



Applications of Pattern Recognition To Protein Classification

Angel Kuri

Instituto Tecnológico Autónomo de México

akuri@itam.mx

CANCÚN, MÉXICO

May, 2004

Motivation

The basic idea is to achieve unbiased protein classification.

In our agenda we would like to:

a) Classify sets of proteins from simple organisms (E. coli and S. cerevisiae)

- We use Kohonen's self-organizing maps*

b) Analyze the clusters in order to determine the reasons why the proteins in the said clusters appear as they do

- We use specific pattern recognition techniques now under development*



Agenda

1. *We make a brief review of*
 - *Proteins*
 - *The genetic code*
 - *Aminoacids*
2. *We talk a little about the SOMs*
3. *We discuss lossless compression algorithms and their relationship to the problem*



Proteins (basic concepts)

- Proteins are the most important macromolecules.
- Form much of the functional and structural machinery of every cell in all organisms.
- Proteins control physicochemical conditions inside the cell, are the basic components of cellular structure, carry out the transport and storage of small molecules and are involved with the transmission of biological signals.

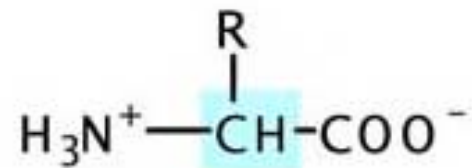


Proteins (...basic concepts)

- Proteins include the enzymes which catalyze and regulate a variety of biochemical processes in the cell as well as antibodies.
- Each type of cell has several thousand kinds of proteins which play a primary role in determining the characteristics of the cell and how it functions.
- Proteins are complex molecules, assembled from 20 different amino-acids.

Proteins (...basic concepts)

- An amino acid is defined as the molecule containing an amino group (NH₂), a carboxyl group (COOH) and an R group. It has the following general formula:



The R group differs among various amino acids.



Proteins (..basic concepts)

- There are over 300 naturally occurring amino acids on earth, but the number of different amino acids in proteins is only 20.
- The Carbon atom at the center of the molecule is called the alpha Carbon.
- The amino acids form the vocabulary that allows proteins to exhibit a great variety of structures and properties.



Proteins (...basic concepts)

- Each protein sequence is encoded in a DNA sequence called a gene, in which every block of three nucleic acids (codon) corresponds to an individual amino acid.
- The set of rules that specify which amino acid is encoded in each codon is called the genetic code.
- The R group is the one that determines the physicochemical characteristics of each amino acid.

The Genetic Code

	U	C	A	G	
U	UUU Phenyl UUC alanine UUG Leucine UUA	UCU UCC Serine UCA UCG	UAU Tyrosine UAC UAA Stop UAG	UGU Cysteine UGC UGA Stop UGG Tryptophan	U C A G
C	CUU CUC Leucine CUA CUG	CCU CCC Proline CCA CCG	CAU Histidine CAC CAA Glutamine CAG	CGU CGC Arginine CGA CGG	U C A G
A	AUU AUC Isoleucine AUA AUG Methionine	ACU ACC Threonine ACA ACG	AAU Asparagine AAC AAA Lysine AAG	AGU Serine AGC AGA Arginine AGG	U C A G
G	GUU GUC Valine GUA GUG	GCU GCC Alanine GCA GCG	GAU Aspartic acid GAC GAA Glutamic acid GAG	GGU GGC Glycine GGA GGG	U C A G

Name (Residue)	3-letter code	Single code	Relative abundance (%) E.C.	MW	pK	VdW volume(Å ³)	Charged, Polar, Hydrophobic
Alanine	ALA	A	13.0	71		67	H
Arginine	ARG	R	5.3	157	12.5	148	C+
Asparagine	ASN	N	9.9	114		96	P
Aspartate	ASP	D	9.9	114	3.9	91	C-
Cysteine	CYS	C	1.8	103		86	P
Glutamate	GLU	E	10.8	128	4.3	109	C-
Glutamine	GLN	Q	10.8	128		114	P
Glycine	GLY	G	7.8	57		48	-
Histidine	HIS	H	0.7	137	6.0	118	P,C+
Isoleucine	ILE	I	4.4	113		124	H
Leucine	LEU	L	7.8	113		124	H
Lysine	LYS	K	7.0	129	10.5	135	C+
Methionine	MET	M	3.8	131		124	H
Phenylalanine	PHE	F	3.3	147		135	H
Proline	PRO	P	4.6	97		90	H
Serine	SER	S	6.0	87		73	P
Threonine	THR	T	4.6	101		93	P
Tryptophan	TRP	W	1.0	186		163	P
Tyrosine	TYR	Y	2.2	163	10.1	141	P
Valine	VAL	V	6.0	99		105	H

Protein Expression

The following sequence is the expression of a protein of E. coli:

MARKTKQEAQETRQHILDVALRLFSQQGVSSTS
LGEIAKAAGVTRGAIYWHFKDKSDLFSEIWELF
RPCKRCQPEKANAQQHRLDKITHACRLLEQETP
VTLEALADQVAMSPFHLHRLFKATTGMTPKAWQ
QAWRARRLRESLAKGESVTTSILNAGFPDSSSY
YRKADETLGMTAKQFRHGGENLAVRYALADCEL
GRCLVAESERGICAILLGDDDATLISELQQMFP
AADNAPADLMFQQHVREVIASLNQRDTPL



Protein Structure

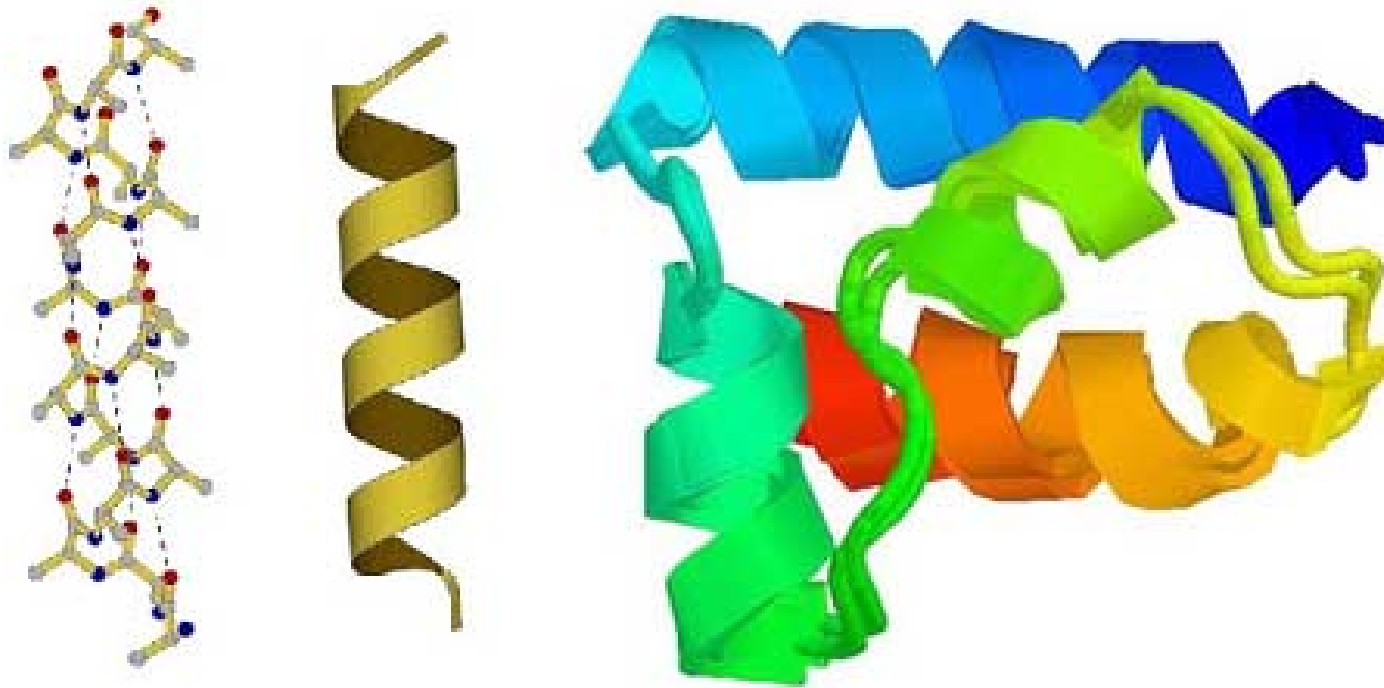
- The specific amino acid sequence of a protein is called the **primary structure** of the protein.
- The average length of a protein sequence is 350 amino acids, but it can be as short as few amino acids, and as long as a few thousand (5000 the longest known).
- According to the central dogma of protein folding, the protein sequence (primary structure) dictates how the protein folds in three dimensions. It is the specific three dimensional structure that enables the protein to function in its particular biological role.



Protein Structure

- **Secondary Structures** are local sequence elements (30-40 aa's long) that have a well determined regular shape, such as alpha helix or beta strand (beta sheet), also other local sequence elements exist and are called loops or coils.
- Secondary structures are packed into what is called the **Tertiary Structure**.

Protein's secondary structure



Alpha helix

Hydrogen bonds play a role in stabilizing the α helix conformation. However, the size and charges of sidechains are also important factors

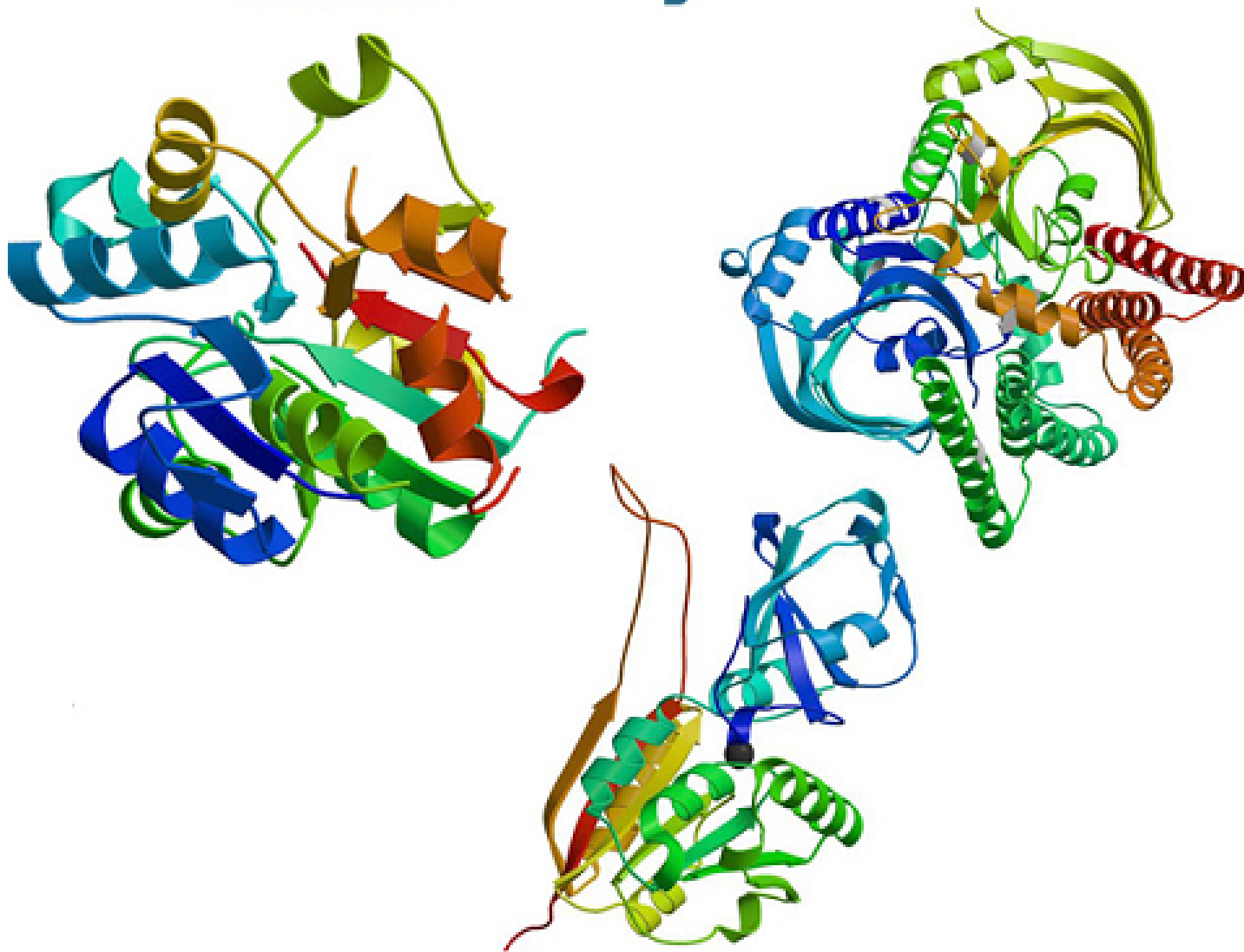
Protein's secondary structure



Beta Strand (sheet)

A Beta sheet consists of two or more hydrogen bonded Beta strands. The two neighboring Beta strands may be parallel if they are aligned in the same direction from one term (N or C) to the other, or anti-parallel if they are aligned in the opposite direction

Protein's tertiary structure





Protein Classification

- Many different criteria has been used in order to clasify the universe of proteins. Some of this criteria has to be with physical properties (solubility), other with chemical properties and some other more with shape and functional characteristics. All this methods were developed to organize the proteins whose structure and function were well known.



Protein Classification

- After the Big Bang of molecular biology, when hundreds of thousand new protein sequences have been described, new classification approaches have been developed that try to organize this new proteins, about which we have little or no information. A unified scheme should be based on a natural (evolutionary) classification approach.
- The three dimensional structure of a protein gives the most information about its biological function, but determining the structure of a protein is difficult. At the moment there are only several thousand known structures.



...Protein Classification

- In the absence of structural data, sequence analysis remains the main source of information for most new proteins.
- In most cases, sequence similarity entails similar or related functions. Detecting similarities between protein sequences can help to reveal the biological function of new protein sequences, as well as their origin and relations with other proteins.



...Protein Classification

- Sequence similarity is not always easily detectable. During evolution, sequences have changed by insertions, deletions and mutations. Some of these evolutionary events may be traced today by applying algorithms for sequence comparison.
- When sequences share a significant sequence similarity they are usually assumed to have a common evolutionary ancestry, and are called **homologous proteins**.



...Protein Classification

- Given a new protein sequence, the current approach to predicting its function and analyzing its properties hinges on pairwise comparisons with the sequences of other proteins whose properties are already known.
- Pairwise comparisons have shown not being enough to organize and classify the big volume of data already available. New techniques, most of them involving multiple alignments have been developed.
- Sequence comparison is not a straightforward process, as many evolutionary and physicochemical elements should be taken into account when comparing two protein sequences. Much nomenclature has been developed to achieve this goal.

...Protein Classification

- A good protein classification system must take into account at least the following elements: 1) Sequence, 2) Structure, 3) Biomolecular interactions and 4) Sub-cellular localization.
- None of the available databases takes into account more than one of these elements.
- But the information exists in the databases and we need a method to make this information converge, to cross information between the databases.
- We may use Kohonen maps to achieve this task.
Problem: Represent sequence, structure and function of each protein with a finite number of variables independent of both, the sequence length and any multiple alignment data.



Hypothesis

*It is possible to achieve classification of the proteins of a living organism (we shall focus on *E. coli* and *S. cerevisiae*) paying attention solely to the structural characteristics (strings of aminoacids) of the proteins.*



Clustering

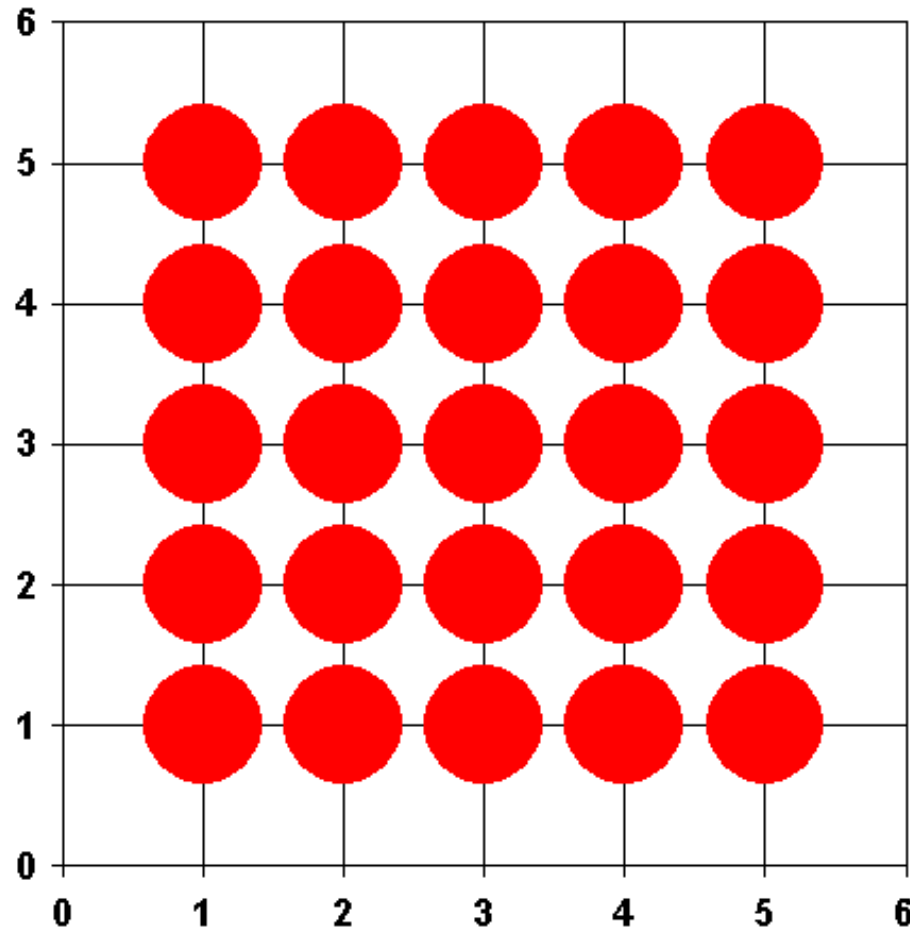
The first problem is to attain the automatic clustering of the diverse proteins.

To do this, we shall use self-organizing maps in which the determination of the cluster membership is achieved using genetic algorithms.

Step 1:

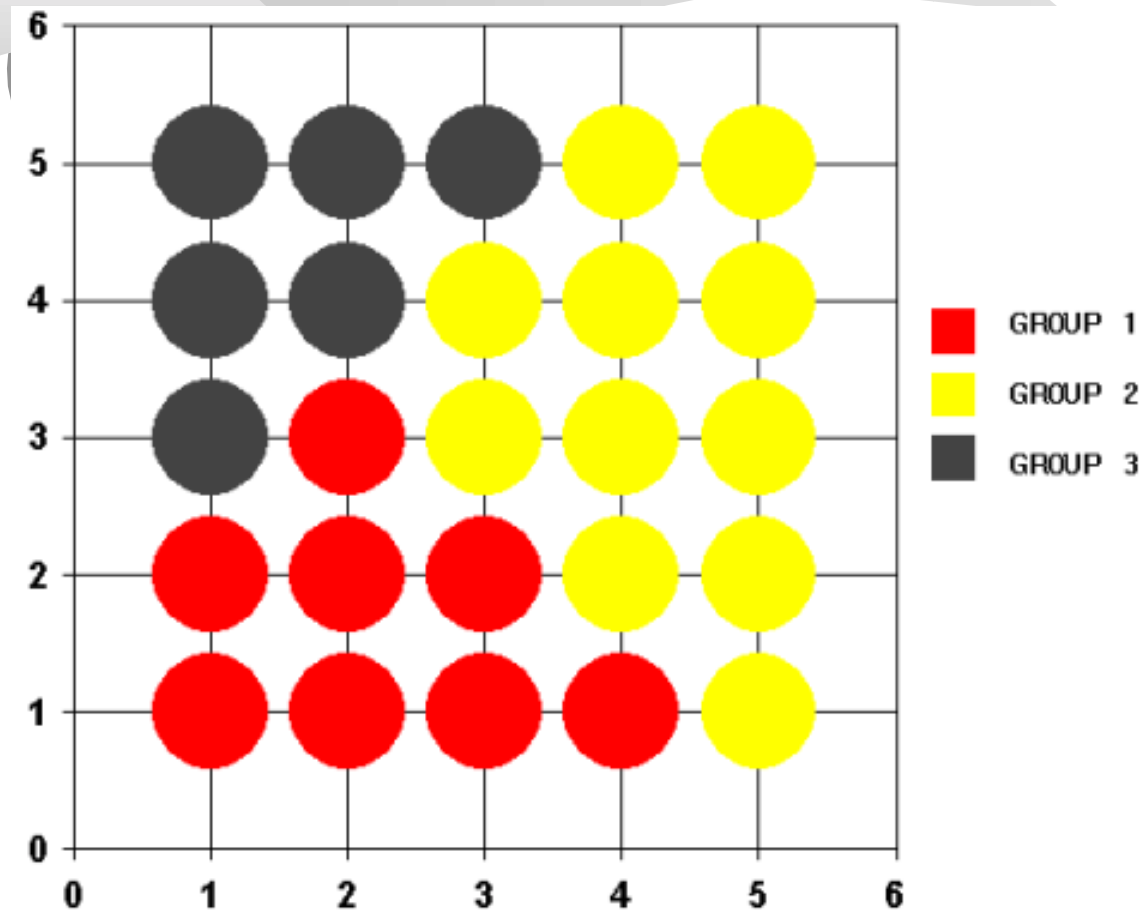
*In this map all
neighbouring neu-
rons belong to a
cluster.*

*But we do not
know the
clusters' bound-
aries.*



An Example of a SOM

In this map the neurons have been labeled, so that we know to which cluster each neuron belongs.



Present Status

- ☞ *As of today, we have achieved initial success by finding sets of proteins whose basic clustering is derived from structural relationships between the aminoacids*
- ☞ *Much work remains to be done*



Explaining the Clusters

By using SOMs we may find non-biased clusters.

*To explain **why** they cluster in such way it is possible to apply meta-symbolic search algorithms.*



Explaining the Clusters

Our second task is, perhaps, more challenging than the first one

Once we achieve structural clustering, we would like to find common structures in the proteins in each cluster

To this effect we apply data compression techniques

The basic idea is to remove redundancy from the proteins and THEN look for similarities



A Protein as a Message

As stated, a protein may be expressed as a string of symbols (aminoacids)

In this sense, what we assume is that the original expression of any such protein may be replaced by a shorter, more compact way

We propose to identify the underlying patterns in order to uncover a similarity measure between different proteins

Lossles Data Compression

Symbol	P(Si)	Sum	Code	Length	Avg(L)	Std
S1	0.50000000	1.00000	0	1	0.500	3
S2	0.25000000	0.50000	10	2	0.500	3
S3	0.12500000	0.25000	110	3	0.375	3
S4	0.06250000	0.12500	1110	4	0.250	3
S5	0.03125000	0.06250	11110	5	0.156	3
S6	0.01562500	0.03125	111110	6	0.094	3
S7	0.00781250	0.01563	1111110	7	0.055	3
S8	0.00781250		1111111	7	0.055	3
			SUMA		1.984	3

Information Theory

The information of a symbol (as per Shannon), is given by:

$$I(S_i) = -\log_2(P_i)$$

where P_i = probability that symbol S_i appears.

The average information (Entropy) is given by:

$$H(S) = \sum_i P_i \cdot I(S_i)$$

Lossles Data Compression

Symbol	$P(S_i)$	$I(S_i)$	Avg. Information
S1	0.50000000	1.00000	0.50000
S2	0.25000000	2.00000	0.50000
S3	0.12500000	3.00000	0.37500
S4	0.06250000	4.00000	0.25000
S5	0.03125000	5.00000	0.15625
S6	0.01562500	6.00000	0.09375
S7	0.00781250	7.00000	0.05469
S8	0.00781250	7.00000	0.05469
		Entropy	1.98438

Lossless Data Compression

Notice that the optimal average length is bounded by the entropy.

When as here, the probabilities are powers of 2, it is possible to reach this limit.

When such is not the case, the theoretical bound cannot be reached using this kind of encoding (*Huffman Coding*, after its creator).

The limitations of Information Theory

One of the tacit premises in classical IT is that the “symbols” are entities defined a priori (bytes, words, etc.) whose grouping relationship implies a topologic closeness.

For example, if we consider letter couples, we normally consider them to be neighbours. In English, the couple “th” implies that $P(e|th)$ is very high.

Ergodicity

One of the assumed characteristics of the data source, for the encoding to be effective, is ergodicity.

Intuitively, a source is ergodic if “its” probabilities “stabilize” after a bounded period of time.

A counter-example would be the one where we transmitted a block of English text, followed by an image (i.e. “jpg”).



Ergodicity

In the last example clearly, the probabilities of the first block will differ from the ones in the second block.

We emphasize the fact that we have called “probabilities”, in practice, refers to the proportions gotten from the statistical analysis of data blocks.

“Transformation” of non-ergodic into ergodic Sources

The agenda we have set is to find sets of not necessarily neighboring symbols in a non-ergodic source.

*If we achieve this, every set of such symbols (called a **metasymbol**) will replace a symbol in an equivalent ergodic source and will allow us to apply first order techniques to independent clusters.*

Huffman

Assume the sample:

A A A B A A A A B A A B A A B B

A appears 11 times

B appears 5 times

Only two symbols.

Huffman assigns: A = 0, B = 1.

Higher Orders

digram	<i>fre</i>	<i>cH</i>	3-gram	<i>fre</i>	<i>cH</i>	4-gram	<i>fre</i>	<i>cH</i>
AA	<i>4</i>	<i>0</i>	AAB	<i>3</i>	<i>0</i>	AAAB	<i>1</i>	<i>00</i>
AB	<i>2</i>	<i>10</i>	AAA	<i>1</i>	<i>10</i>	AAAA	<i>1</i>	<i>01</i>
BA	<i>1</i>	<i>111</i>	BAA	<i>1</i>	<i>111</i>	BAAB	<i>1</i>	<i>10</i>
BB	<i>1</i>	<i>110</i>	B##	<i>1</i>	<i>110</i>	AABB	<i>1</i>	<i>11</i>
<i>Total</i>	<i>14 bits</i>			<i>11 bits</i>			<i>8 bits</i>	

Dictionary Methods

- ➡ *A dictionary with **frequent strings** is built.*
- ➡ *Every instance of the string is replaced by a reference to the dictionary.*

Example

(a piece of a poem by Sor Juana)

AL QUE INGRATO ME DEJA, BUSCO AMANTE;
AL QUE AMANTE ME SIGUE, DEJO INGRATA;
CONSTANTE ADORO A QUIEN MI AMOR MALTRATA;
MALTRATO A QUIEN MI AMOR BUSCA CONSTANTE

- | | |
|----------------------|--------------|
| 1. AL_QUE_ | 6. MALTRAT |
| 2. INGRAT | 7. CONSTANTE |
| 3. _ME_ | 8. DEJ |
| 4. _AMANTE | 9. BUSC |
| 5. _A_QUIEN_MI_AMOR_ | |



Result

12O38A, 9O 4;
143SIGUE, 8O 2A;
7 ADORO56A;
6O59A 7



Or else...

AL_QUE_INGRATO_ME_DEJA,_BUSCO_AMANTE;_
AL_QUE_AMANTE_ME_SIGUE,_DEJO_INGRATA;_
CONSTANTE_ADORO_A_QUIEN_MI_AMOR_MALTRATA;_
MALTRATO_A_QUIEN_MI_AMOR_BUSCA_CONSTANTE

☞ *Build a dictionary with **patterns**,
not merely with strings.*

A possibility...

D H C F E C B A
 B C A D G A D G
 E F C D F C E B
 A D G A D A D H
 G F D H B D B E
 E G B D A C E F
 H G E F E A H A
 G A D F A D G H

A	4	E	3
B	1	F	2
C	2	G	1
D	2	H	1

Entropy : 2.506





Max. Entropy : 2.585






Pattern	Frequency	Size
F B α E G A D	2	6
D H β B E F	3	5
C γ G H	2	3
A D G δ	3	3
D C ϵ A	2	3

Sim.	Freq.	Prob.	\log_2	Huffman
α	2	0.154	2.699	001
β	3	0.231	2.114	01
γ	2	0.154	2.699	110
δ	3	0.231	2.114	10
ϵ	2	0.154	2.699	111
η	1	0.076	3.718	000

...but not the only one

D	H	C	F	E	C	B	A
B	G	A	D	G	A	D	G
E	E	C	D	F	C	E	B
A	D	G	A	D	A	D	H
G	F	D	H	B	D	B	E
E	G	B	D	A	C	E	F
H	G	E	F	E	A	H	A
G	A	D	F	A	D	G	H

	E	4
	A	2
	F	2
	G	1
	C	1

Pattern	Frequency	Size	Cover
α 	2	12	24
β 	2	12	18
γ 	2	8	8
δ 	3	2	4
ϵ 	1	10	10

10

64

$$H_{max} = 2.3219 \text{ (5 symbols)}$$

$$H = 2.2464$$

Full message : $\alpha\beta\epsilon\beta\gamma\gamma\delta\alpha\delta\delta$


Metasymbol compression process

- 1. A message is given. Find the set of patterns which more frequently appear in the message.*
- 2. Find the set of patterns which allows the shortest expression of the message including the “catalog” (a description of the metasymbols).*
- 3. Encode the message using the patterns in the catalog*
- 4. Optionally also encode such catalog in the shortest possible way.*

Finding Patterns...

Characteristics of the Patterns:

- They are NOT strings of consecutive symbols. They show “gaps”.*
- The size of the patterns and of the gaps are arbitrary.*

 *Reported algorithms to search for these sort of patterns have exponential complexity (on the size of the message).*



...is hard!

- ☞ Look for the place where the pattern overlaps with itself; find patterns of frequency = 2.*
- ☞ Find the intersections of these; then the intersections of the intersections...and so on.*
- ☞ The number of intersections grows exponentially.*
- ☞ We have proven that finding the largest arbitrary pattern of maximum length is NP-complete.*

The MaximumCommonPattern Problem

- ➡ *It is P-verifiable: given a maximum length pattern proposal it takes polynomial time to determine whether it is really common to all strings.*
- ➡ *Reduction: given any other NP-complete problem, show that it may be mapped in polynomial time to the MaximumCommonPattern*
- ➡ *VertexCover was chosen as the NP-complete “template”.*

Finding a subset

- ☞ If we assume that we already have a set of frequent patterns we must then find a subset of them which allows us to express the message and the catalog in the shortest possible way.*
- ☞ We proved that this problem is also NP-complete.*

The OptimalPatternSubset Problem

- ☞ *It is P-verifiable: given a proposal of a subset and the optimal compression ratio we may verify in polynomial time whether the subset is really optimal.*
- ☞ *Reduction: given any instance of an NP-complete problem known to be NP-complete, show that it can be mapped to an instance of the OptimalPatternSubset.*
- ☞ *Knapsack 0-1 was selected.*

To wrap up

- ☞ *We need approximation algorithms and/or heuristics.*
 - *To find the subset we found a promising cover-based heuristic.*
 - *We must find those patterns which better cover the message*
 - *We may then refine with hill-climbers*

Heuristics and Meta-heuristics

Given the above, we have complemented the heuristics with a genetic algorithm.

This GA is not Holland's Simple Genetic Algorithm but, rather, one we have called "Vasconcelos' GA" (VGA)

Putting the heuristics and VGA together we have found some interesting results.

Mini-agenda

- ➡ *Vasconcelos' Genetic Algorithm*
- ➡ *Encoding*
- ➡ *Mutation*
- ➡ *Crossover*
- ➡ *Experiments and some results*

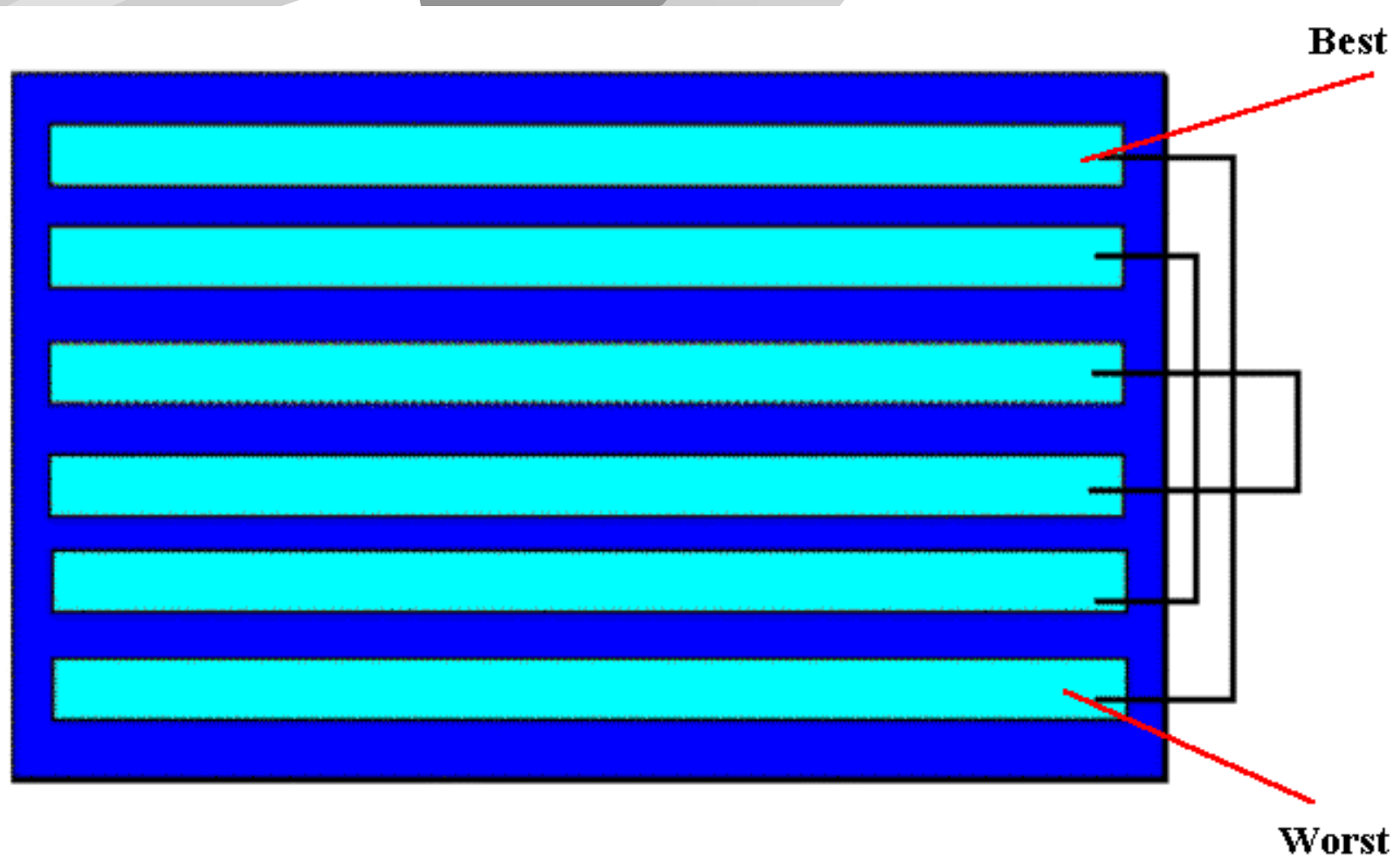
Vasconcelos' GA

To overcome the limitations of a SGA we introduced the so called Vasconcelos GA.

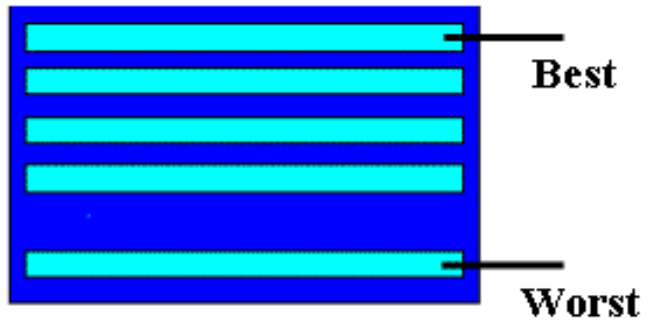
It displays:

- a) Deterministic ($i \rightarrow n-i+1$) coupling***
- b) Full elitism***
- c) Annular crossover***
- d) Uniform mutation***

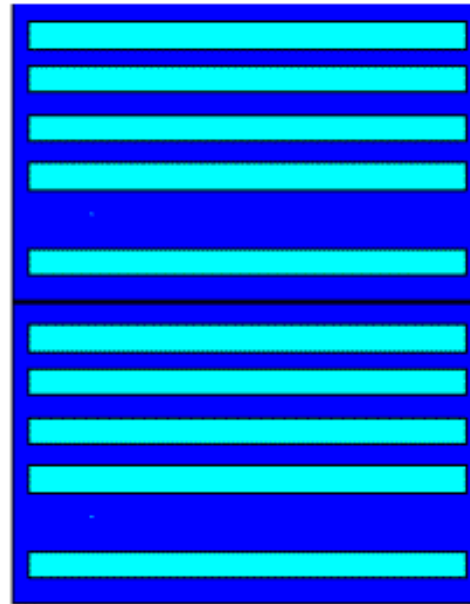
VGA



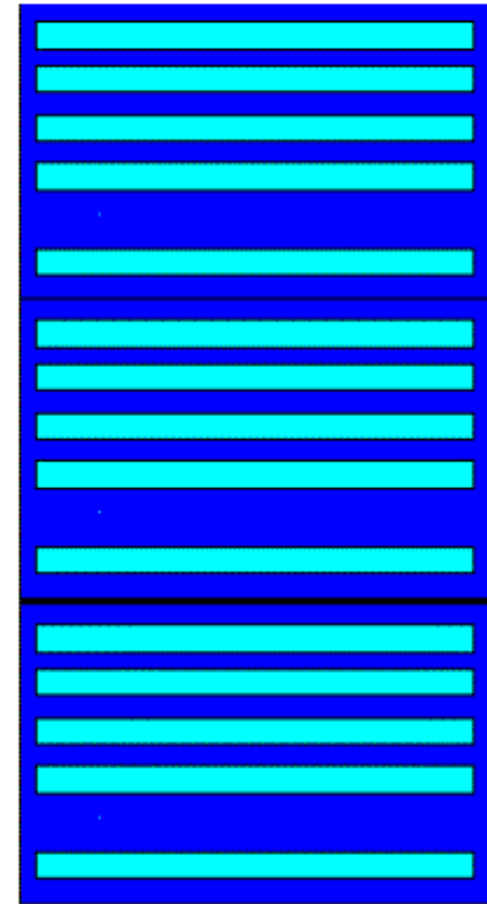
VGA



1st generation

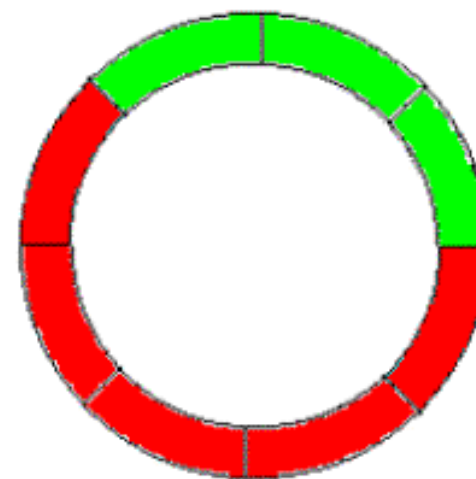
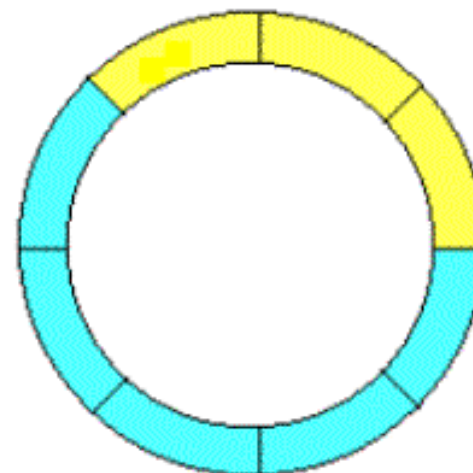
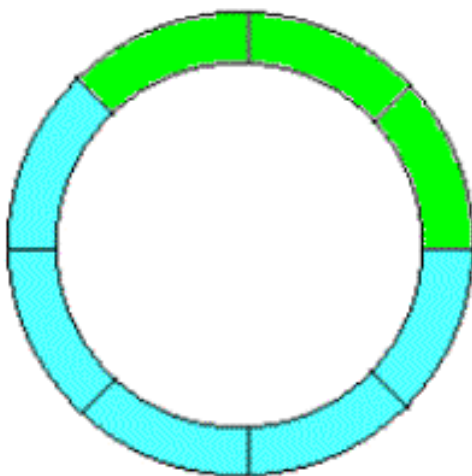
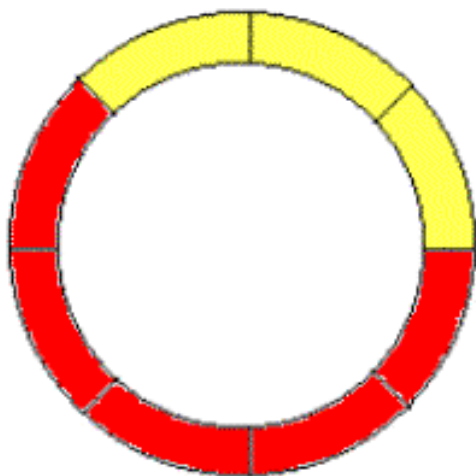


2nd generation



3rd generation

VGA




Encoding

The message is looked upon as an array of symbols “ $A_0B_1d_2C_3A_4B_5e_6C_7f_8$ ”

The genome is made up of the indices of the symbols

$013 \ 457 \ 268 \quad \rightarrow \quad A_0B_1C_3 \ A_4B_5C_7 \ d_2e_6f_8$



Mutation

It consists of a permutation of two indices

4 **5** 7 0 1 3 **2** 6 8

4 **2** 7 0 1 3 **5** 6 8

Crossover

4 5 7 0 1 8 2 6 3 Individual A

8 4 1 3 2 0 7 6 5 Individual B



4 5 7 3 1 8 2 6 3 Individual A

8 4 1 0 2 0 7 6 5 Individual B

4 5 7 3 1 8 2 6 0 Individual A

8 4 1 0 2 3 7 6 5 Individual B

Catastrophe

cuando_cuentas_cuentos_cuantos_cuentos_cuentas

*cuando_*uentas_*uentos_*uantos_*uentos_*uentas*

*cuentas_*uentos_*uantos_*uentos_*uentas*

*cuentos_*uantos_*uentos_*uentas*

*cuantos_*uentos_*uentas*

*cuentos_*uentas*

cuentas

Catastrophe

cuando_cuentas_cuentos_cuantos_cuentos_cuentas



*cuando_**entas_**entos_**antos_**entos_**entas*

*cuentas_**entos_**antos_**entos_**entas*

*cuentos_**antos_**entos_**entas*

*cuantos_**entos_**entas*

*cuentos_**entas*

cuentas

Catastrophe

cuando_cuentas_cuentos_cuantos_cuentos_cuentas



*cuando_**e*ta*_**e*to*_**a*to*_**e*to*_**e*ta**

*cuentas_**e*to*_**a*to*_**e*to*_**e*ta**

*cuentos_**a*to*_**e*to*_**e*ta**

*cuantos_**e*to*_**e*ta**

*cuentos_**e*ta**

cuentas

Catastrophe

cuando_cuentas_cuentos_cuantos_cuentos_cuentas

↓ ↓

cuando_**e**tas*_**e**os_**a**os_**e**os_**e**as
cuentas_**e**tos*_**a**os_**e**os_**e**as
cuentos_**a**tos*_**e**os_**e**as
cuantos_**e**tos*_**e**as
cuentos_**e**tas*
cuentas

“Garbage collector”

cuando_cuentas_cuentos_cuantos_cuentos_cuentas



*cuando_**e**as_**e**os_**a**os_**e**os_**e**as*

*cuentas_**e**os_**a**os_**e**os_**e**as*

*cuentos_**a**os_**e**os_**e**as*

*cuantos_**e**os_**e**as*

*cuentos_**e**as*

cuentas

Catastrophe

cuando_cuentas_cuentos_cuantos_cuentos_cuentas



cuando_ ^{*e*}^{a*} ^{*e*}^{o*} ^{*a*}^{o*} ^{*e*}^{o*} ^{*e*}^{a*}

cuentas_ ^{*e*}^{o*} ^{*a*}^{o*} ^{*e*}^{o*} ^{*e*}^{a*}

cuentos_ ^{*a*}^{o*} ^{*e*}^{o*} ^{*e*}^{a*}

cuantos_ ^{*e*}^{o*} ^{*e*}^{a*}

cuentos_ ^{*e*}^{a*}

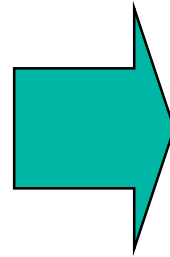
cuentas

cuando_ cuentas_ cuentos_ cuantos_ cuentos_ cuentas

Catastrophe

cuando_cuentas_cuentos_cuantos_cuentos_cuentas

cuando_ **e*a_* **e*o_* **a*o_* **e*o_* **e*a**
*cu*ent*a*s_ **e*o_* **a*o_* **e*o_* **e*a**
*cu*ent*a*s_ **a*o_* **e*o_* **e*a**
*cu*ant*a*s_ **e*o_* **e*a**
*cu*ent*a*s_ **e*a**
*cu*ent*a*s



Metasímbolo

c₁u₂n₁t₂s

cuando_ *cu*ent*a*s_ *cu*ent*a*s_ *cu*ant*a*s_ *cu*ent*a*s_ *cu*ent*a*s

Finding the Metasymbols

➡ *Applying the previous operators of coupling, selection, crossover and mutation to arbitrary messages it is possible to find the solution to the compression problem by using the metasymbolic transform*

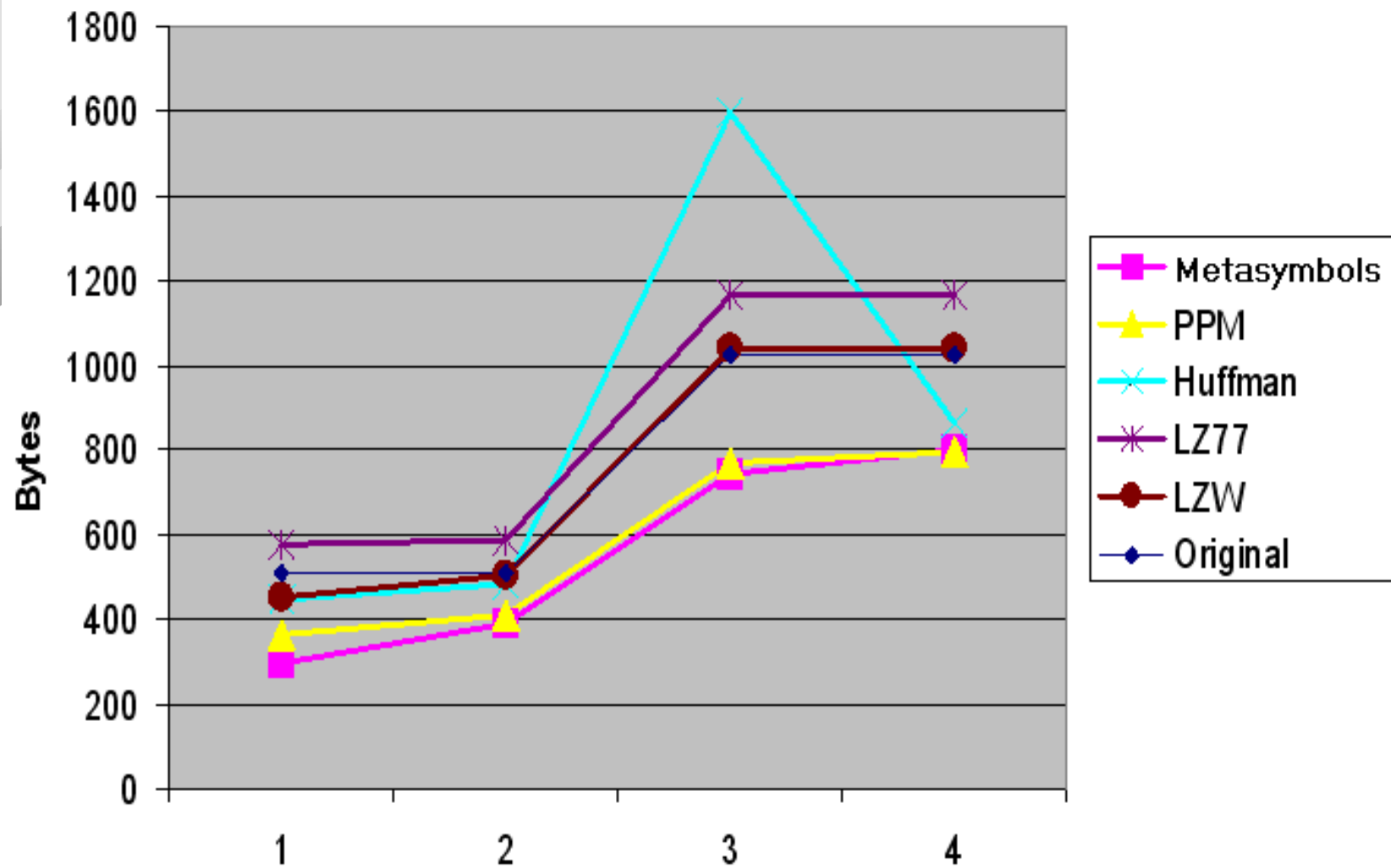
Comparing compression methods

- ➡ *It is now possible to establish a comparison of several compression schemes*
- ➡ *In the following table we show the results of compressing sets of size 512 and 1,024*

Results

Original	Metasymbols	PPM	Huffman	LZ77	LZW
512	294	366	448	580	454
512	388	412	482	589	505
1,024	746	770	1,596	1,165	1,041
1,024	803	796	863	1,164	1,043

...Results





Explaining the messages

To these algorithms a string of aminoacids is not distinguishable from a string of letters, or pixels, or...

Hence, the meta-symbols embedded in the clusters may explain why the clusters arise as they do.



Conclusions

It is possible to find unbiased clusters of proteins from protein expression as aminoacids

It is possible (and hard) to find metasymbols in arbitrary sets of data

Applying genetic algorithms + heuristics we are able to approximate the solution of these NP problems



Conclusions

Once proteins are re-expressed as collections of metasymbols the underlying patterns are easier to detect

Applying search techniques originally stemming from lossless data compression it is possible to find the reasons behind protein clustering