AGH University of Science and Technology

Parallel approach for visual clustering of protein databases

Patryk Orzechowski[1], Krzysztof Boryczko[2]

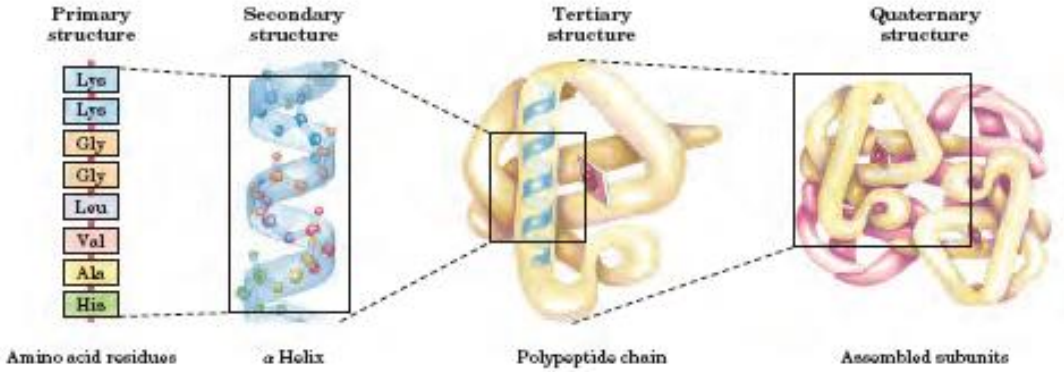
[1] Institute of Automatics, Kraków, Poland, patrick@agh.edu.pl
[2] Institute of Computer Science, Kraków, Poland, bory@agh.edu.pl

Agenda

1. Introduction
2. Methodology
    a. Proteins distance calculation
    b. Clustering
    c. Multidimensional Scaling
    d. Quality assessment
3. Tools
4. Results
5. Conclusions

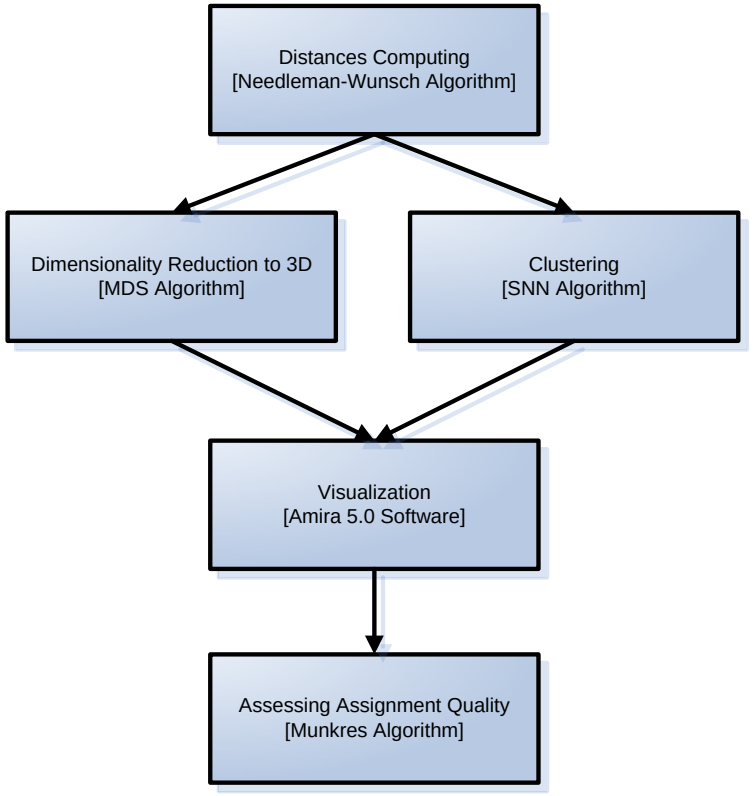
Introduction

- conventional classification bases on Hidden Markov Model (HMM) and Multiple Sequence Alignment (MSA)
- alternative aproach with clustering algorithms used
- no attempt taken so far to visually represent whole dataset basing on structural similarity of proteins



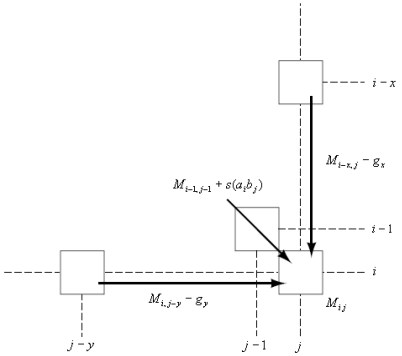
D.L. Nelson, M.M. Cox „Lehninger Principles of Biochemistry”

Approach


Distances Computing
[Needleman-Wunsch Algorithm]
Dimensionality Reduction to 3D
[MDS Algorithm]
Clustering
[SNN Algorithm]
Visualization
[Amira 5.0 Software]
Assessing Assignment Quality
[Munkres Algorithm]

Methodology – proteins distances

- Needleman-Wunsch algorithm



$$M_{ij} = \max\{ M_{i-1,\,j-1} + s(a_i b_j);\ \max_{x \geq 1}(M_{i-x,\,j} - g_x);\ \max_{y \geq 1}(M_{i,\,j-y} - g_y)\ \}$$

	–	J	G	A	H	D	E	K	J	F	I
–	0	–1	–2	–3	–4	–5	–6	–7	–8	–9	–10
A	–1	-2	-3	3	2	1	0	–1	–2	–3	–4
B	–2	-3	-4	2	1	0	–1	–2	–3	–4	–5
J	–3	3	2	1	0	–1	–2	–3	3	2	1
D	–4	2	1	0	–1	5	4	3	2	1	0
E	–5	1	0	–1	–2	4	10	9	8	7	6
J	–6	0	–1	–2	–3	3	9	8	14	13	12
K	–7	–1	–2	–3	–4	2	8	14	13	12	11
D	–8	–2	–3	–4	–5	1	7	13	12	11	10
J	–9	–3	–4	–5	–6	0	6	12	18	17	16
E	–10	–4	–5	–6	–7	–1	5	11	17	16	15
J	–11	–5	–6	–7	–8	–2	4	10	16	15	**14**

M_{ij} – result of aligning first i characters in sequence a against first j in sequence b,
$s(a_i b_j)$ – similarity between i-th character in sequence a and j-th in b,
g_x – gap penalty for sequence a,
g_y – gap penalty for sequence b.

```
1: __JGAHDE_K_J__FI
+: ABJGAHDEJKDJEJFI
2: ABJ___DEJKDJEJ__
```

Methodology – proteins distances

- Similarity Matrices

PAM-250

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	2																			
R	-2	6																		
N	0	0	2																	
D	0	-1	2	4																
C	-2	-4	-4	-5	12															
Q	0	1	1	2	-5	4														
E	0	-1	1	3	-5	2	4													
G	1	-3	0	1	-3	-1	0	5												
H	-1	2	2	1	-3	3	1	-2	6											
I	-1	-2	-2	-2	-2	-2	-2	-3	-2	5										
L	-2	-3	-3	-4	-6	-2	-3	-4	-2	2	6									
K	-1	3	1	0	-5	1	0	-2	0	-2	-3	5								
M	-1	0	-2	-3	-5	-1	-2	-3	-2	2	4	0	6							
F	-3	-4	-3	-6	-4	-5	-5	-5	-2	1	2	-5	0	9						
P	1	0	0	-1	-3	0	-1	0	0	-2	-3	-1	-2	-5	6					
S	1	0	1	0	0	-1	0	1	-1	-1	-3	0	-2	-3	1	2				
T	1	-1	0	0	-2	-1	0	0	-1	0	-2	0	-1	-3	0	1	3			
W	-6	2	-4	-7	-8	-5	-7	-7	-3	-5	-2	-3	-4	0	-6	-2	-5	17		
Y	-3	-4	-2	-4	0	-4	-4	-5	0	-1	-1	-4	-2	7	-5	-3	-3	0	10	
V	0	-2	-2	-2	-2	-2	-2	-1	-2	4	2	-2	2	-1	-1	-1	0	-6	-2	4

BLOSUM-62

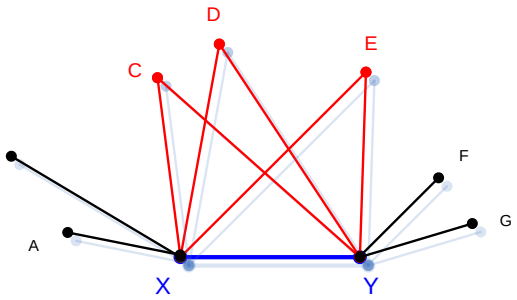
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	4																			
R	-1	5																		
N	-2	0	6																	
D	-2	-2	1	6																
C	0	-3	-3	-3	9															
Q	-1	1	0	0	-3	5														
E	-1	0	0	2	-4	2	5													
G	0	-2	0	-1	-3	-2	-2	6												
H	-2	0	1	-1	-3	0	0	-2	8											
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

$$d(a,b) = \left(1 - \frac{align(A,B)}{align(A,A)}\right)\left(1 - \frac{align(A,B)}{align(B,B)}\right)$$

Methodology – clustering

- Shared Nearest Neighbors (SNN)
  - based on Jarvis and Patrick algorithm
  - uses point distance to k-nearest neighbors
  - density-based clustering algorithm
  - similarity measure:

$$sim(x, y) = \#\{NN(x) \cap NN(y)\}$$



✓ algorithm efficiently eliminates outliers and noise from datasets

✗ number of resulting clusters is not known *a priori*

✗ algorithm is very sensitive to any change in user defined parameters

Methodology – clustering

- Shared Nearest Neighbors (SNN)

1. Compute the similarity matrix (containing nearest points)
2. Keep only the *k* most similar neighbors.

   (***k*** nearest neighbors of the similarity graph remain)
3. Construct the SNN-graph (Jarvis-Patrick algorithm).

   Similarity threshold is applied, components are connected to obtain the clusters.
4. Find the SNN density of each point.

   Using a user specified parameters, ***Eps***, find the number of points that have an SNN similarity of *Eps* or greater to the point. This is the **SNN density** of the point.
5. Find the core points.

   Using a user specified parameter, ***MinPts***, find the core points, i.e. all points that have an SNN density greater than *MinPts*.
6. Form clusters from the core points.

   If two core points are within a radius ***Eps*** of each other, then they are placed in the same cluster.
7. Discard all noise points.

   All non-core points that are not within a radius of *Eps* of a core point are discarded.
8. Assign all non-noise, non-core points to clusters.

   Assign points to the nearest core point.

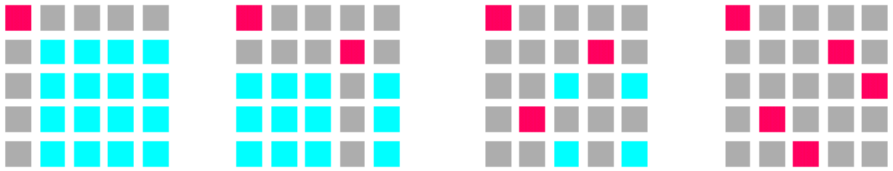
[Ertoz L., Steinbach M., Kumar V., *Finding Clusters of Different Sizes, Shapes, and Densities in Noisy, High Dimensional Data*]

Methodology – Multidimensional Scaling

- Multidimensional Scaling (MDS)
  - aim: reduction of dimensionality to smaller features set
  - method: creating a mapping reflecting distances between original and mapping set
  - algortithm: generating configuration of points in reduced space
    - calculating differences between distances in original space and generated
    - deflection from equilibrium determines formation of forces
    - minimizing non-linear stress function

Methodology – quality assessment

- Munkres Algorithm
  - founded by J. Munkres, H. Kuhn in 1957
  - known also as Hungarian Algorithm
  - solves in polinomial time the Assignment Problem (AP) of agents to tasks



Tools

- Protein database: Pfam 4.0 seed alignments database containing 27650 sequences grouped in 1467 families
- Reduction to families containing at least 25 members: 12977 sequences grouped in 279 protein families
- Algorithms implemented in C++ programming language
- OpenMP v2.0 standard used for parallelism
- Computations on SGI Altix 3700 machine, running 128 1.5GHz Intel Itanium2 processors
- Amira 5.0 Software used for visualization

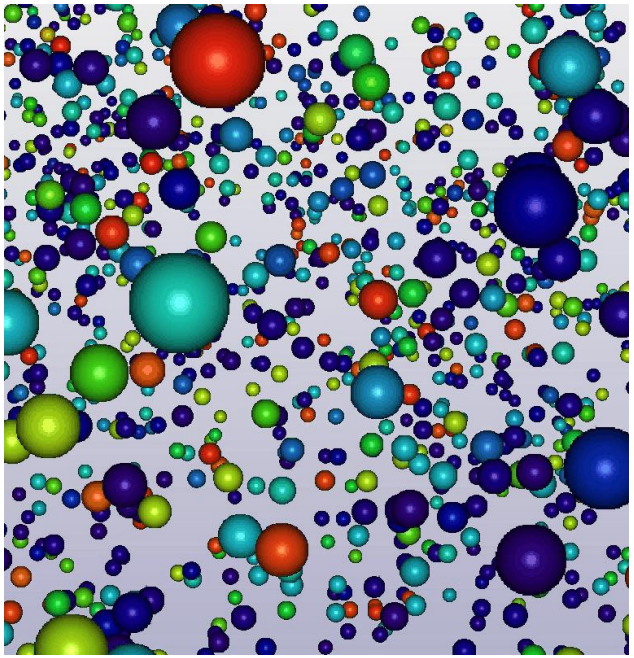
Results

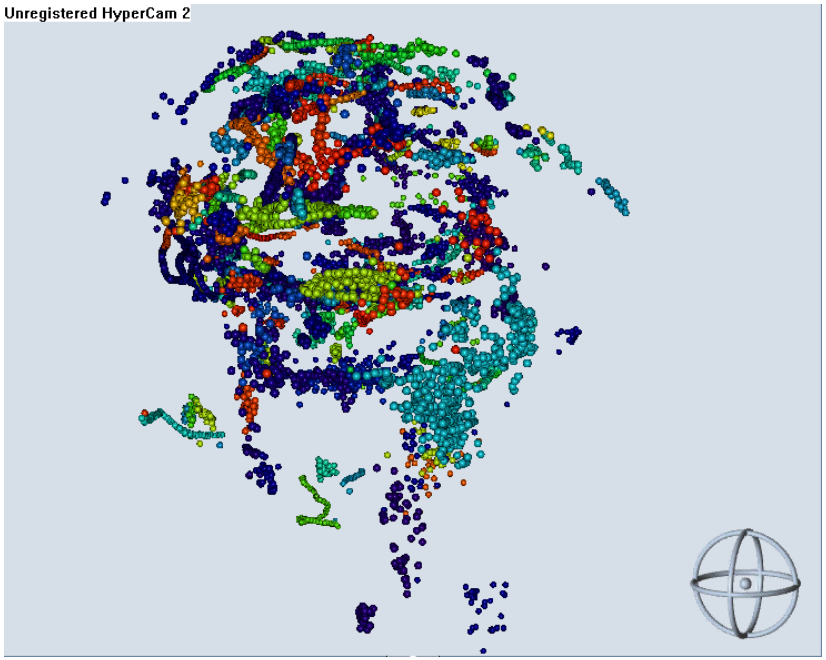
Reduced dataset (12977 protein sequences) - 90.7%

(K=30, MinPts=18, Eps=12)

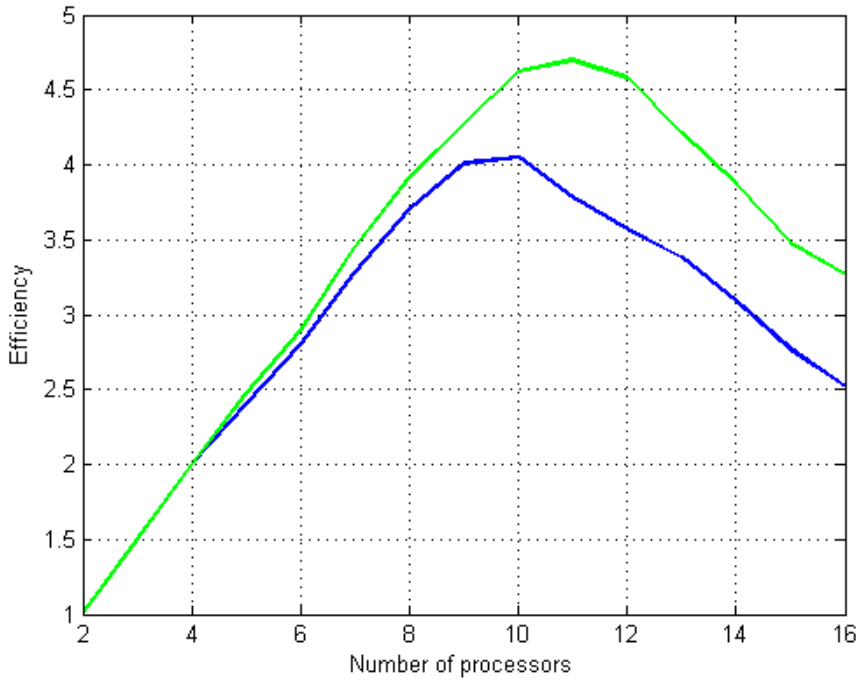
Full dataset (27650 protein sequences) – 85.4%

(K=15, MinPts=4, Eps=4)





Efficiency of calculations



Efficiency of processing 1000 (blue) and 3000 (green) random sequences of 50-500 residues on computer cluster

Conclusions

- proposed approach let us visualize similarity between the protein sequences
- computer intensive methods are being applied
  - Needleman-Wunsch Algorithm – $O(n^2)$,
  - SNN – $O(n^2)$,
  - MDS – $O(n^2)$,
  - Hungarian Algorithm – $O(n^3)$.
- complexity reduction may be achieved in MDS by storing M nearest and N futhest neighbors or by using histogram of distances
- execution time was shortened by parallel computation using OpenMP paradigm
- parallel implementation efficiency of all components needs further improvement
- latest Pfam release: 23.0 (July 2008, 10340 families, 3 925 943 sequences)

AGH

Thank you for attention.

Questions?