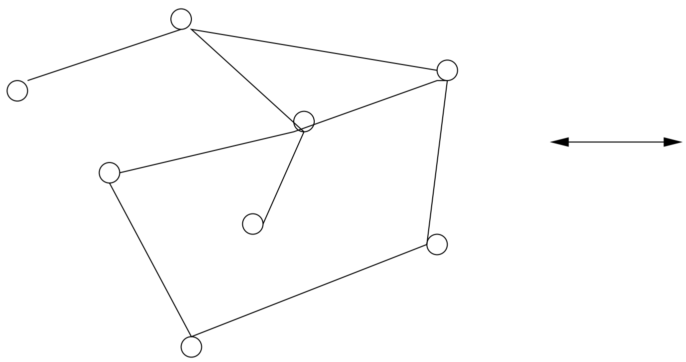
Protein Landscape Analysis in the Grand Canonical Model

James Aspnes, Julia Hartling, Ming-Yang Kao, Junhyong Kim, Gauri Shah

Outline

- Introduction
- Grand Canonical Model
- Tools
  - Linear Programming
  - Network Flow
  - A Compact Representation of all Minimum Cuts
- Algorithmic Results
- Computational Hardness Results
- Experimental Results

Introduction



HHPPHPPH

Basic Question : Find a *fittest sequence* given a target protein structure

- Heuristic to search space of sequences
  Sun et al 1995

- Computational Tractability : open
  Hart 1997

- Polynomial time solution
  Kleinberg 1999

Grand Canonical Model (Sun et al)

Given : Target Structure with n residues

Design : Sequence x

x_i = Hydrophobic(H) or Polar (P)

Fitness Function :

$$\phi(S) = \alpha \sum_{\substack{i<j-2 \\ i,j \in S_H}} g(d_{ij}) + \beta \sum_{i \in S_H} s_i$$

s_i : Solvent-accessible surface of residue i ($\mathring{A}$)
d_{ij} : Distance between residues i and j ($\mathring{A}$)

- Low solvent-accessible area (s_i)
- Compact hydrophobic core
- $\alpha < 0$ and $\beta > 0$
- g $= \begin{cases} \frac{1}{1+\exp(d_{ij}-6.5)} & \text{when } d_{ij} \leq 6.5 \\ 0 & \text{when } d_{ij} > 6.5 \end{cases}$

1. Linear Programming

i-th residue : 0-1 variable x_i

$$\Phi(S) = \alpha \sum_{\substack{i<j-2 \\ i,j \in S_H}} g(d_{ij}) + \beta \sum_{i \in S_H} s_i$$

$$\Phi(x) = - \sum_{\substack{i<j-2 \\ i,j}} a_{ij} x_i x_j + \sum_i b_i x_i$$

Let Δ = # of terms in the fitness function

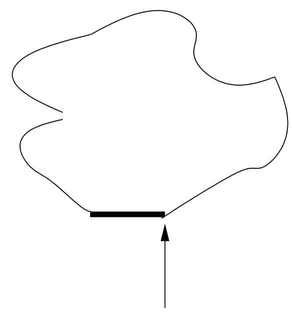
minimize
$g(x,y) = -\sum a_{ij} y_{ij} + \sum b_i x_i$
subject to

$$\begin{array}{ll} 0 \le x_i \le 1 & \forall i \\ \left.\begin{array}{r} 0 \le y_{ij} \le 1 \\ y_{ij} \le x_i \\ y_{ij} \le x_j \end{array}\right\} & \forall i,j : a_{ij} \neq 0 \end{array}$$

- Linear (not quadratic)
- $y_{ij} = \min(x_i, x_j) = x_i x_j$
- Unimodular matrix $\rightarrow$ Integral solutions

Theorem 1 *$\widehat{x}$: 0-1 vector. Can find fittest x with Minimum Weighted Hamming Distance $\sum_i w_i|x_i - \widehat{x}_i|$*

Proof:



$$0 < \epsilon < \frac{1}{Wd(n+1)};\ W \geq \max|w_i|$$

Minimize $f_\epsilon(x) = f(x) + \sum_i \epsilon w_i\, |x_i - \widehat{x}_i|$

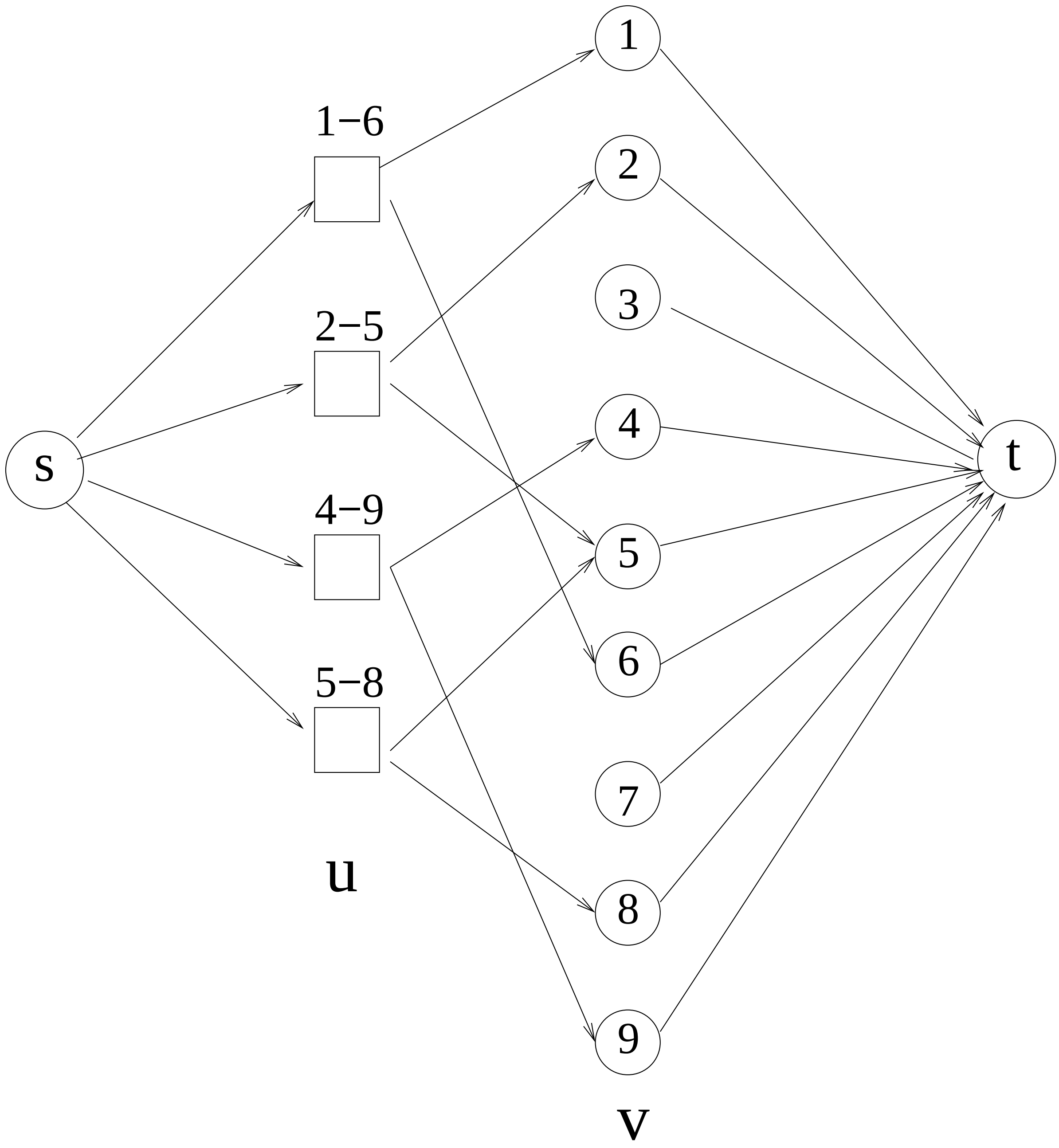
Suppose x minimizes $f(x)$.

$$\begin{aligned} f_\epsilon(x) &\leq f(x) + \frac{Wn}{Wd(n+1)} \\ &< f(x) + \frac{1}{d} \\ &\leq f(x') \\ &\leq f_\epsilon(x') \end{aligned}$$

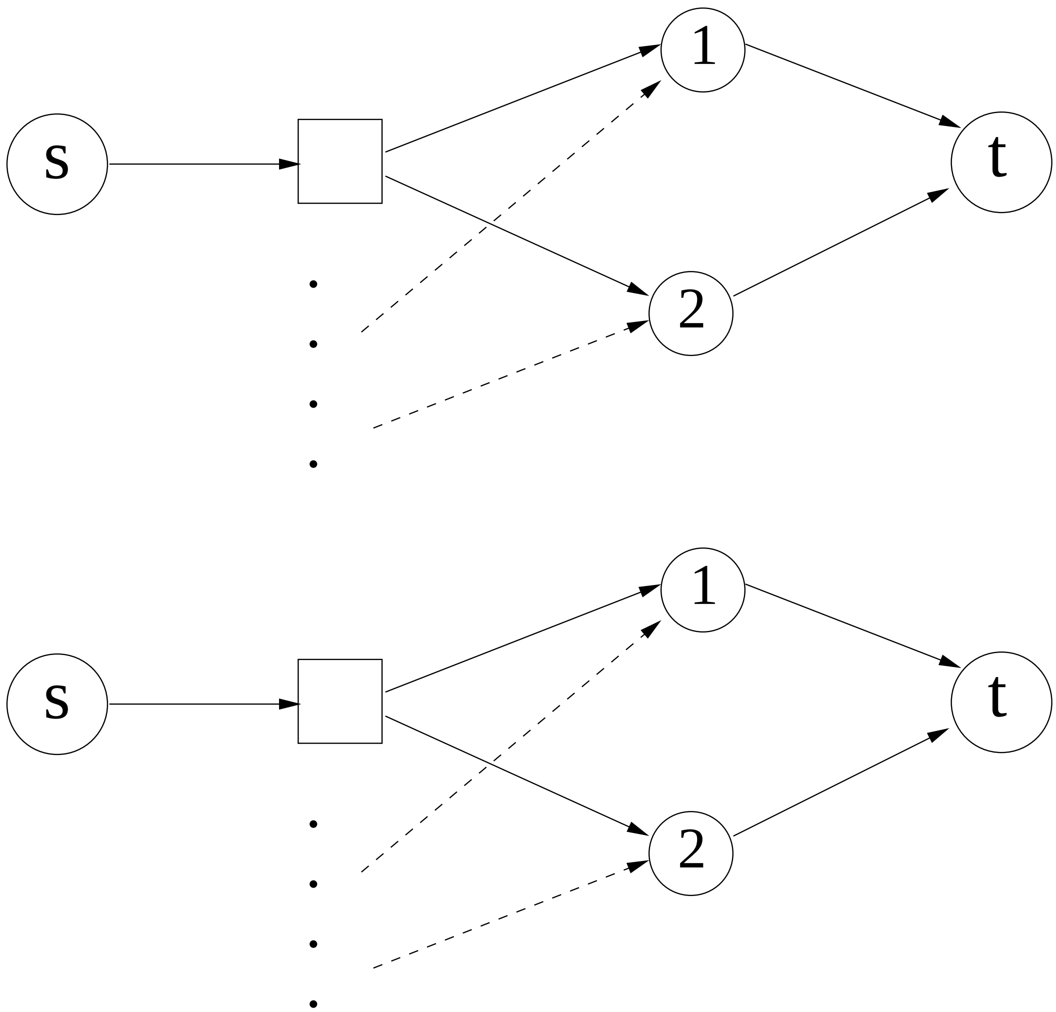
x minimizes $f_\epsilon(x) \Rightarrow x$ minimizes $f(x)$

Running Time : $O(\Delta^2 \log \Delta)$ ∎

2. Network Flow (Kleinberg 99)



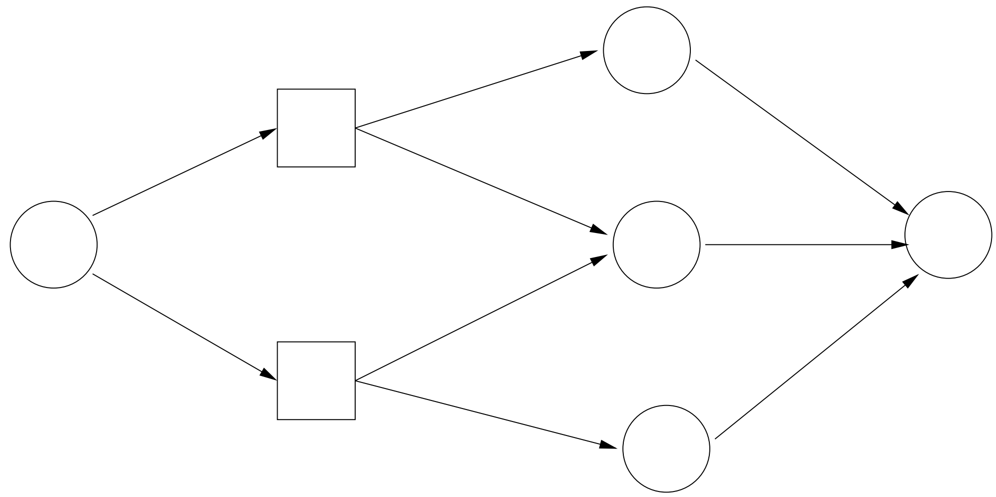
Capacity of Min-Cut = $\Phi(S)$ + B

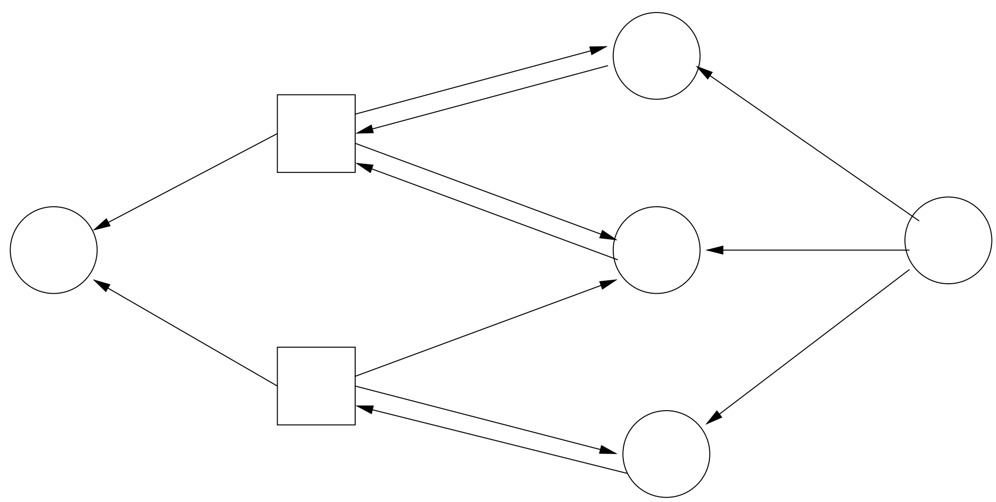


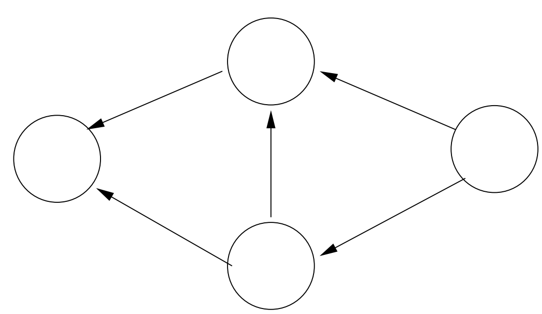
$$\Phi(S) = \alpha \sum_{i<j-2} g(d_{ij}) + \beta \sum_{i \in S_H} s_i$$

Goldberg-Tarjan Min Cut : $O(VE \log(V^2/E))$

Compact Min Cut Representation







Picard-Queyranne 1980

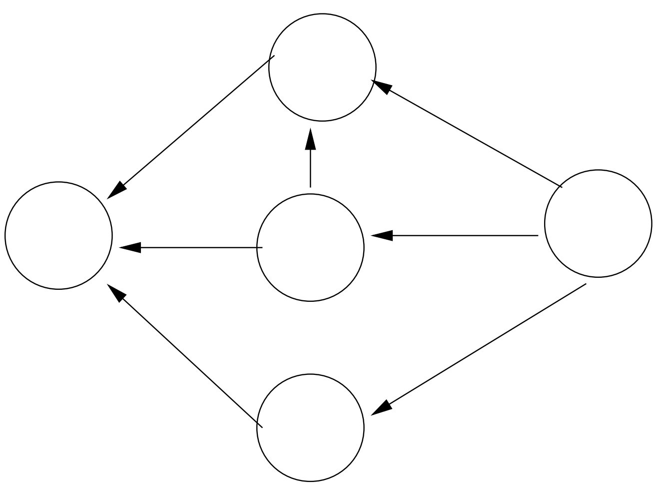
1-1 correspondence : Min Cuts and Ideals

Space of all Fittest Sequences

Theorem 2 *Can enumerate all* 0-1 *f-minimizing vectors x in $O(n)$ time per vector.*

Proof: Steiner 1986 : Enumerates the ideals in $O(n)$ time per ideal $\blacksquare$

Theorem 3 *Can find diameter (k) in Hamming distance of the set of* 0-1 *vectors x minimizing $f(x)$ in $O(n)$ time.*

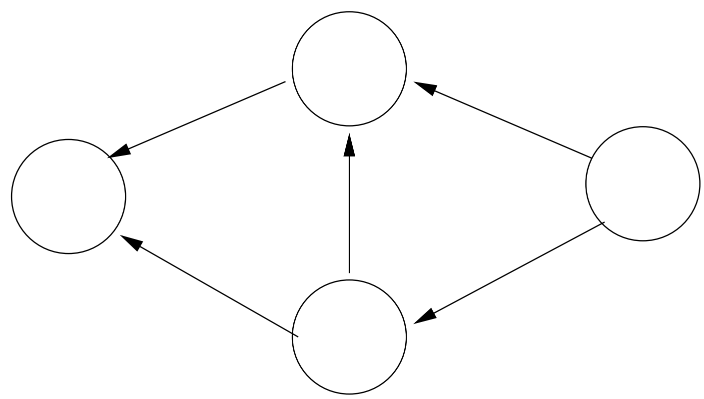


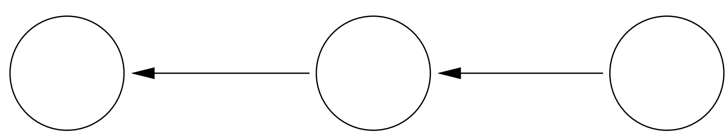
HHH	HHHH	PP
	⋮	
HHH	HHPP	PP
HHH	HHHP	PP
	⋮	
HHH	PPPP	PP

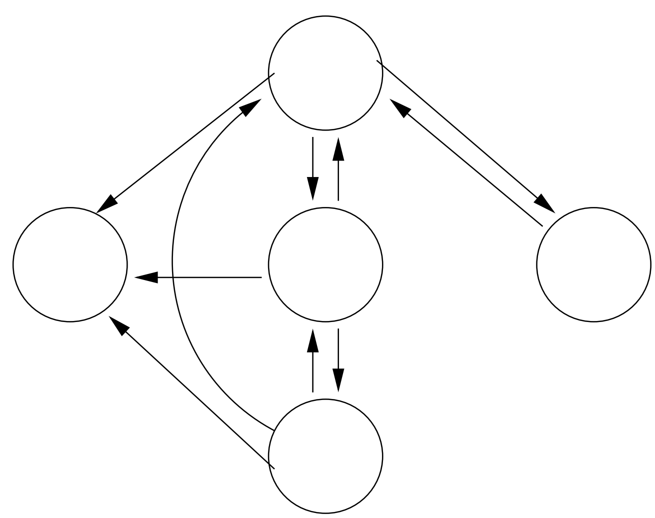
$$k = |\rho^{-1}(V(\widehat{G}^{f}_{s,t}))| = 4$$

Fittest Sequence for k functions

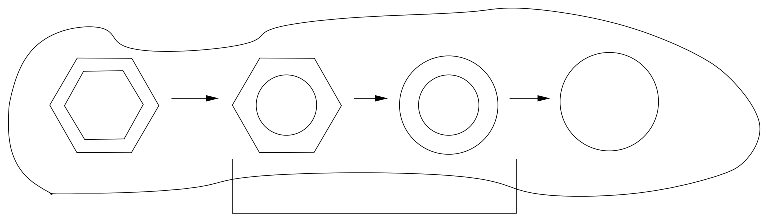
Theorem 4 *Given k fitness functions, determine if no simultaneous solution exists, or construct a graph which represents all possible solutions.*







Adjacency of fittest sequences



Λ = Set of allowable mutations

Downward closed system of subsets of $\{1, \ldots, n\}$

e.g. $\{\{1, 3, 5\}, \{1\}, \{3\}, \{5\}, \{1, 3\}, \{3, 5\}, \{1, 5\}\}$

HPPPHHPPHP

HPHPPHPPHP

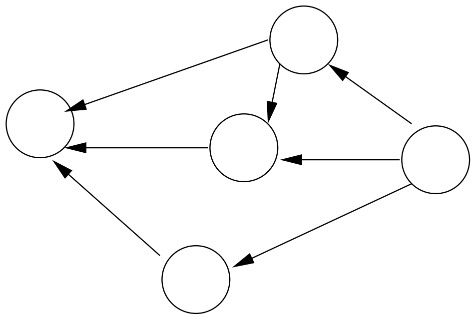
PPHPPHPPHP

RESIDUES $\rightleftarrows$ PQ GRAPH

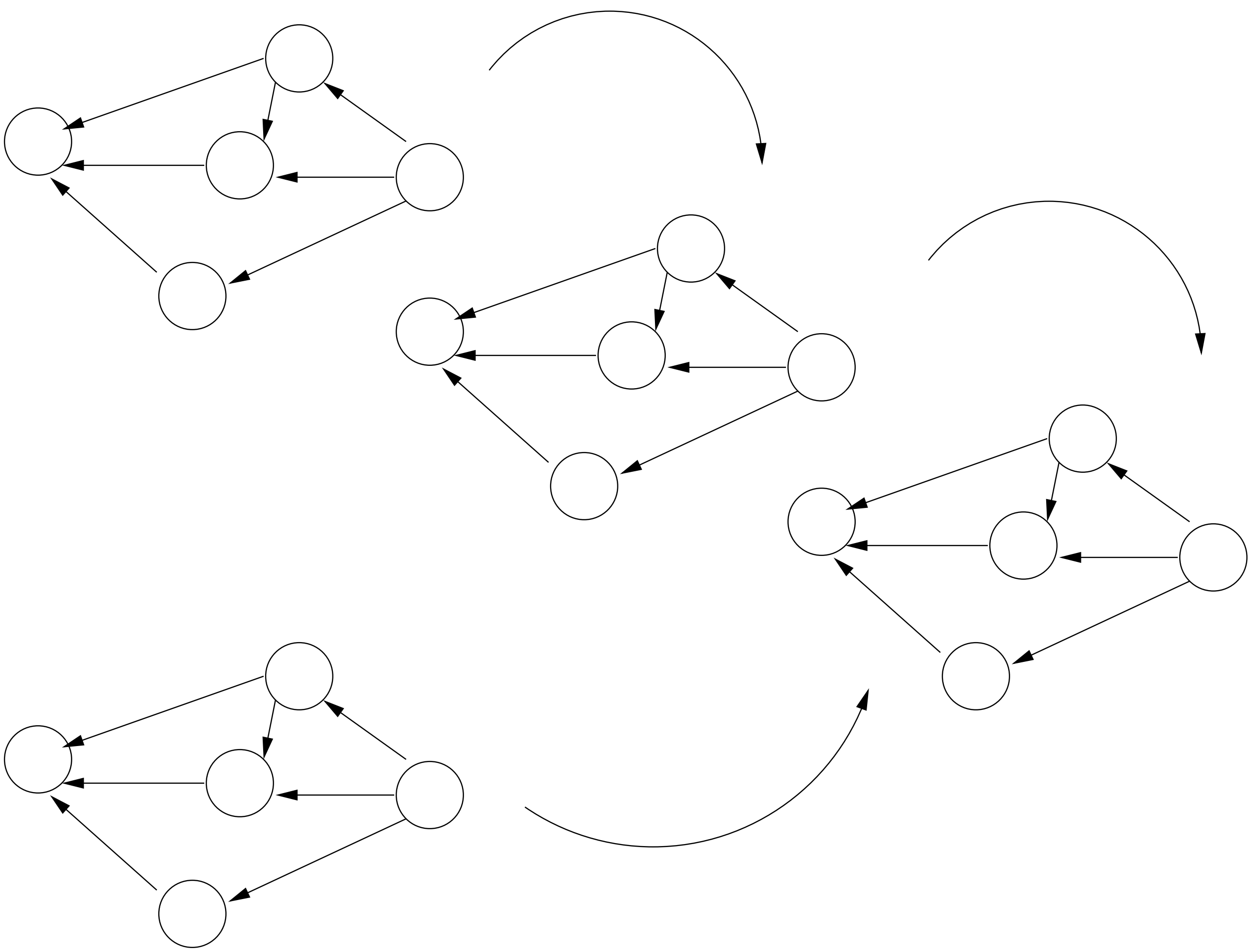
Smallest $\Lambda = \rho^{-1}(I \triangle I')$

I : Ideal of x
I' : Ideal of x'

Necessary :

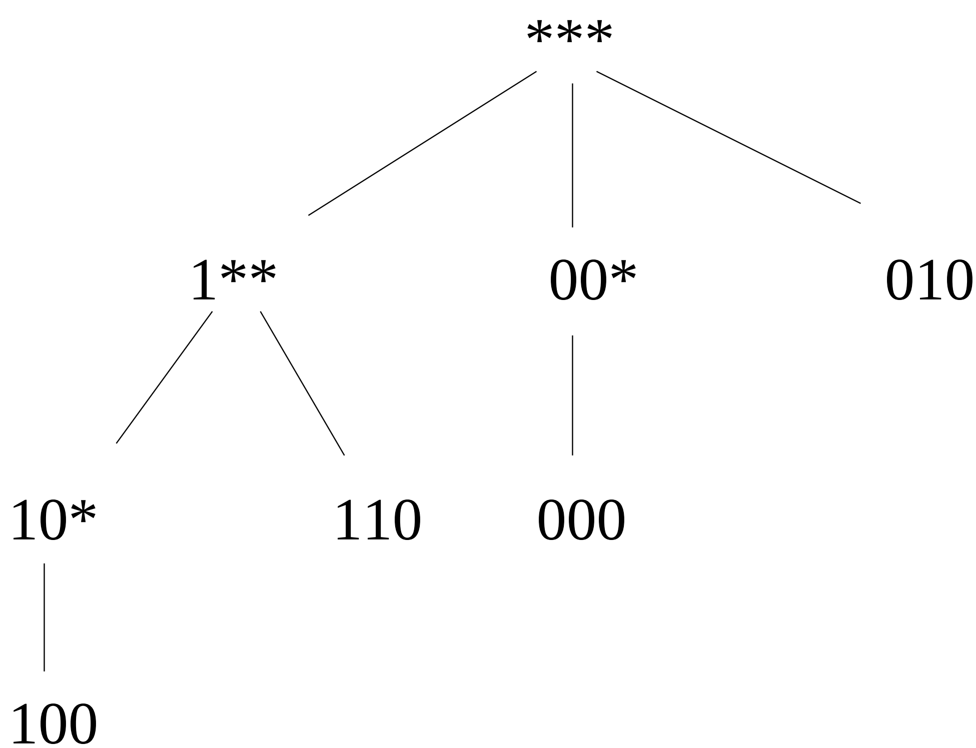


Sufficient :



Enumerating sub-optimal sequences

Lawler's method (1972)
Best-First Search

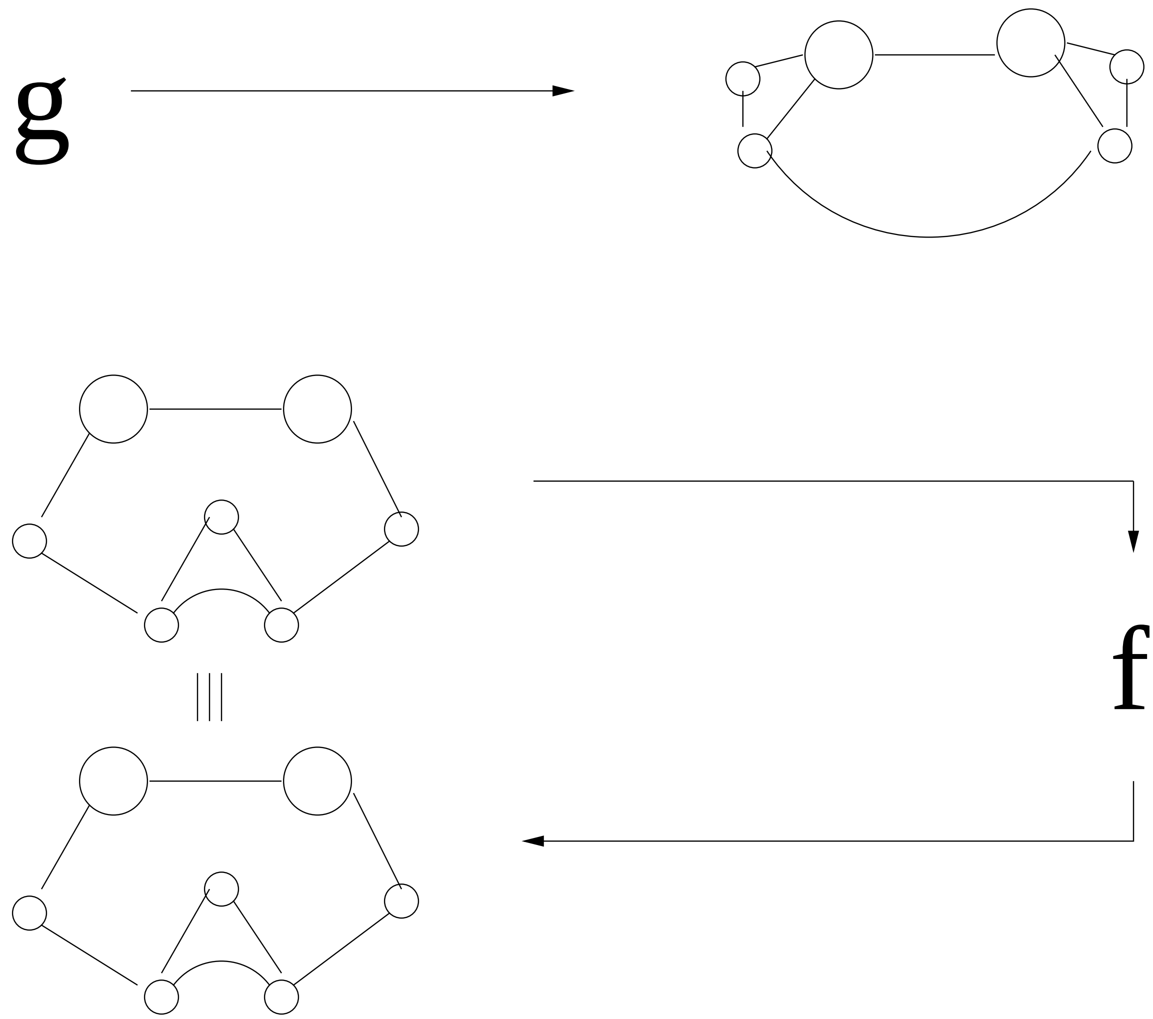


At each step :

- Pick f-minimizing vector
- Remove and replace by n pairs
- Update priority queue

Total cost : $O(n\Delta^2 log\Delta)$ per value

Structure of space of all fittest seq



Computational Hardness Results

Theorem 5 Ω_f *= set of all* x *that minimize* $f(x)$*. #P-complete problems:*

1. *Computing the size of* Ω_f*.*

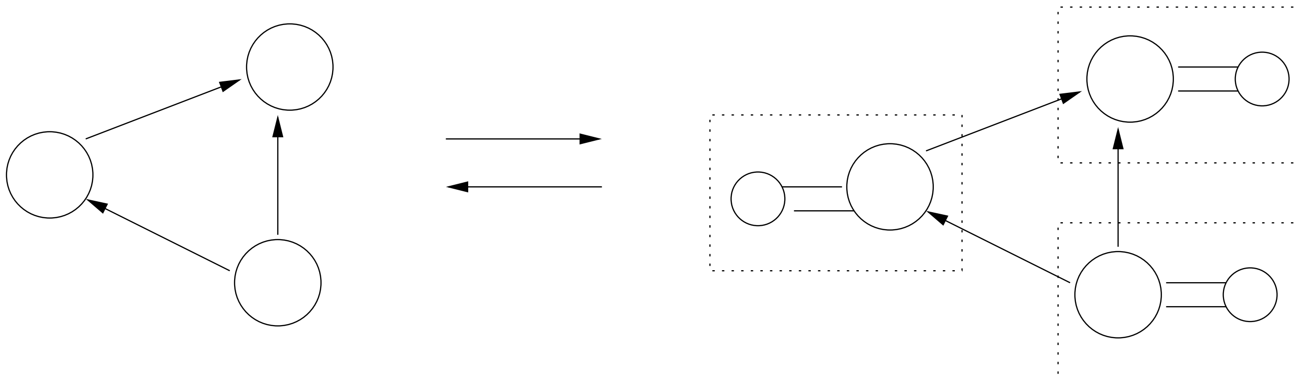
2. *Counting the number of vectors* x *that simultaneously minimize* $f^\ell(x)$ *for all* ℓ *(for any* ℓ*).*

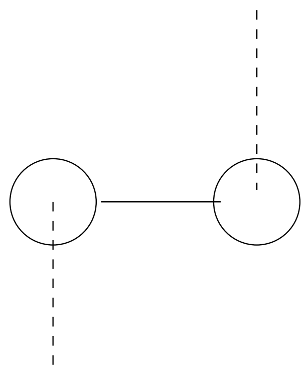
3. *Computing the average weight* $|x|$ *of elements of* Ω_f*.*

4. *Computing the average Hamming distance* $|x - \hat{x}|$ *of elements of* Ω_f *from a given target* $\hat{x}$*.*

5. *Computing the number of elements of* Ω_f *at a given Hamming distance* k *from a given target* $\hat{x}$*.*

Proof:





(x_{v_i}, x_{w_i}) contributes 1 to the HD

$\therefore |\Omega_f| = \#$ of f'-minimizing vectors at HD
k from $\widehat{x}$

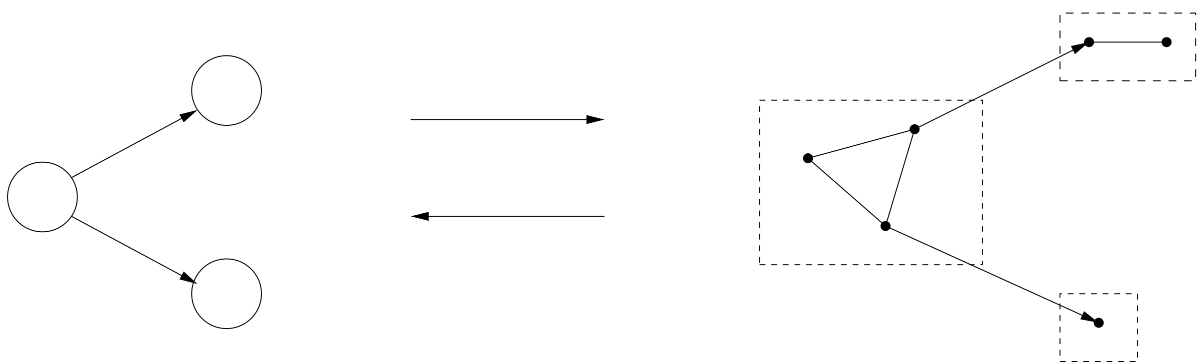
where k $= |\widehat{G}^f_{s,t}| = 3$ ∎

Theorem 6 *NP-complete to find a fittest seq x such that* $k \leq |x| \leq l$

Proof:

Partially Ordered Knapsack:
$s(u) = v(u) = \{3, 1, 2\}$

$$\sum_{u \in ideal\ I} s(u) \leq B \qquad\qquad \sum_{u \in ideal\ I} v(u) \geq K$$



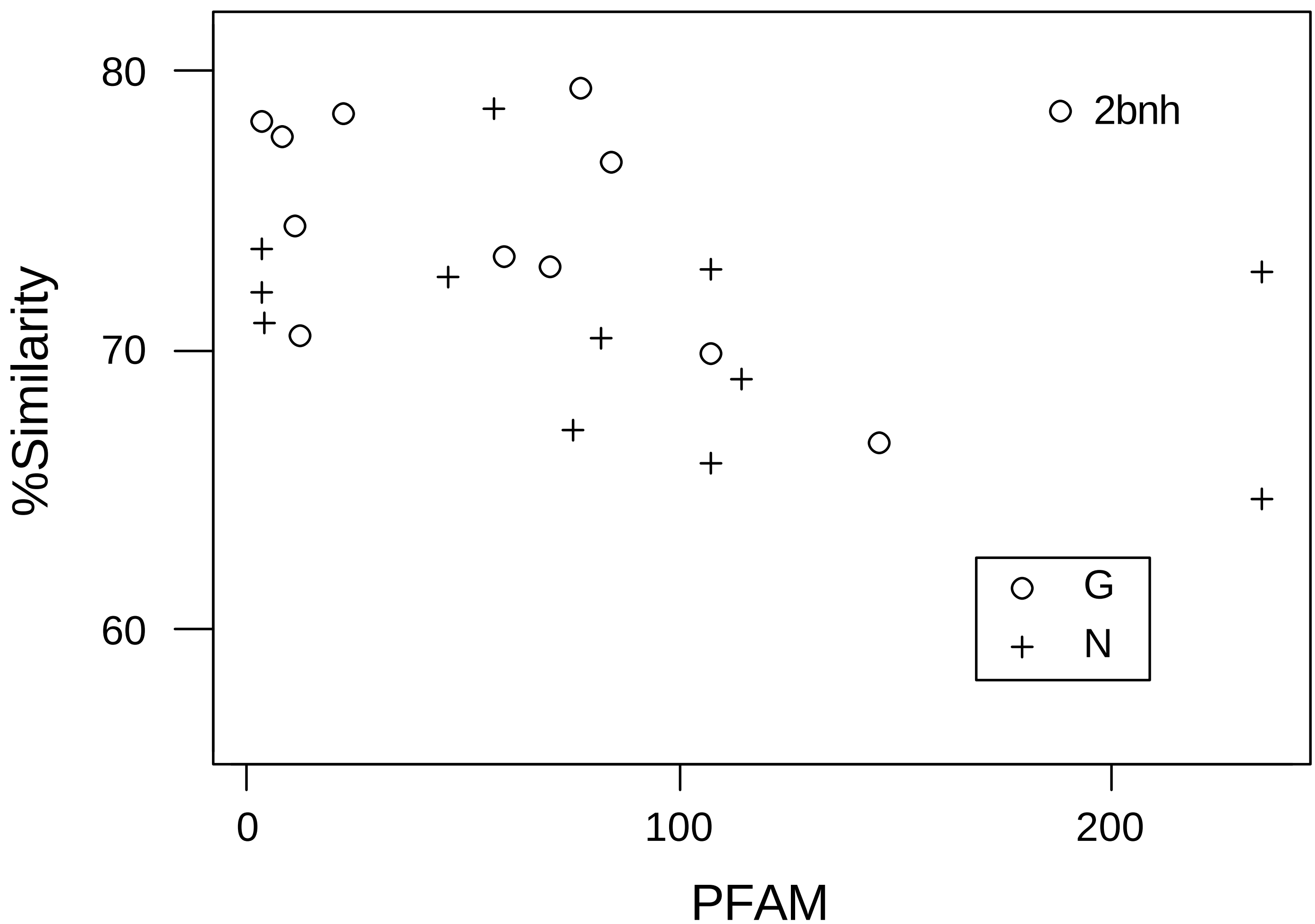
$$\widehat{G}^{f}_{s,t} \cong G_0 \cong U$$

An ideal of $\widehat{G}^{f}_{s,t}$ corresponds to an ideal of U

$$|x| = \sum_{u \in I} |C(u)| = \sum_{u \in I} s(u)$$

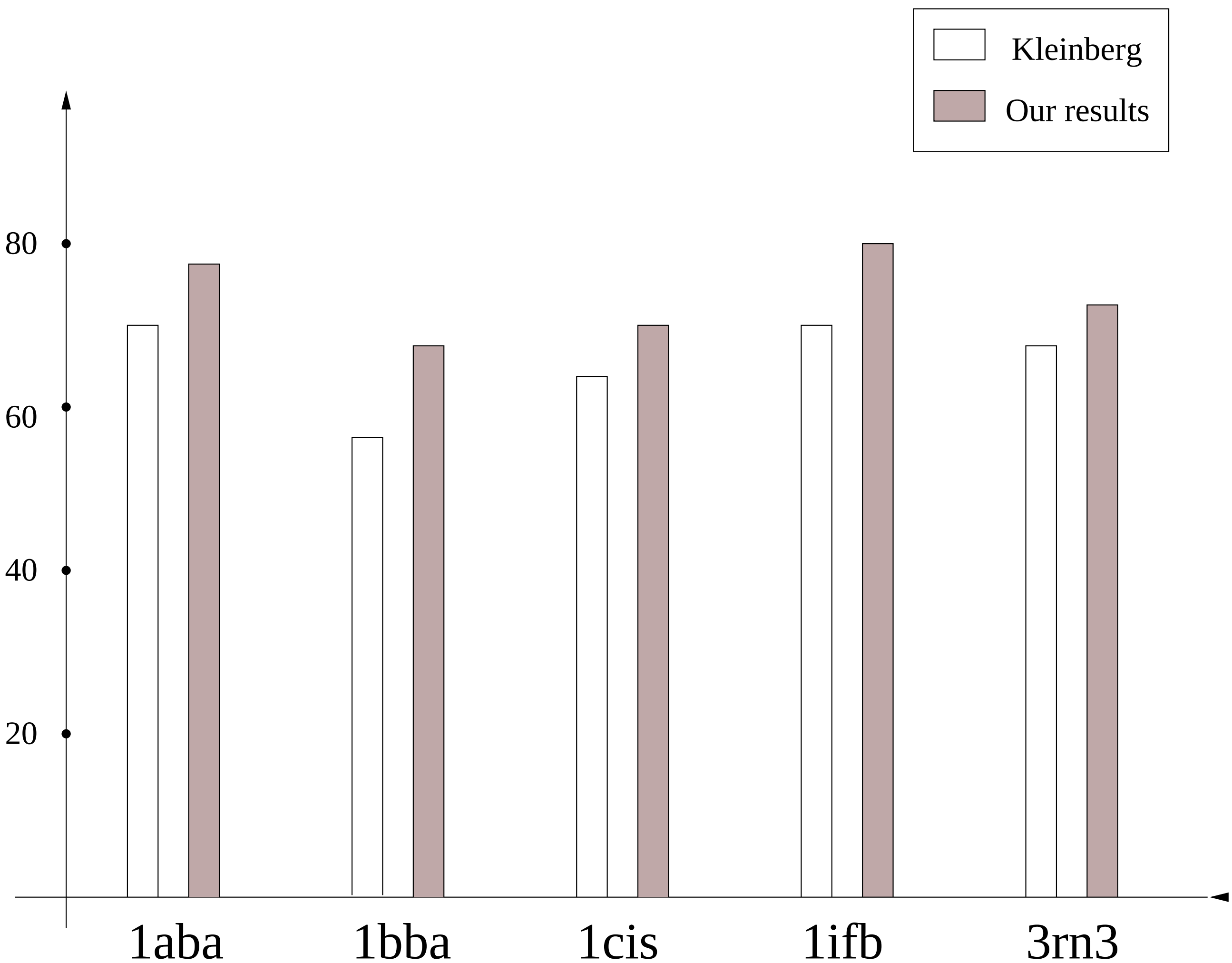
Set $k = K$ and $l = B$. ∎

Applications to Empirical Structures



Relationship between % similarity to native proteins vs. PFAM family size.

Similarity between natural and designed sequences



Open Questions

- Use another folding code (bigger alphabet)

- Improve quality of Fitness Function

Apply model to all PDB structures

Summary of Results

- Enumerate all f-minimizing vectors in $O(n)$ time per vector

- Find diameter in Hamming Distance of all f-minimizing vectors in $O(n)$ time

- Find fittest sequence for k functions or determine if it does not exist

- Compute smallest set of mutations required for adjacency of sequences

- Enumerate sub-optimal sequences by Lawler's method in $O(n\Delta^2 \log \Delta)$ time per sequence