Modeling and Simulation of Proteins

V. Sundararajan
C-DAC, Pune

Overview

- What are Proteins?
- What is protein folding?
- Protein Structure Prediction
- Genetic Algorithms
- Sample results
- Conclusion

What are Proteins ?

Proteins

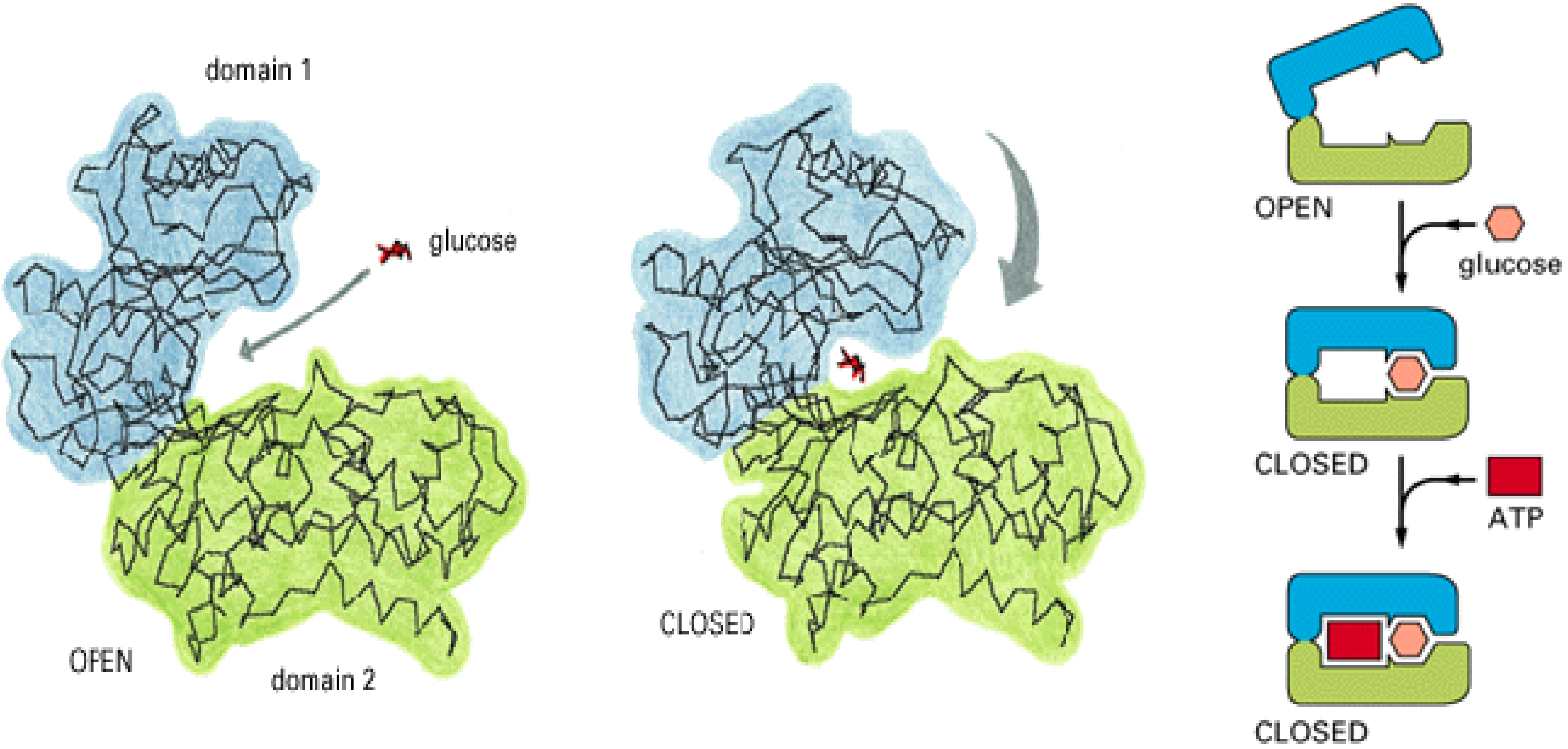
- Linked amino acid chains
- Covalently bonded

What are Proteins ?

1. Structural - viral coat proteins, molecules of the cytoskeleton, epidermal keratin
2. Catalytic - enzymes
3. Transport and storage – haemoglobin, myoglobin, ferritin
4. Regulatory – including hormones and many proteins that control genetic transcription
5. Proteins of immune system
6. Immuno-globulin superfamily involved in cell-cell recognition and signalling

http://biop.ox.ac.uk/www/mol_of_life/index_c.htm

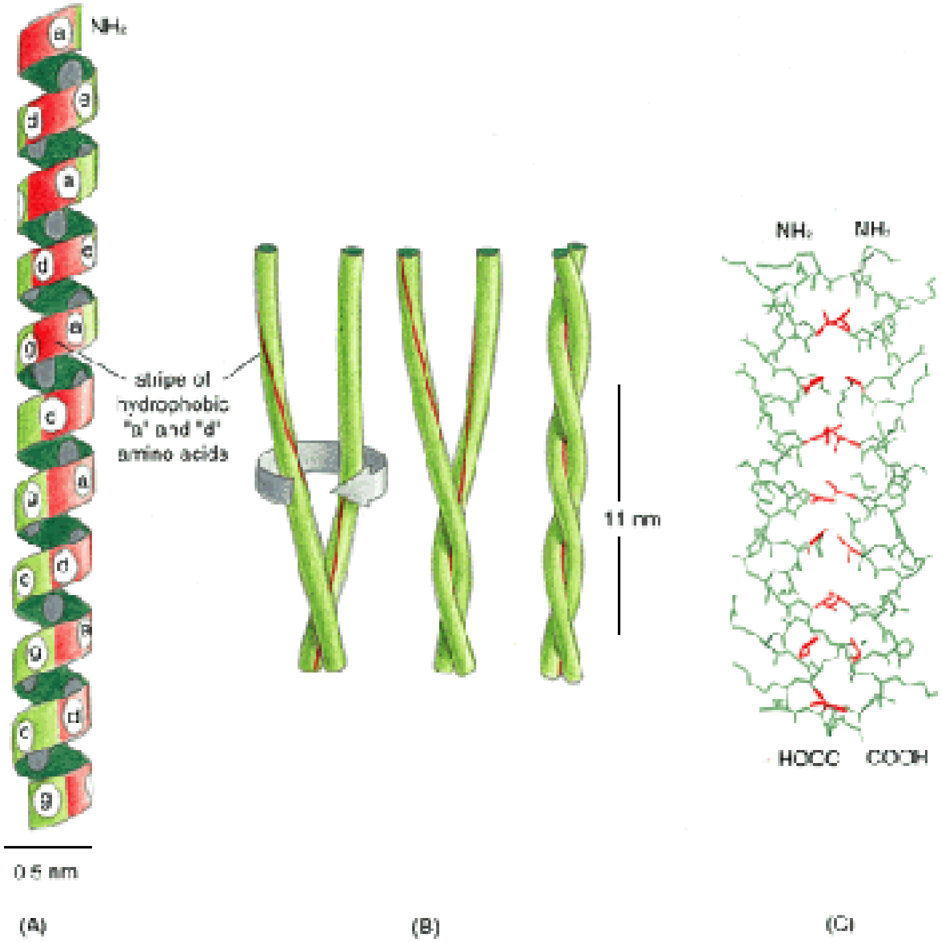
What are Proteins ?



The conformational change in hexokinase caused by glucose binding which helps Adenosine Triphosphate (ATP) binding

Molecular Biology of the Cell, Alberts, et al

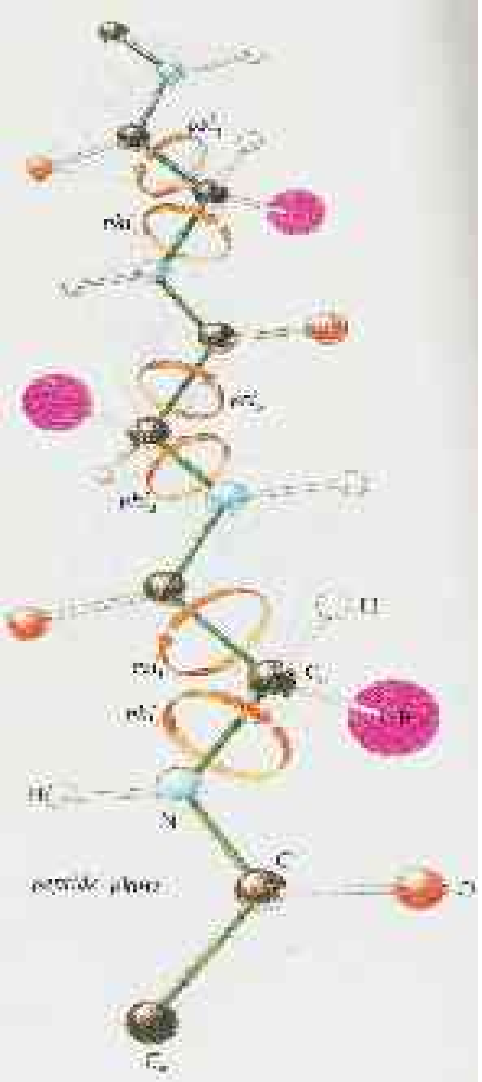
What are Proteins ?



Coiled coils serve as dimerisation domain in gene regulatory protein and as a building block for large fibrous structure

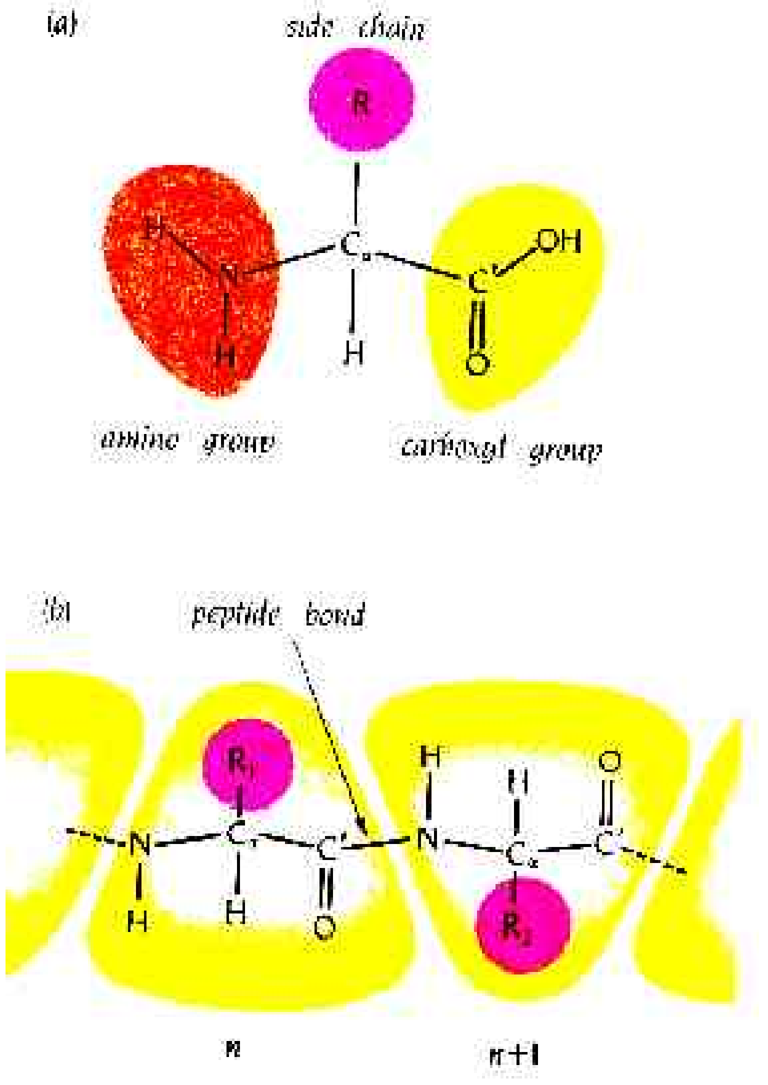
Molecular Biology of the Cell, Alberts, et al

Polypeptide



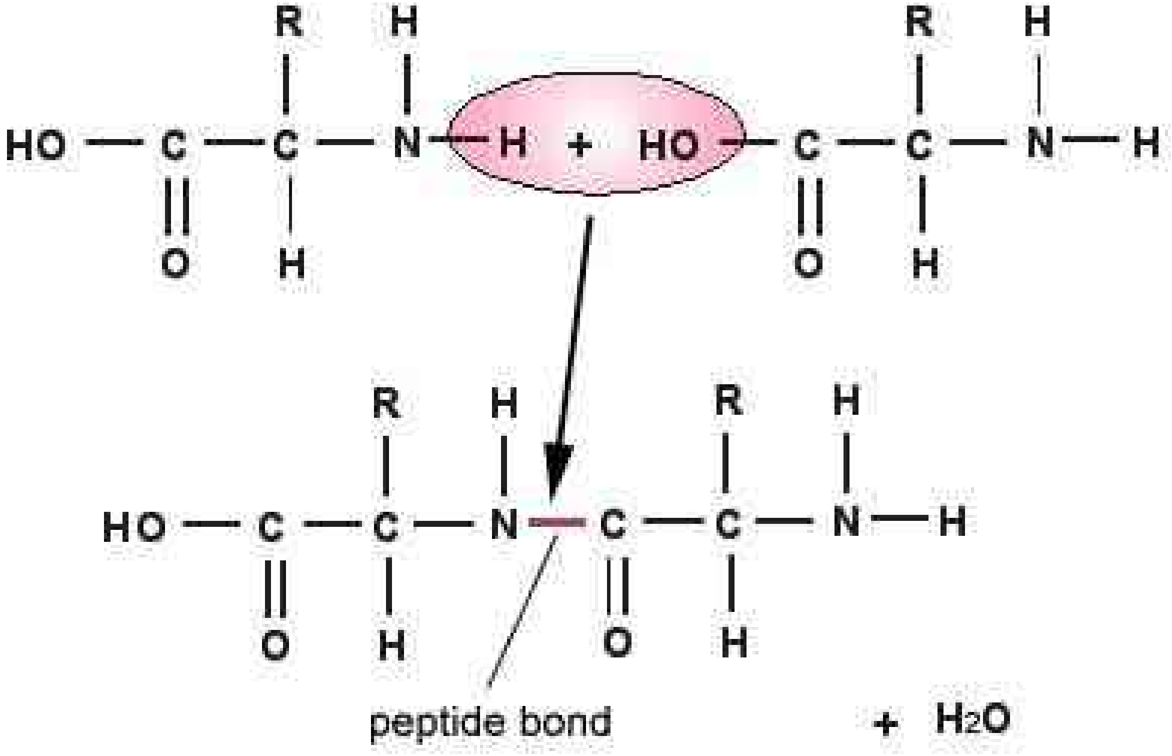
Introduction to Protein Structure, Branden and Tooze

Peptide Bond



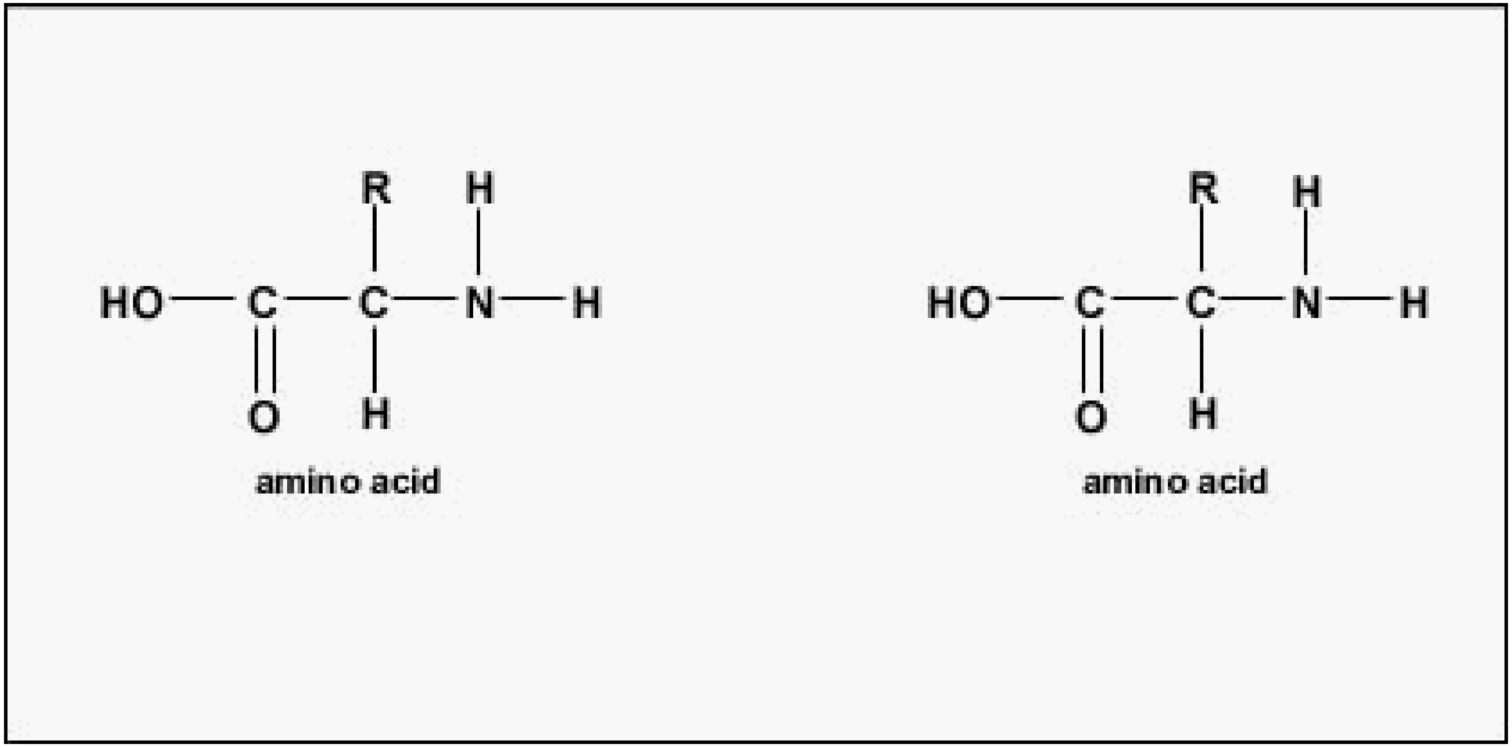
Introduction to Protein Structure, Branden and Tooze

Peptide Bond



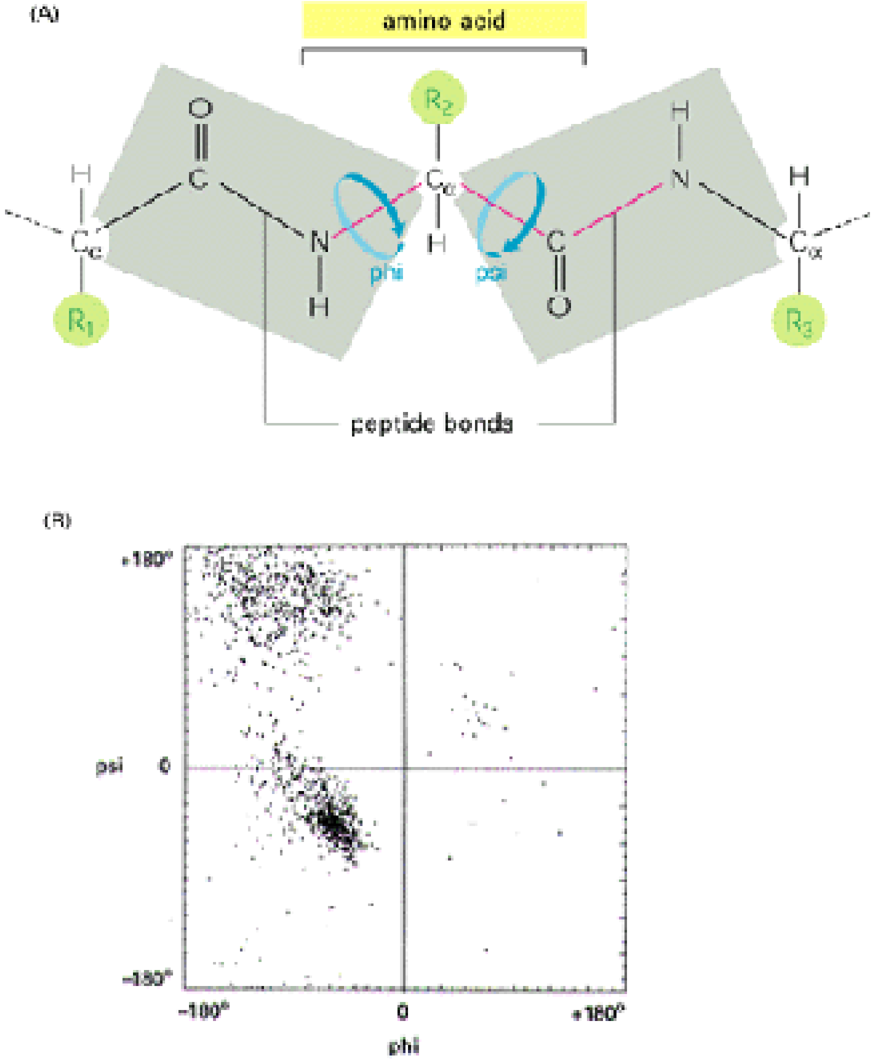
http://www.cat.cc.md.us/biotutorials/proteins/protein.html

Peptide Bond



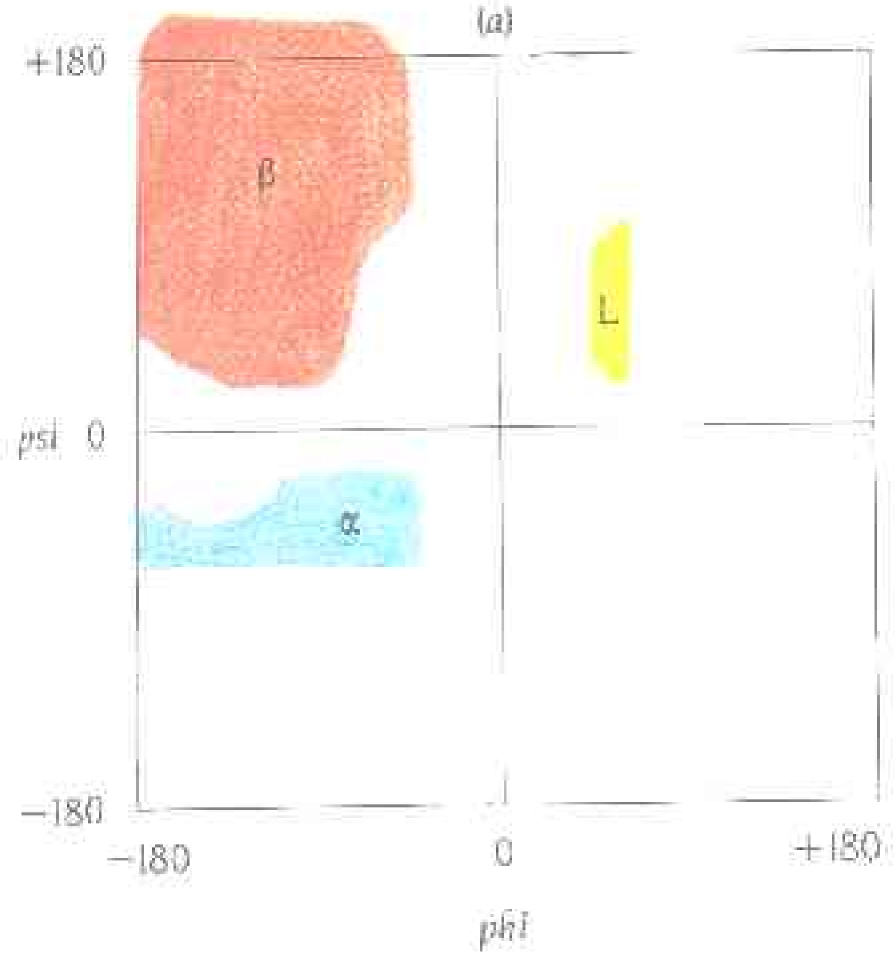
http://www.cat.cc.md.us/biotutorials/proteins/protein.html

Torsion Angles



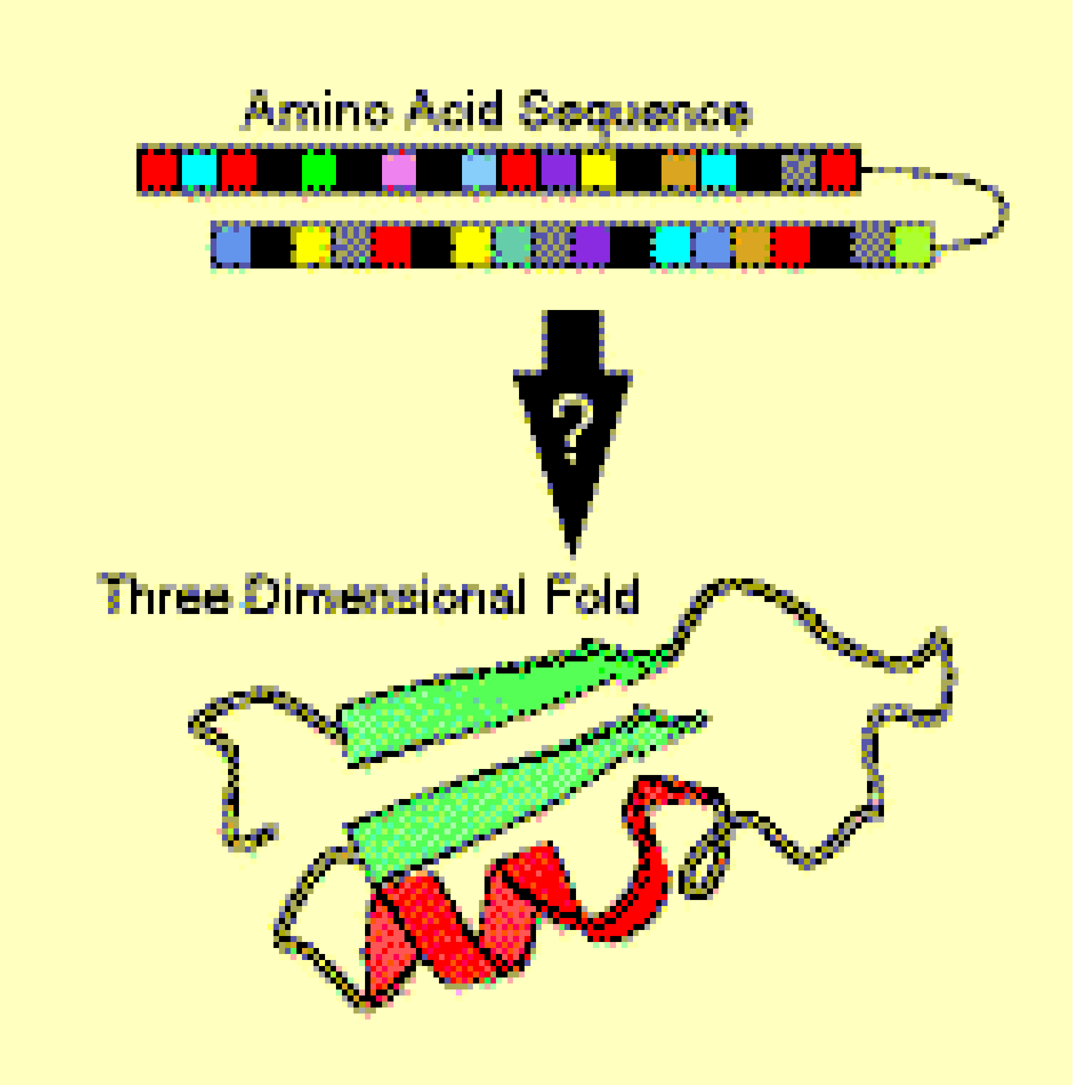
Molecular Biology of the Cell, Alberts, et al

Allowed Regions



Introduction to Protein Structure, Branden and Tooze

What is Protein Folding?

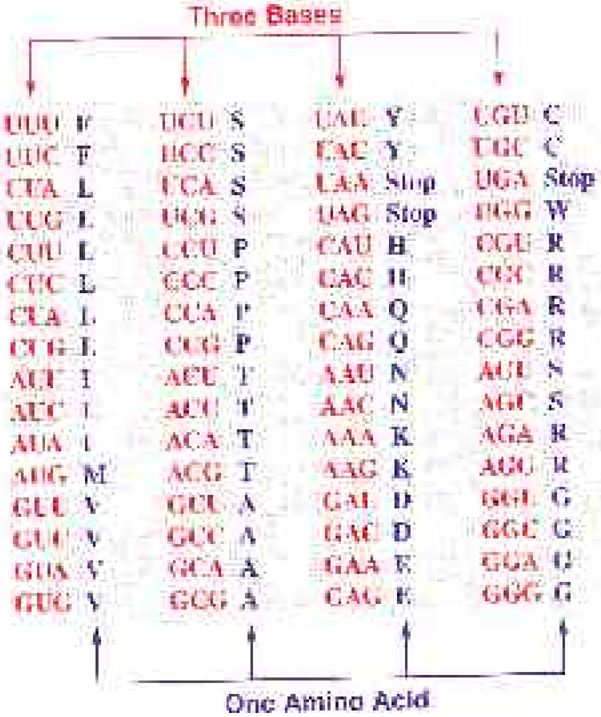


http://biop.ox.ac.uk/www/mol_of_life/index_c.htm

What is Protein Folding?

A Sequence of Bases in DNA...

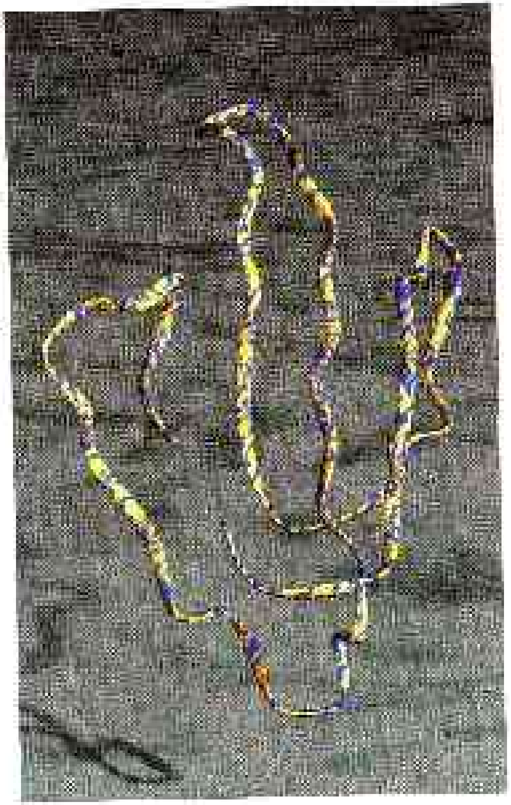
Is Translated to a Sequence of Amino Acids in a Protein...


Three Bases
One Amino Acid

UUU	F	UCU	S	UAU	Y	UGU	C
UUC	F	UCC	S	UAC	Y	UGC	C
UUA	L	UCA	S	UAA	Stop	UGA	Stop
UUG	L	UCG	S	UAG	Stop	UGG	W
CUU	L	CCU	P	CAU	H	CGU	R
CUC	L	CCC	P	CAC	H	CGC	R
CUA	L	CCA	P	CAA	Q	CGA	R
CUG	L	CCG	P	CAG	Q	CGG	R
AUU	I	ACU	T	AAU	N	AGU	S
AUC	I	ACC	T	AAC	N	AGC	S
AUA	I	ACA	T	AAA	K	AGA	R
AUG	M	ACG	T	AAG	K	AGG	R
GUU	V	GCU	A	GAU	D	GGU	G
GUC	V	GCC	A	GAC	D	GGC	G
GUA	V	GCA	A	GAA	E	GGA	G
GUG	V	GCG	A	GAG	E	GGG	G

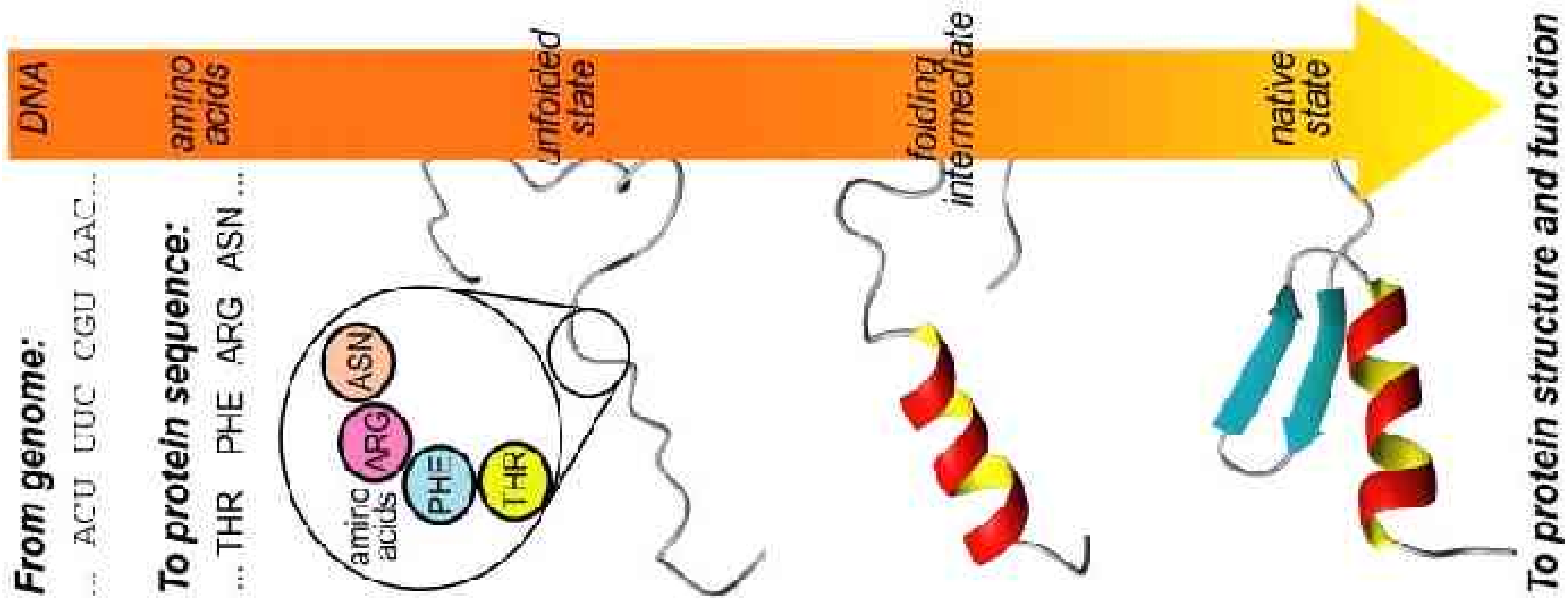
Genetic Code
'Translation Table'

Which Folds Spontaneously to a Precise Three-Dimensional Structure



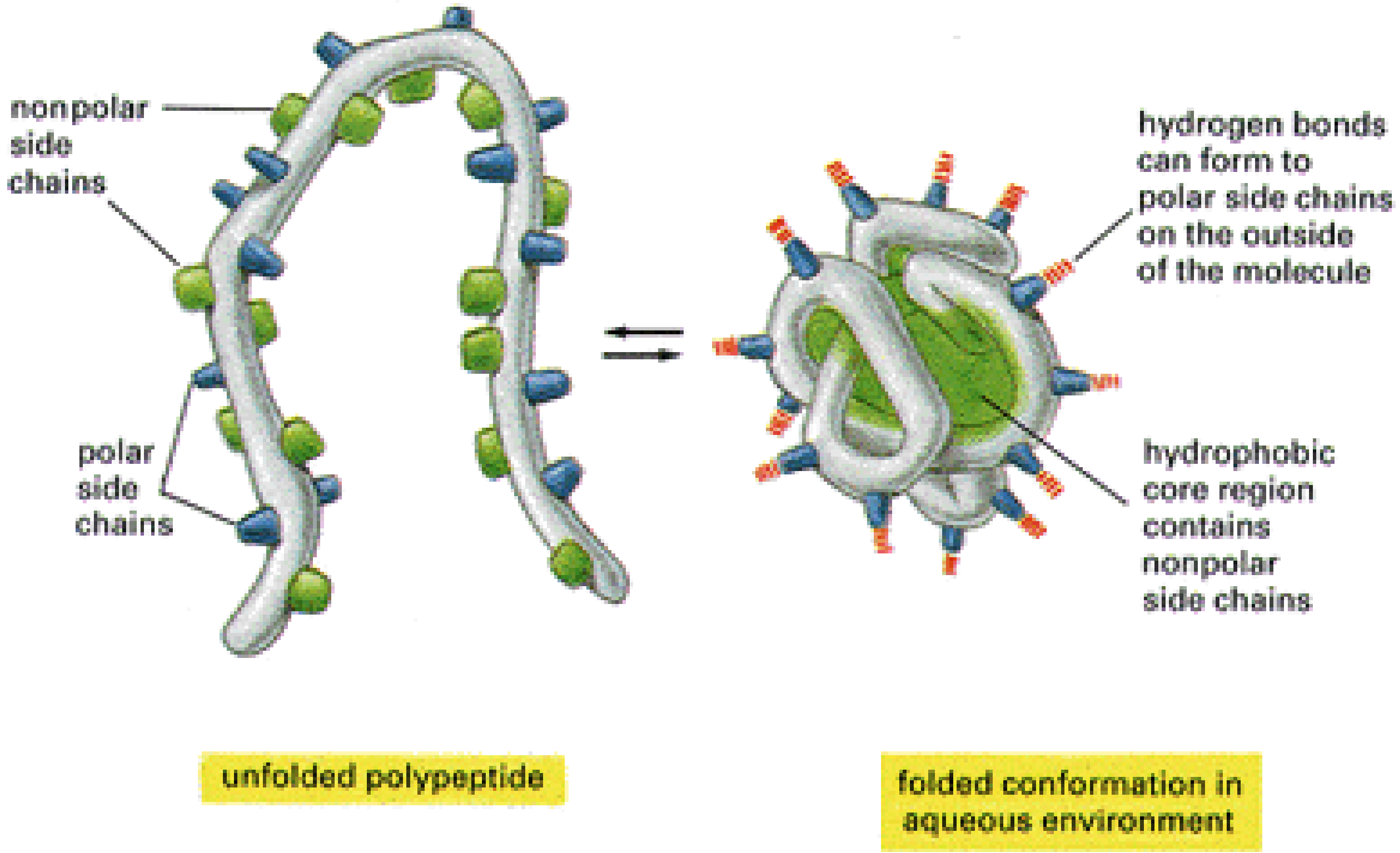
Introduction to Protein Architecture, A.M. Lesk

What is Protein Folding?



http://www.stanford.edu/group/pandegroup/Cosm/

What is Protein Folding?


nonpolar side chains
polar side chains
hydrogen bonds can form to polar side chains on the outside of the molecule
hydrophobic core region contains nonpolar side chains
unfolded polypeptide
folded conformation in aqueous environment

Molecular Biology of the Cell, Alberts, et al

What is Protein Folding?

A fully extended backbone of a typical protein is of the order of 1000s Angstrom ; the largest dimension of the tertiary structure is less than 100 Angstrom.

Yet this `knotted ball' is functional with some order to it. How does the protein fold?

Levinthal Paradox

Many Naturally occurring proteins fold reliably and quickly to their native state despite the astronomical number of possible configurations.

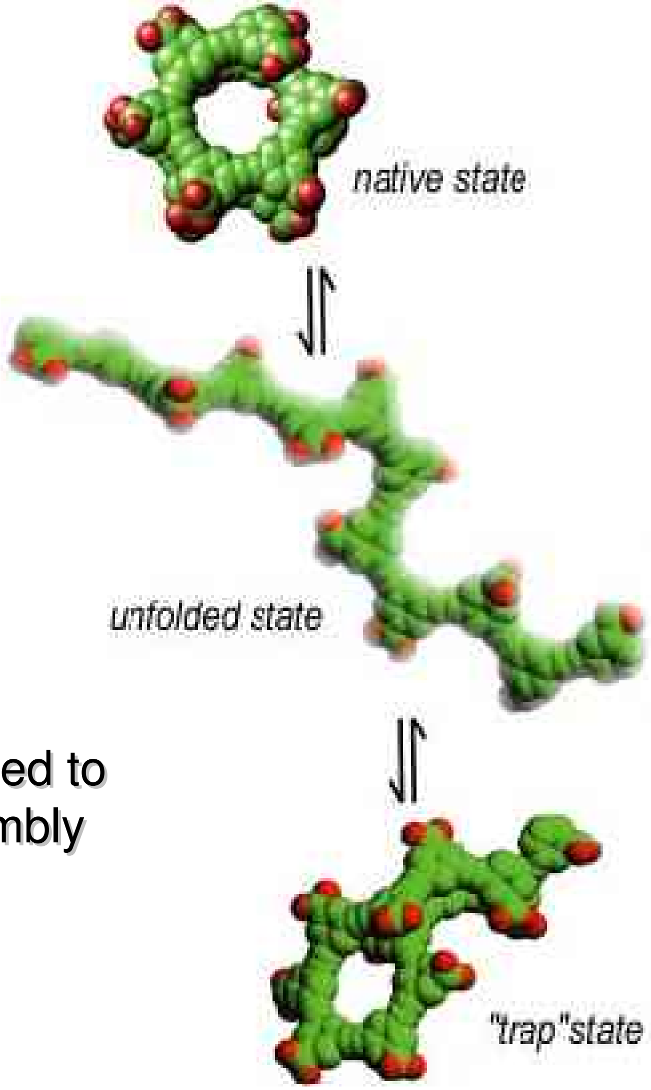
Hence, proteins have to fold through some directed process.

What is Protein Folding?

Anfinsen (*Science*, **181** (1973) p223)

showed that the only information required for the protein sequence to fold correctly is the sequence itself

Wrong Folding



A synthetic polymer designed to have protein-like self-assembly

http://www.stanford.edu/group/pandegroup/Cosm/

Wrong Folding

- Prion protein an infectious agent and self-replicating
- Responsible for mad cow and its equivalent in human and sheep
- Its aggregation damages nerve cells in mad cow disease is constantly being produced by the body.
- Normally, it folds properly, remains soluble, and is disposed of without problem.

Robertson et al (2002)

Three kinds of Studies

- Structure Prediction
- Folding Pathway Characterization
- Folding Kinetics

Protein Structure Prediction

- Sequence and structural analysis may help to understand folding pathways and energetics
- Improved understanding of folding kinetics and stability may help in the design of prediction algorithms

Ref: B.Honig JMB (1999) 293,pp283-93

Open Problem

There is no reliable procedure which begins with homologous model and then relaxes the structure using MD to yield a conformation close to native. This is an important problem where database analysis cannot help.

Why Structure Prediction ?

- Protein Structure prediction is one of the most challenging areas of research for the structural biologist.
- For 100 residues 4^{100} (or 10^{60}) possible conformations 10 years to search the whole space assuming 1 nanosecond Per energy calculation (Levinthal: *T. Creighton (editor), Protein Folding, W. H. Freeman (1992)*

Number of protein sequences

Family Collection Name	No. of sequences (Domains)
COGs	332
Pfam	169524
PIRALN	40739
ProClass	54535
ProDom	15454
ProtoMap	31119
PSSP	148377
Systers	80201

Why Structure Prediction ?

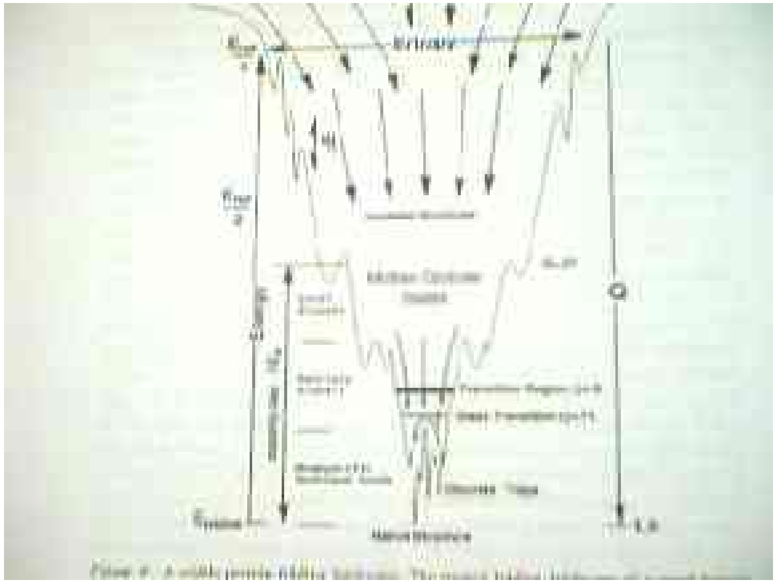
- Out of few lakhs protein sequences only few thousands have known structures (X-ray, NMR)
- At the current rate it may take 500 years to solve all protein structures

Why Structure Prediction?

- Some proteins cannot be crystallized easily
- About 10% of protein sequences exhibit unrelated and unidentified folds
- It’s important to evolve the structures without the help of databases
- Useful in modelling the differences between structures that databases may not facilitate
- Provide good starting point for molecular dynamics simulations

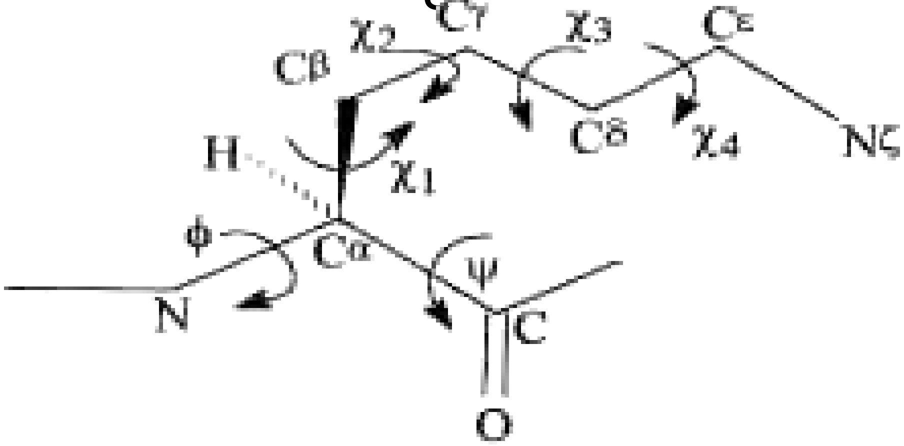
Energy Landscape

- Large ensemble of states for unfolded protein and far fewer to folded protein
- A funnel shaped landscape, a decrease in energy and concomitant loss of entropy with increasing structure; like simple lattice models, (Ann Rev Phys Chem (1997) 48, pp545-600)

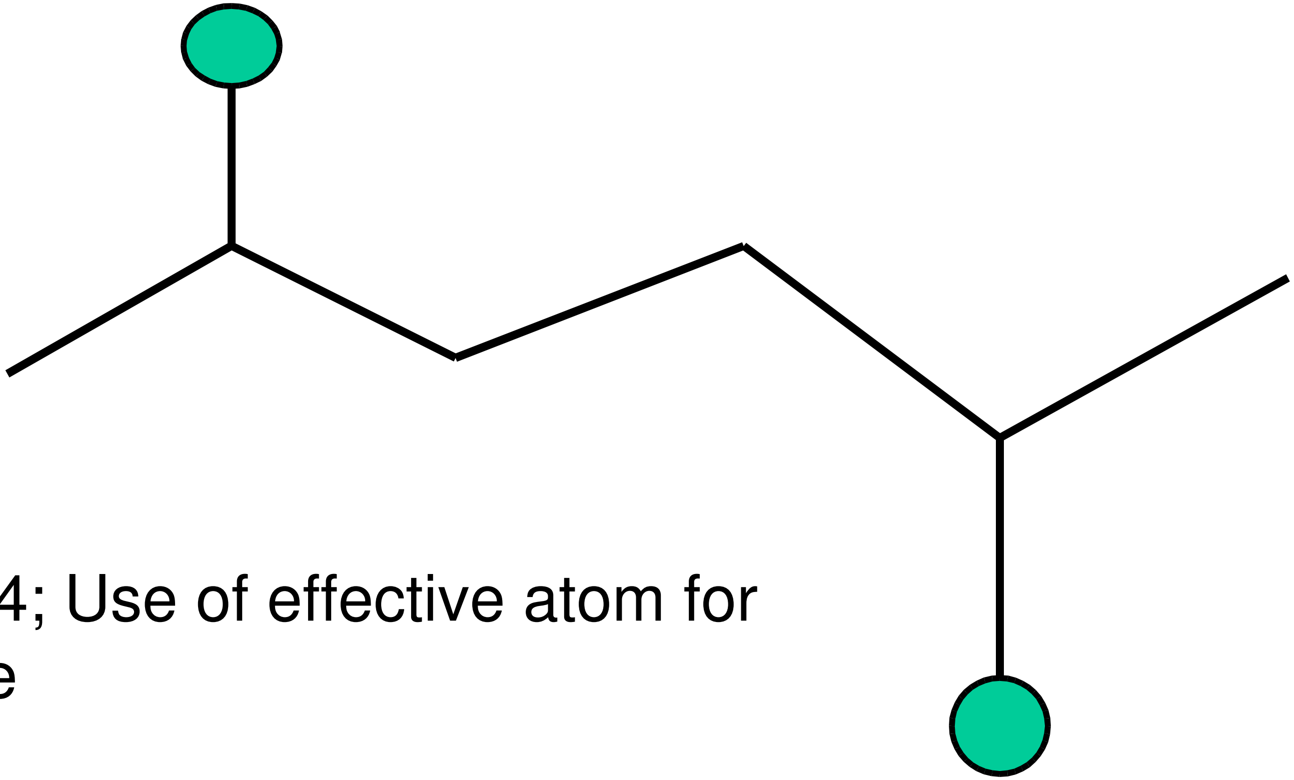


Model of Protein Structure

- Internal coordinate system restricted to torsion angle variation
- Energy = non-bond + torsion angle interactions (AMBER force field)
- Assumptions
  - Fix bond lengths, bond angles
  - Also fix omega and side-chain torsion angles



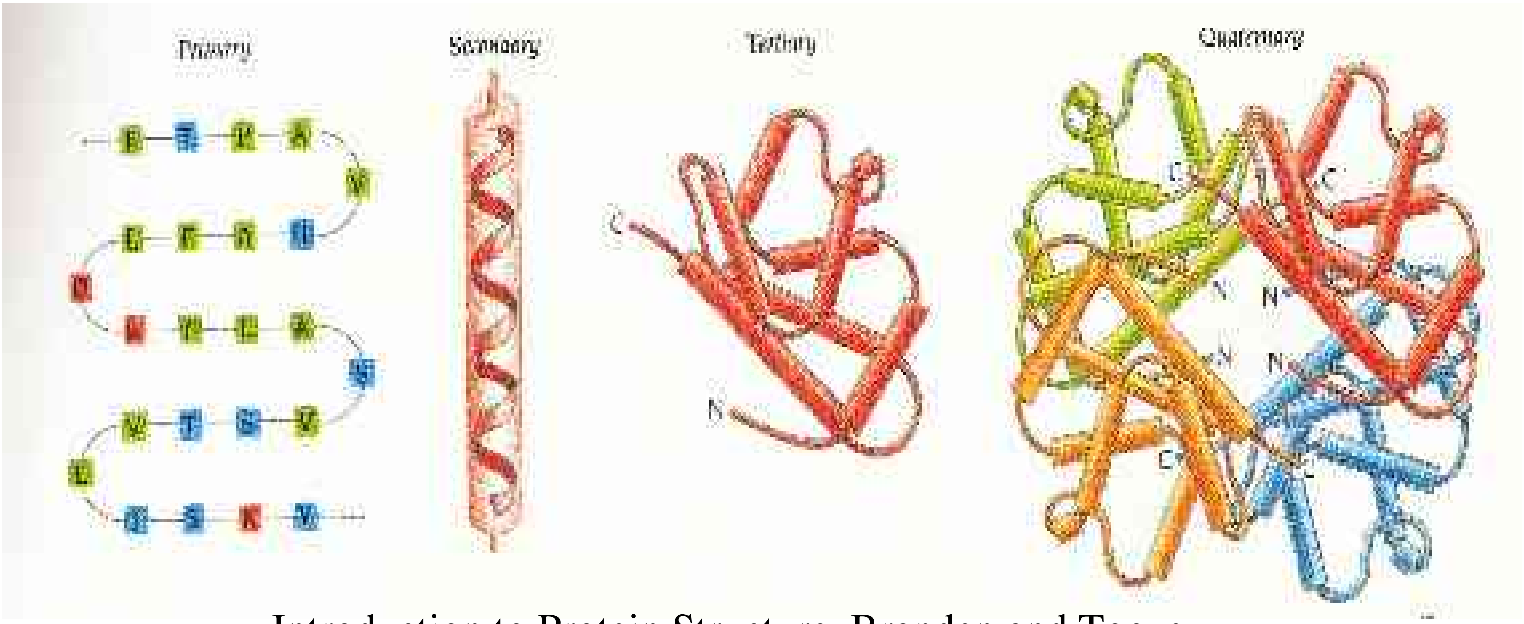
Effective atom



Sun 1994; Use of effective atom for a residue

Hierarchical Pathway

- Formation of local secondary structures
- Interact to produce larger structural fragments
- Undergo further assembly to yield native conformation



Introduction to Protein Structure, Branden and Tooze

Library of fragments

Bowie and Eisenberg (1994) Proc. Natl Acad Sci. 91, p4436

Choose fragments based on the amino acid profile environments and optimally assemble them

Genetic algorithms

Nature’s Genetics

The structure and behaviour of each individual is controlled by a set of instructions called *genes* written in a four letter alphabet on long strands of DNA.

The process of species changing over time to become better at survival within the environment is called *evolution.*

Reproduction
mutation

Genetic algorithms

Evolution is the aggregation of thousands of semi-random events and the natural pressure to reproduce or die.

Rabbit as an example:

Given a population, smarter and faster ones are less likely to be eaten by foxes. So, they survive to do what rabbits do best: make more rabbits.

Some of the slower and dumber rabbits also survive by luck. The surviving population starts breeding resulting in a good mixture of rabbit genetic material:

Genetic algorithms …

some *slow* rabbits breed with *fast* rabbits,
some *fast* rabbits breed with *fast* rabbits,
some *smart* rabbits breed with *dumb* rabbits, ...on.

And on top of that, nature throws in a *wild hare* every once in a while, by mutating some of the rabbit genetic material.

The resulting rabbits (on an average) are faster and smarter than those in the original population because more faster, smarter parents survived the foxes.

(It is a good thing that the foxes are also undergoing similar process- otherwise, the rabbits might become too fast and smart for the foxes to catch any of them).

GA : Definition 1

- GAs are general purpose search algorithms which use principles inspired by natural genetic evolution and selection
- Basic idea is to maintain a population of chromosomes, that evolves over time through a process of competition & controlled variation

GA : Definition 2

- Is a method of simulating the action of evolution within a computer. A *population* of fixed-length strings is evolved with a GA by employing *crossover* and *mutation* operators along with a *fitness function* that determines how likely individuals are to reproduce. GAs perform a type of search in a fitness landscape

Structure & Design Procedures

- Terminology
- An overview
- General Mechanism
- GA as a Procedure
- Representation
- Selection
- Crossover
- mutation

GA Terminology

- Generation successively created populations.GA iterations
- Population set/pool of trial solutions/individuals/chromosomes exhibiting similar gene structure
- Chromosome coded form of a trial solution vector/string consisting of genes made of alleles
- Parent member of the current generation
- Child/offspring member of the next generation
- Fitness a number assigned to an individual representing a measure of goodness
- Natural Selection filtering process by which individuals with higher fitness are more likely to reproduce
- Crossover where generally two parents produce two offspring by gene exchange
- Mutation random change of the value of a gene

An Overview

- A GA starts with a population of randomly generated chromosomes and advances towards better chromosomes by applying genetic operators
- The population undergoes evolution in the form of natural selection. During successive generations chromosomes in the population are rated for their adaptation as solutions

Cont.

An overview [2]

- Based on these evaluations (fitness values) a new population of chromosomes is formed using a selection mechanism, crossover & mutation operators

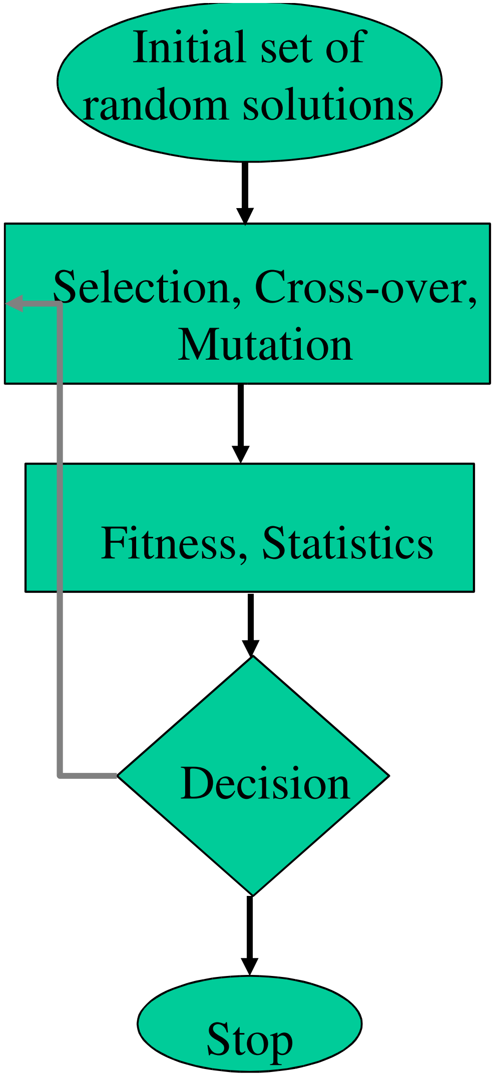
General Mechanism

- Define a Genetic representation of the problem
- Creation of an initial population(Generated Randomly)
- Evaluation of individual fitness
- Formation of an Intermediate Population through selection mechanism
- Recombination through Crossover & Mutation operators

GA as a Procedure

```
Function GeneticAlgo ()
{
    popNum = 0 ;
    initialize  Population (popNum) ;
    evaluate Population (popNum) ;
    while (NOT termination-condition)
    {
          popNum ++ ;
          select Population(popNum) from Population(popNum-1) ;
          recombine Population(popNum) ;
          evaluate Population(popNum) ;
    }
}
```

GA as a Procedure [2]


Initial set of random solutions
Selection, Cross-over, Mutation
Fitness, Statistics
Decision
Stop

Iterate

Termination Conditions:

•Generation Number
• Evolution Time
• Fitness Convergence
• Population Convergence
• Gene Convergence

Schema Theorem

Definitions

Characters { 1, 0, * }

Schema is a similarity template

schema S $* \ \underline{1} * \underline{0} * \underline{0}\ \underline{0} * *$

0 1 1 0 1 0 0 0 1

1 1 0 0 0 0 0 1 0

1 1 1 0 1 0 0 1 1

Definite bits

δ(s) - Defining length (= 5)

Distance between the leftmost and the rightmost definite bits

Schema Theorem

O(s) - Order of the Schema (= 4)
Number of definite bits.

schema S $* \ \underline{1} \ * \ \underline{0} \ * \ \underline{0} \ \underline{0} \ * \ *$

0 1 1 0 1 0 0 0 1
1 1 0 0 0 0 0 1 0
1 1 1 0 1 0 0 1 1

l - length of the Schema
Pc - probability of crossover
Pm - probability of mutation

Schema Theorem

Population *m(s,t) of SCHEMA ‘S’ at time ‘t’*

SELECTION

at time t+1

$$m(s, t+1) = m(s,t)\ f(s)/\bar{f}$$

CROSS OVER

Survival probability $P_s \geq 1 - P_c\, \delta(s)/ (l - 1)$

$$m(s,t+1) \geq m(s,t)(\ f(s)/\bar{f}\ )\ [1 - P_c .\ \delta(s)/ (l - 1)]$$

MUTATION

$$P_s = 1 - O(s)\ P_m$$

$$m(s,t+1) \geq m(s,t)\ (\ f(s)/\bar{f}\ )[1 - P_c\, \delta(s)/ (l - 1\ ) - O(s)\ P_m\ ]$$

Schema Theorem

Short, low order, above average Schemata receive exponentially increasing trials in subsequent generations

By iterating selection, cross-over and mutation, overall fitness improves and the individual would represent improved solution of the problem posed through fitness.

Representation Issues

- Representation of the problem(chromosome) is a key issue and the representation schema can severely limit the search space
- The encoding mechanism depends on the nature of the variables and the problem
- Examples:

Fixed length strings, Binary coded strings, Vectors of floating point numbers

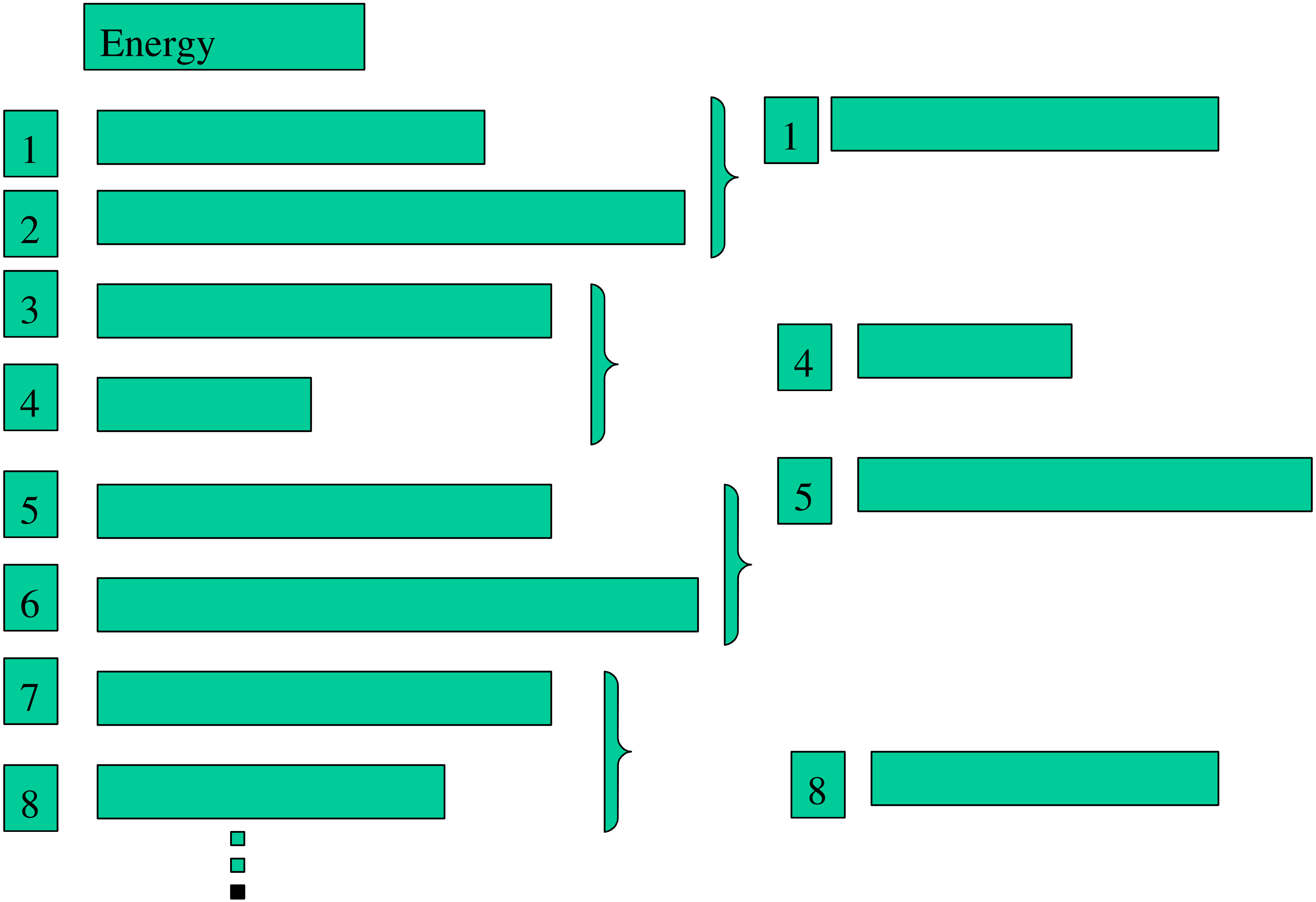
Binary string:

1	0	0	1	0	1	1	0	1	0

Selection Mechanism

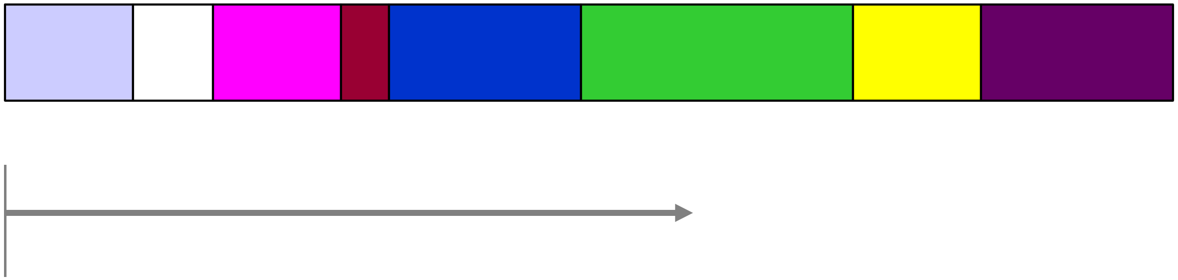
- Given a population P, the selection mechanism produces an intermediate population P’ with copies of chromosomes of P
- The number of copies of each chromosome depends on its fitness
- Examples of selection procedures are:
  - Tournament selection
  - Roulette wheel selection

Tournament Selection



Roulette Wheel

$\text{Sum} = F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_7 + F_8$



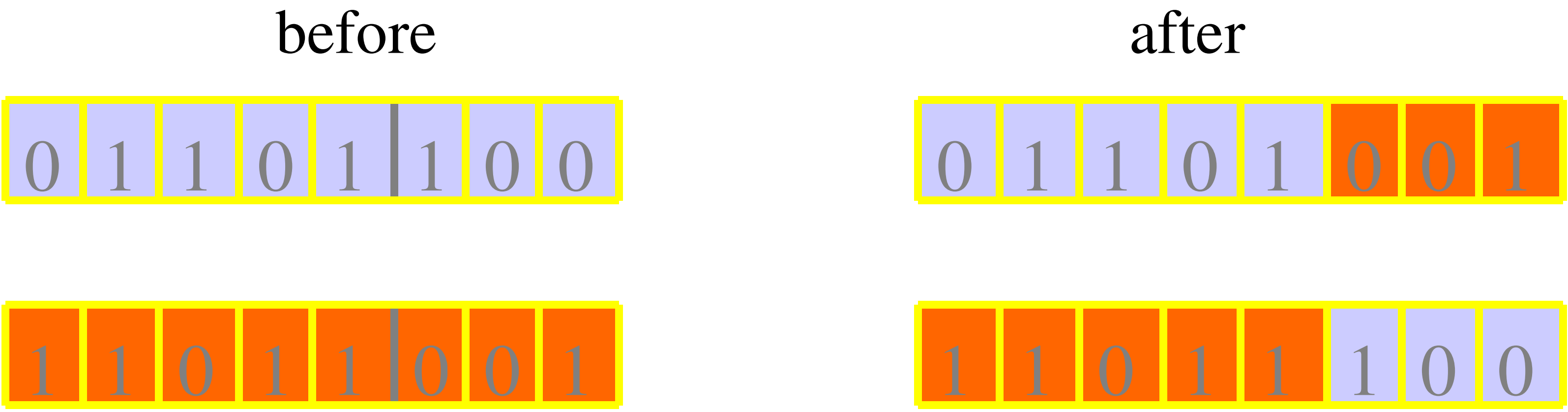
r

$(F_1+F_2+F_3+F_4+F_5)/\text{Sum} < r < (F_1+F_2+F_3+F_4+F_5+F_6)/\text{Sum}$

Recombination : Crossover

- Combines the features of two parent chromosomes to form two offspring, with the possibility that good chromosomes generate better ones
- Crossover is applied to a random choice of chromosomes and the likelihood of crossover being applied depends upon the crossover probability

Crossover [2]



before

0 1 1 0 1 | 1 0 0

1 1 0 1 1 | 0 0 1

after

0 1 1 0 1 0 0 1

1 1 0 1 1 1 0 0

$r < \mathcal{P}_c$

Types of Crossovers:

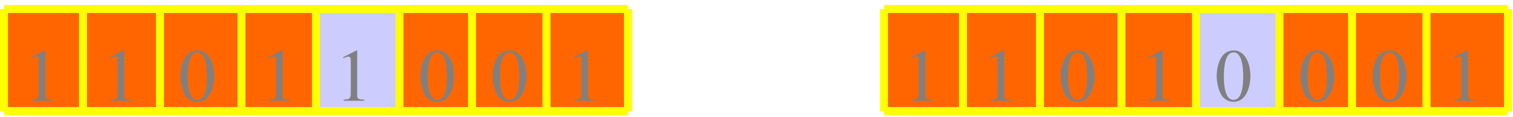
- One Point
- Two Point
- Uniform
- Arithmetic
- Heuristic

Recombination : Mutation

- The purpose of Mutation operator is to increase the structural variability of the population
- Mutation ensures that the probability of reaching any point in the search space is never zero
- It arbitrarily alters one or more components of a selected chromosome
- Mutation occurs with a user defined mutation probability

Mutation [2]

before after



1 1 0 1 1 0 0 1

1 1 0 1 0 0 0 1

$r < \mathcal{P}_m$

Types of Mutations:

- Flip Bit
- Boundary
- Uniform
- Gaussian

Building Block Hypothesis

Short, low-order and highly fit schemata (called building blocks) has reduced the complexity of the problem; instead of building high-performance strings by trying every conceivable combination, building blocks help in constructing better and better strings from the best partial solutions of past samplings.

A trivial example

To find the square root or solve $x^2=64$ (Dr. Dobb's journal)

- Consider 5 bit strings for x
- Randomly construct initial population
- Minimise $|64-x^2|$

🟩 *0*

🟥 *1*

		$|64-x^2|$
	🟩🟥🟩🟥🟥	**57**
{	🟥🟩🟩🟥🟩	**260**
	🟩🟥🟥🟩🟥	**105**
{	🟩🟩🟥🟥🟩	**28**

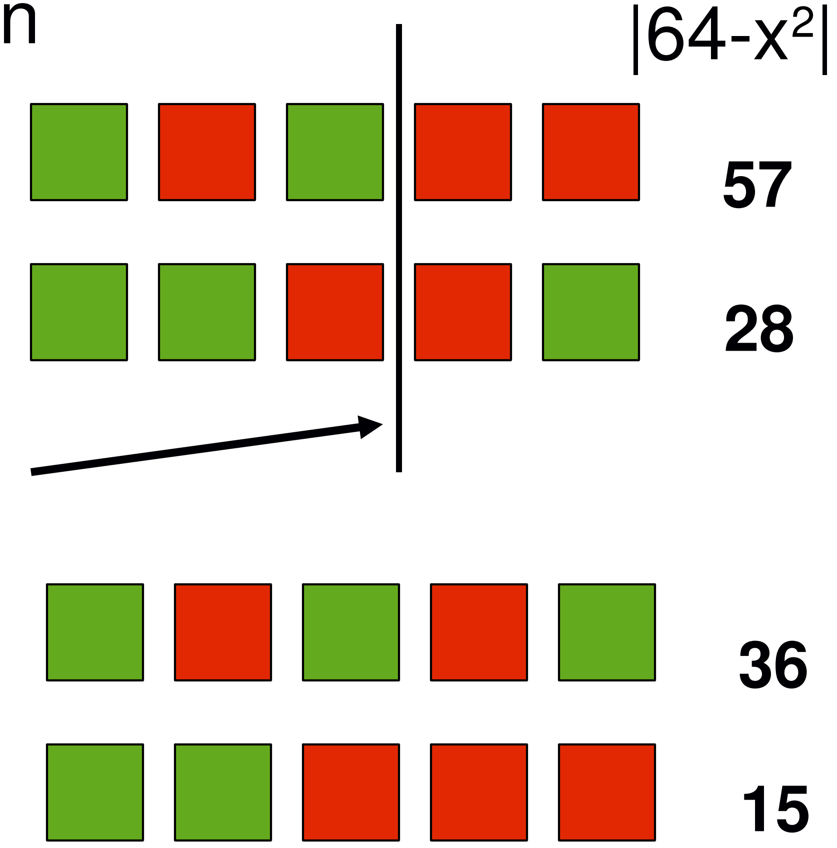
Assume a hypothetical situation

After Selection

r < P$_c$ (~ 0.6)

Random crossover site

After Crossover



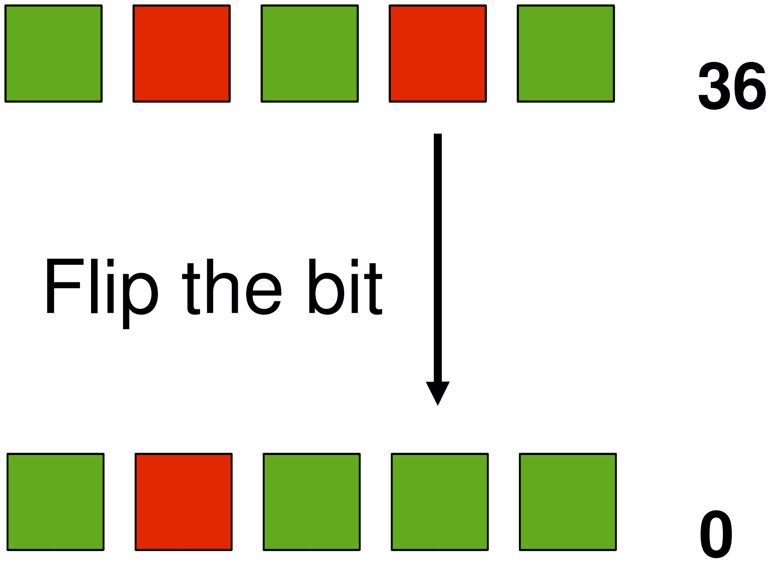
Assume a hypothetical situation

For each bit check

$r < P_m$ (~ 0.05)

Mutation at 4th bit

After Mutation



🟩🟥🟩🟥🟩 36

Flip the bit

🟩🟥🟩🟩🟩 0

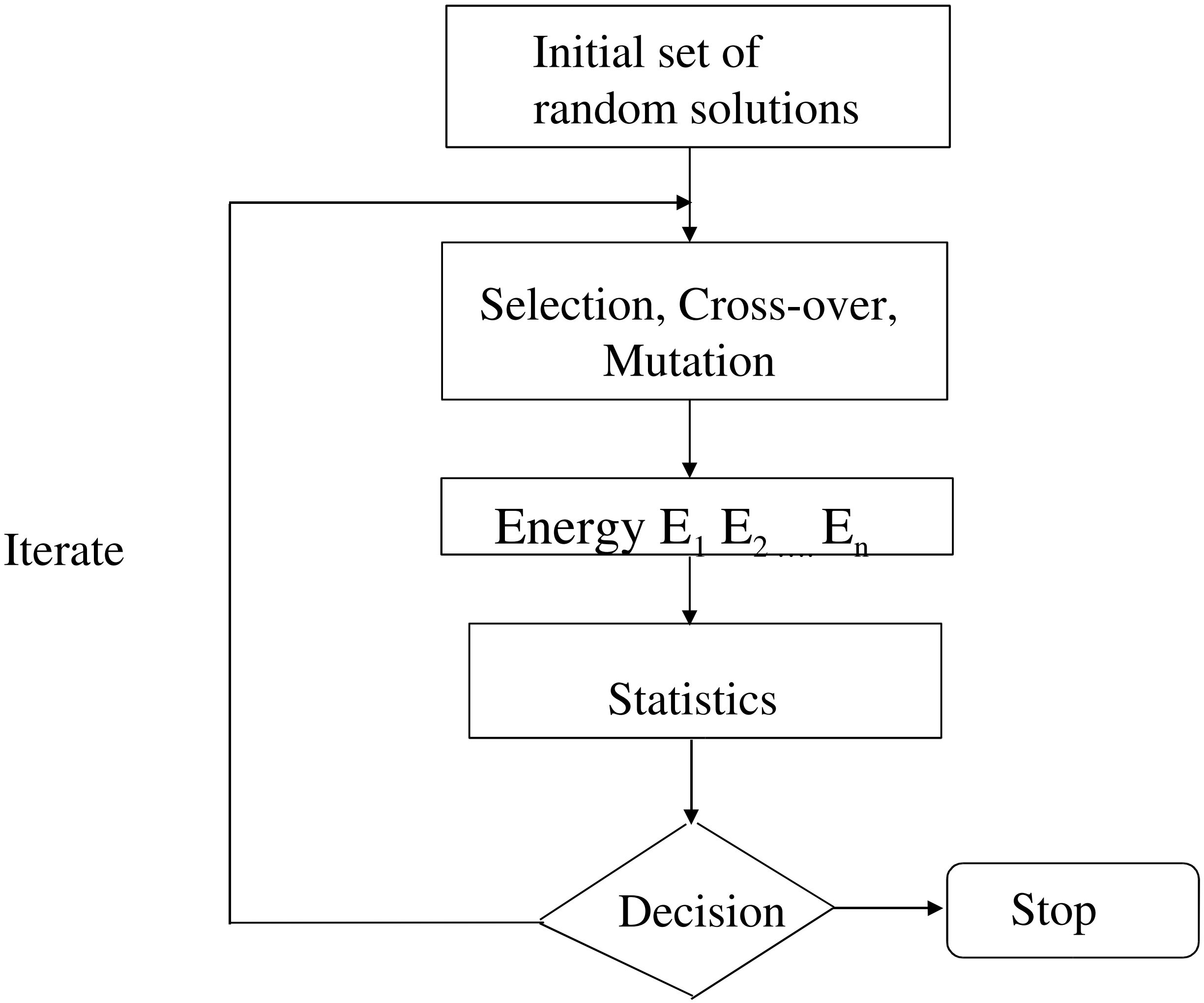
Solution of $x^2=64$ ⇒ **x=8**

Min { $|64-x^2|$ }

Advantages of GAs

- Flexibility, robustness with global search characteristics, much less likely to get stuck on local optima
- Easy to implement
- Search from a population of points/solutions
- No derivatives, therefore can solve non-linear, discontinuous in parallel configuration
- Inherent parallelism
- Works on the representations rather than on the variables themselves

The Flow



Initial set of random solutions

Selection, Cross-over, Mutation

Energy E_1 E_2 .... E_n

Statistics

Decision

Stop

Iterate

Features

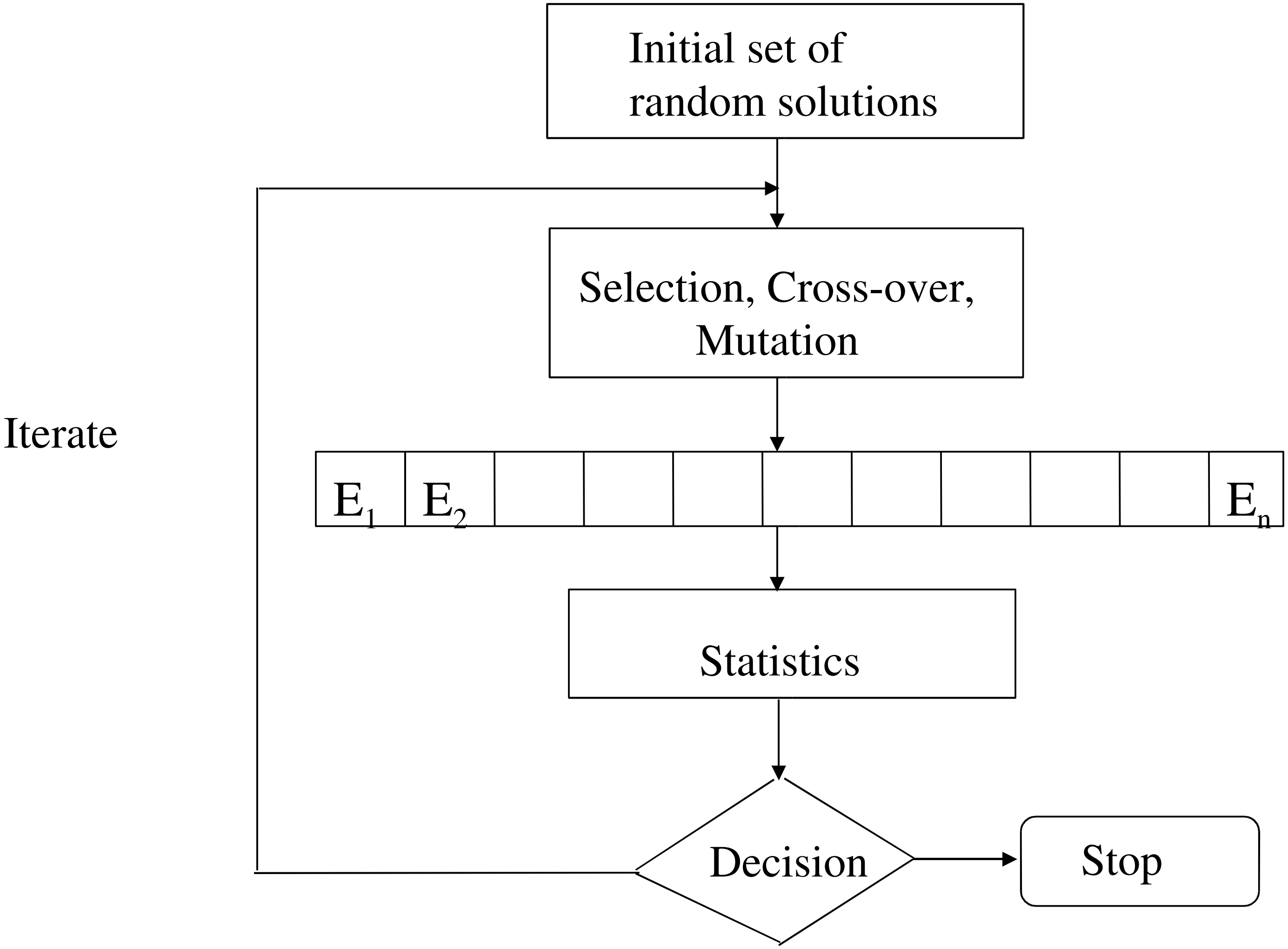
- A representation is chosen
- Start with a population of solutions
- Go through the iterative process of evolution
- Pick up the best possible solutions
- Inherent parallelism

“Why would one like to go through a long process of evolution “ A Critique of John Holland

Parallelism

- Data Parallel Model
- Island or Migration Model
- Fine Grain Model

Data Parallel Model



Initial set of random solutions

Selection, Cross-over, Mutation

E_1 E_2 E_n

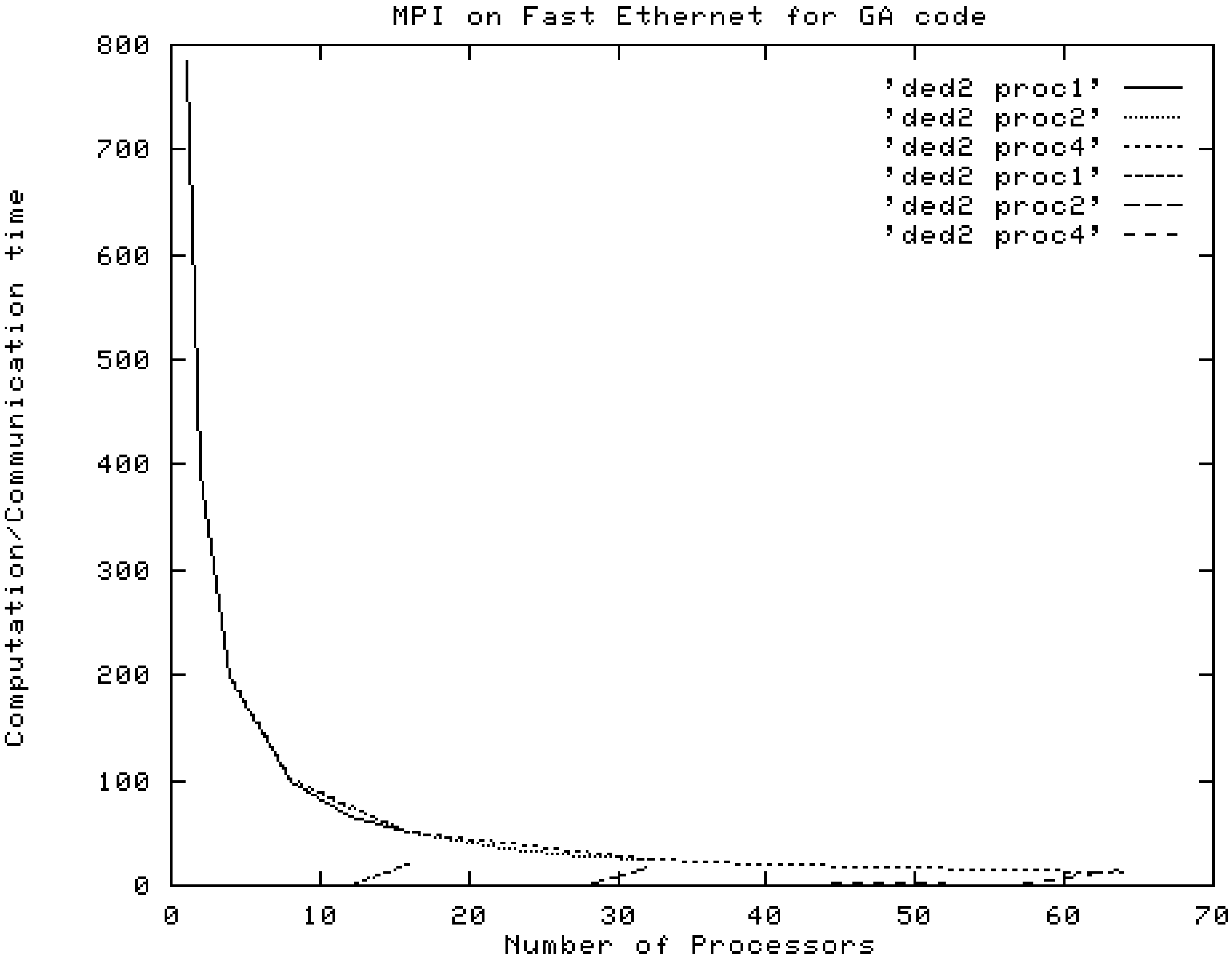
Statistics

Decision

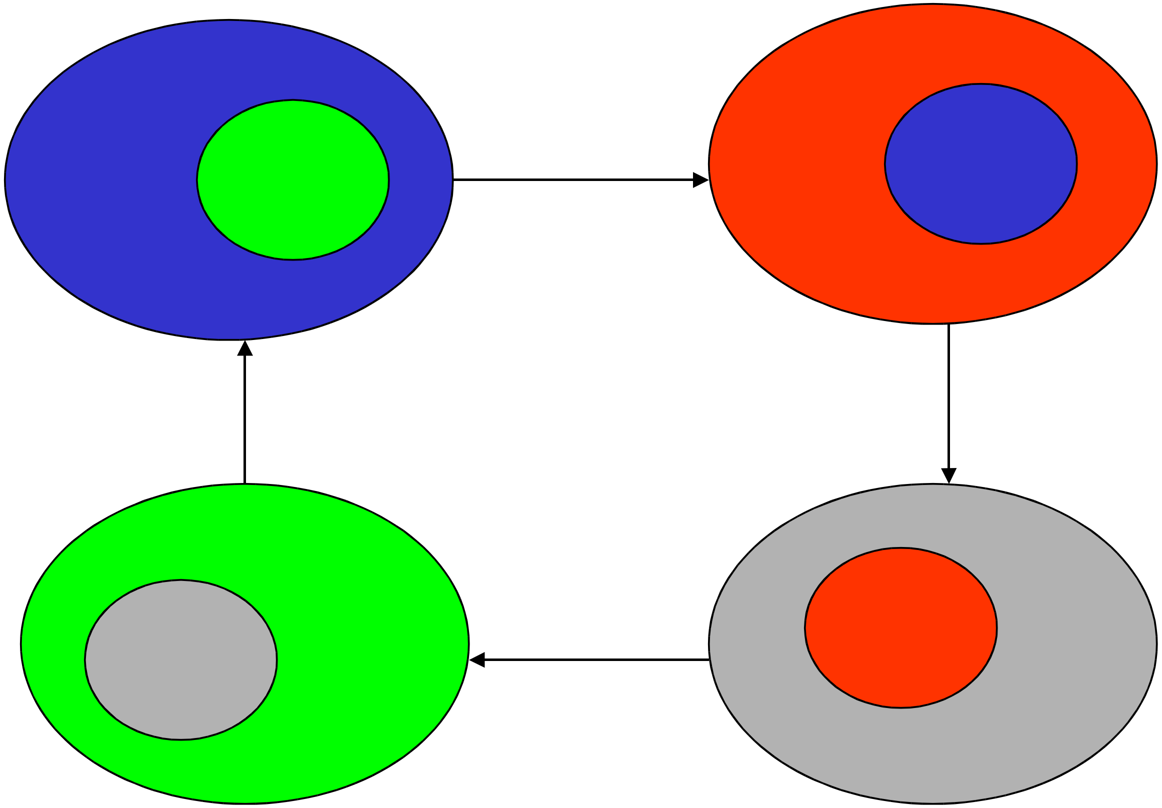
Stop

Iterate

Performance on PARAM 10000



Migrating GA Model

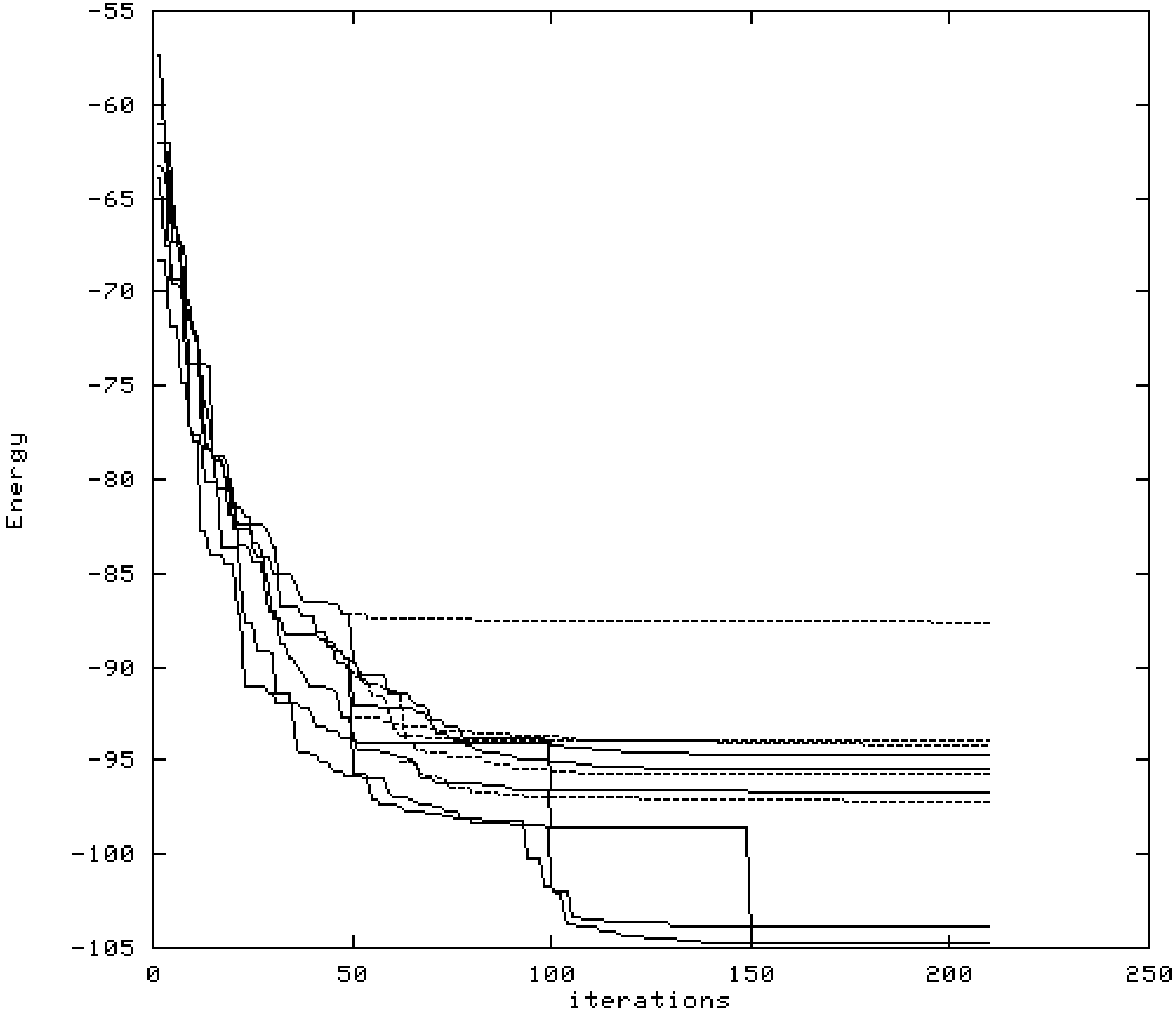


Migrating GA Model

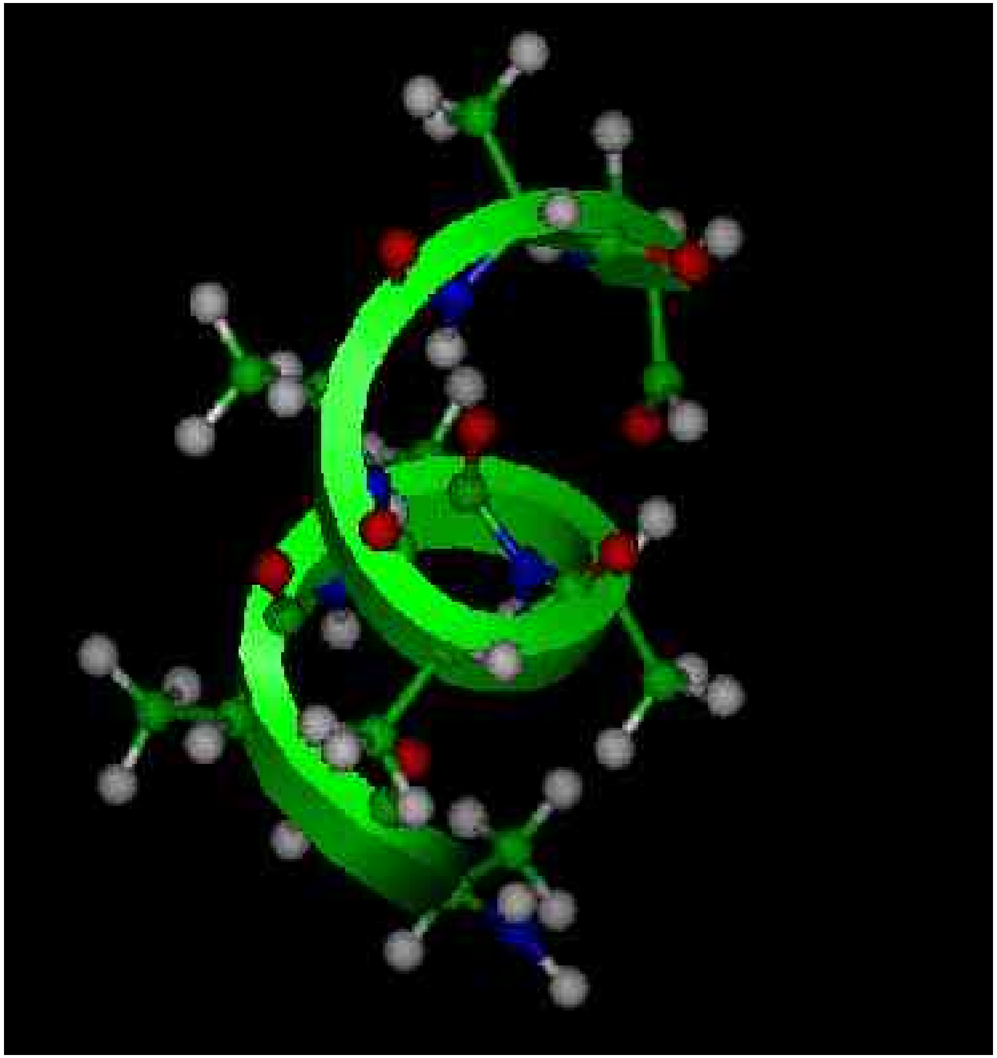
- Multiple runs
- Diversity (Additional Operator :**Migration**)
- Enhancement in efficiency
- Minimal Communication
- Ideal candidate for Parallel Computing

Octa Alanine

- Performance



Octa Alalnine



V, Sundararajan &
A.S. Kolaskar
Ed. S.S. Iyengar
Computer Modeling
And simulation of complex
Biological systems
CRC Press (1998)

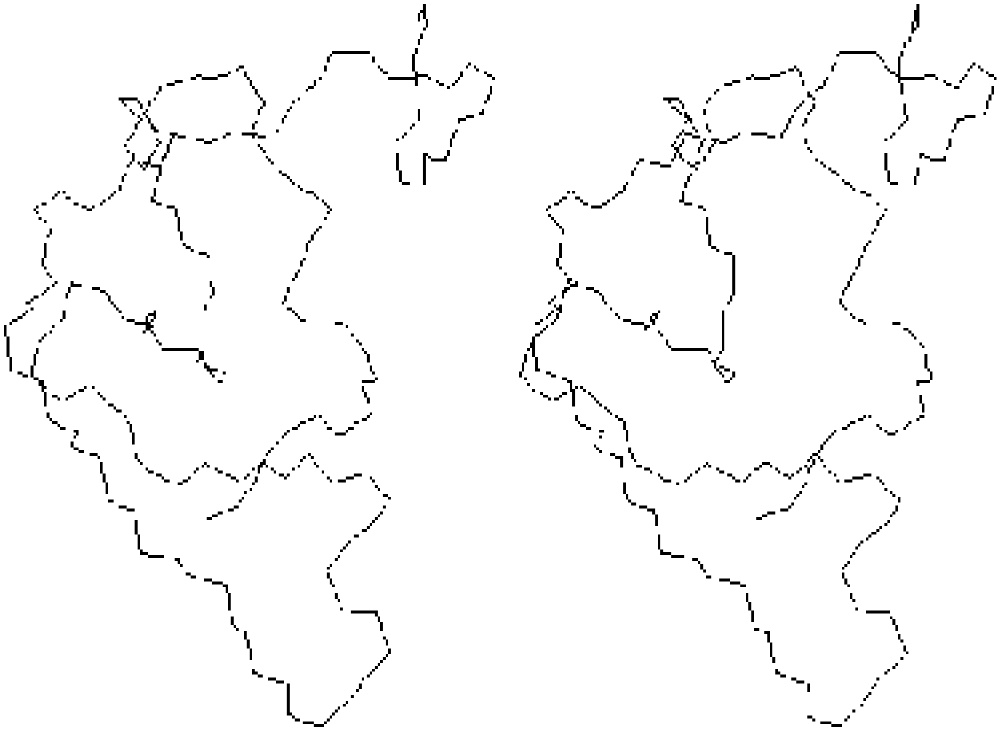
Crambin

- Crambin is a plant seed protein Containing 46 residues
- Real coded string was used
- Mutation is taken as a change to most probable torsion angles occurring in 129 proteins taken from PDB
- Added improper torsion energy and pseudoentropic terms
- varied the crossover and mutation probabilities across the iterations by equal steps

Schulze Kremer (1996)

Crambin

Individual	R.m.s.	Individual	R.m.s.
P1	10.07	P6	10.31
P2	9.74	P7	9.45
P3	9.15	P8	10.18
P4	10.14	P9	9.07
P5	9.95	P10	8.84



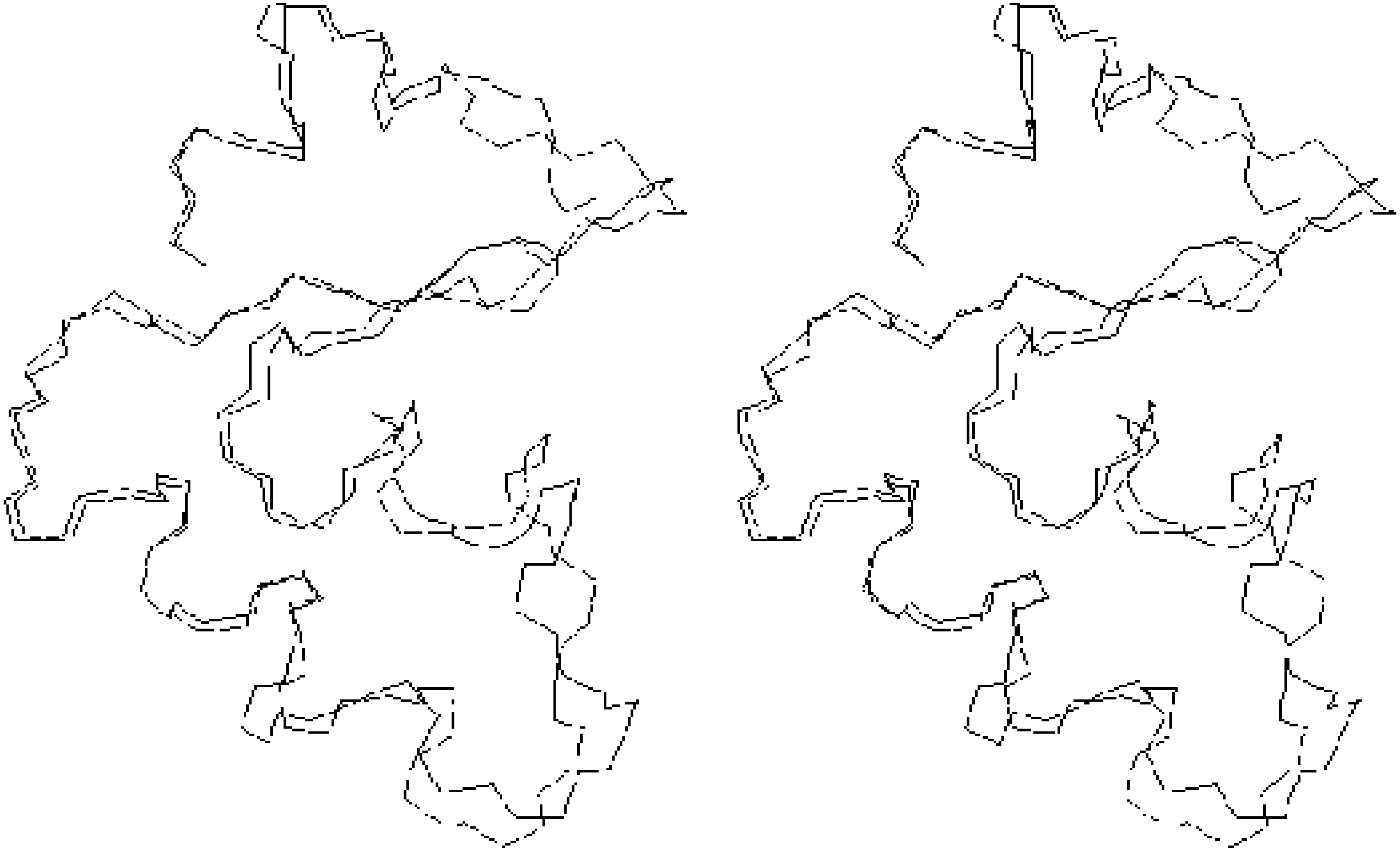
Schulze Kremer (1996)

Crambin

Multi-objective GA was used (hydro-phobicity, potential energy, RMS deviation etc)
to achieve 1.08 Angstrom accuracy

Schulze Kremer (2000)

Crambin

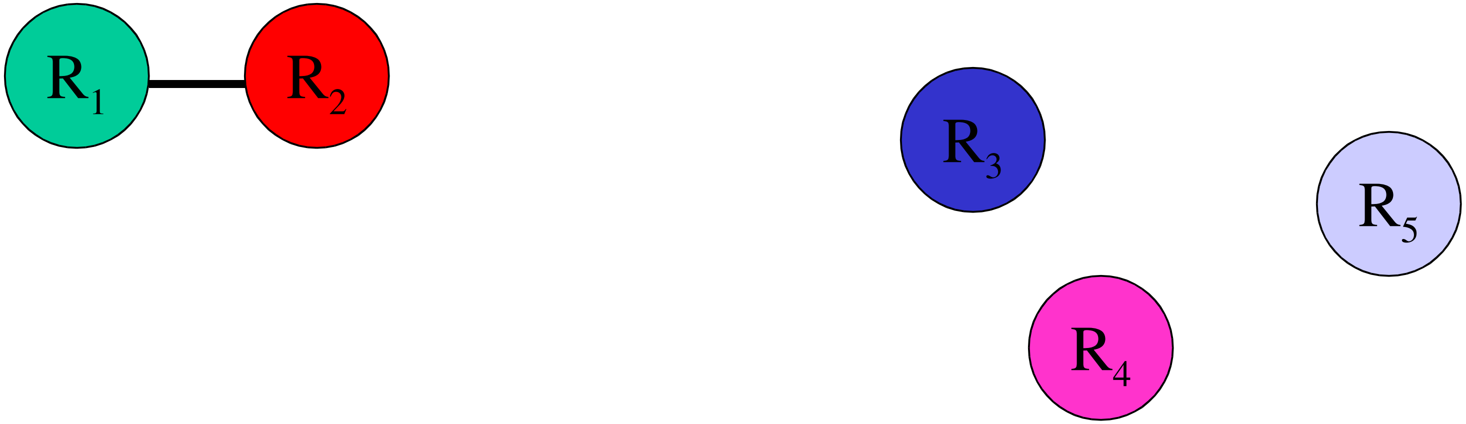


Schulze Kremer (2000)

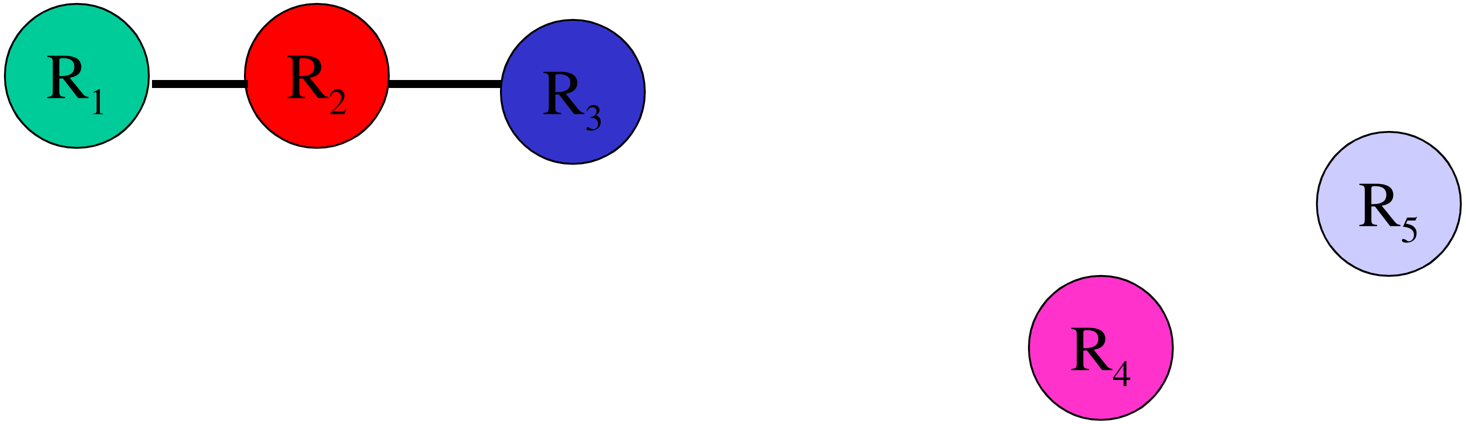
Hypothesis

Protein folding is an evolution process and it takes place with one or few amino acids added one after another.

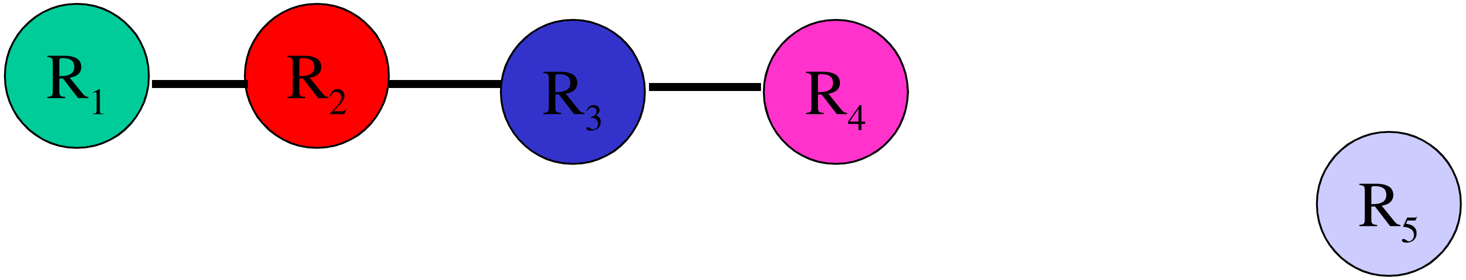
The Evolving Method



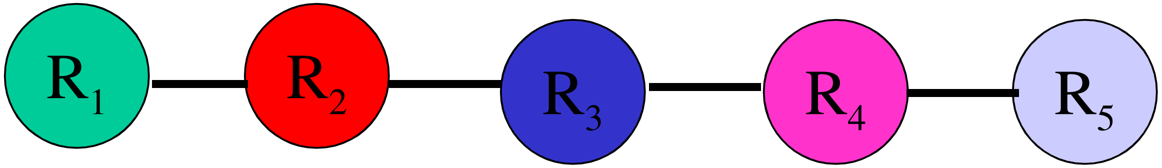
The Evolving Method



The Evolving Method



The Evolving Method



Strategy

Genetic Algorithms modified to evolve the protein structure by the addition of one or few amino acids at a time.

The Evolving Method

```
Program Evolve
    Inititialise the parameters (popsize,length, number of amino aicds, etc)
    Initial population for first few amino acids generated randomly
    Build the structures and evaluate energies
    Call Simple Genetic Algorithms
    Loop over number of cycles
         Add the  next few amino acids
         Generate the additional random structures to increase population
         Call Simple Genetic Algorithms
    End of the Loop over number of cycles
End Program Evolve
```

The Evolving Method

Schedule of Simulation:

Each cycle is no of iterations ~ population size

P_c 0.5 to 0.9

P_m 1 to 10 /(total no of bits in the population)

Shake applied if min of variance in angles < 1 and its breadth < 200

duplicates (>80% identity) replaced randomly or randomized

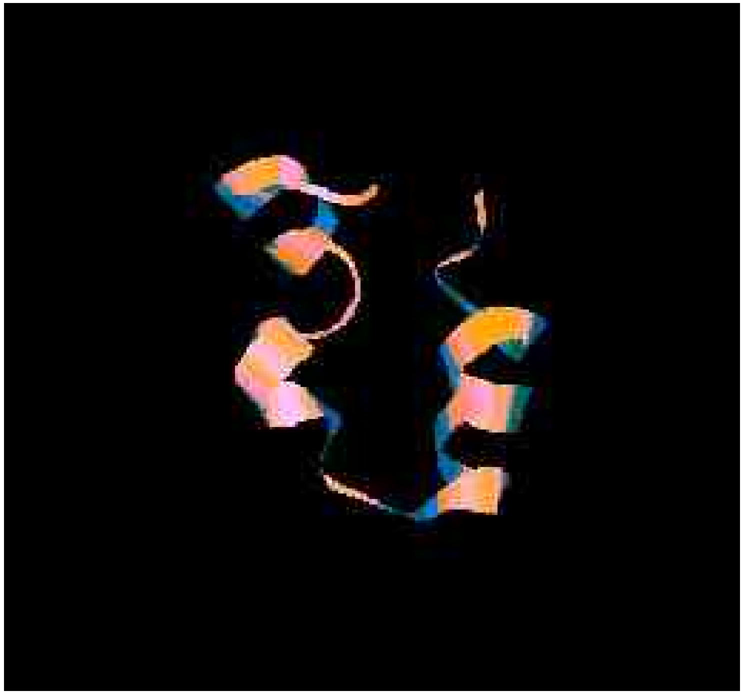
Last cycle performed 5*popsize iterations

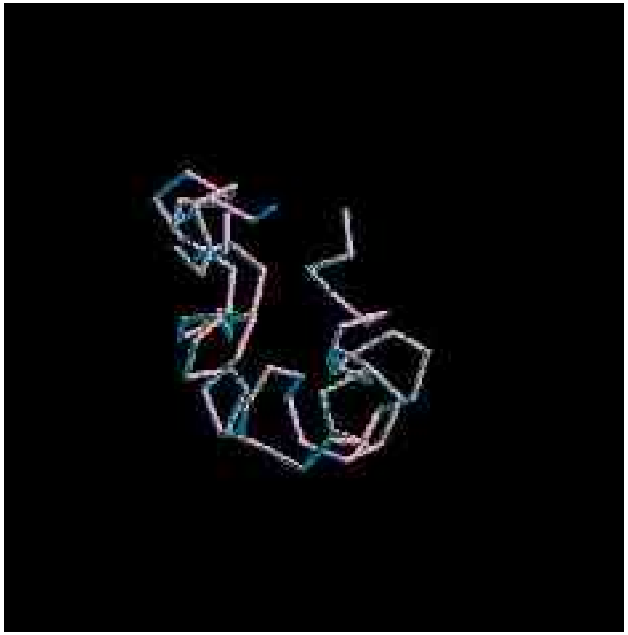
The Evolving Method

Advantages:

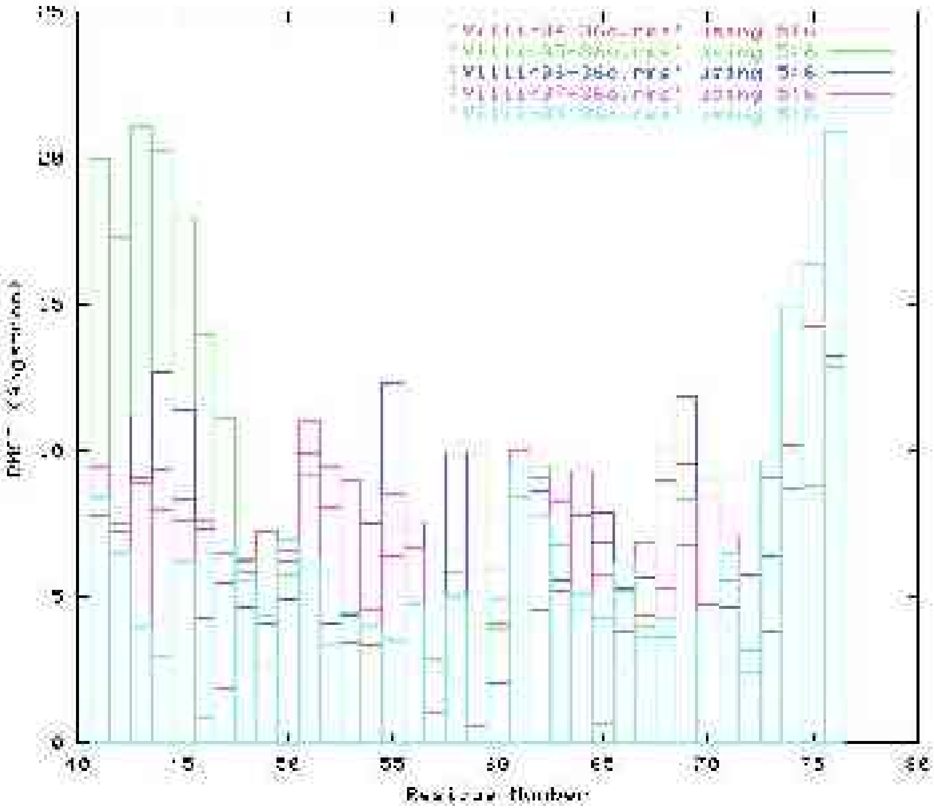
- It can mimic the natural evolution and easily adapted to an hypothesis
- More efficient than the sequential algorithms
- A number of variations in algorithms possible
  - Islands segmenting the molecule
  - Multiobjective optimisations

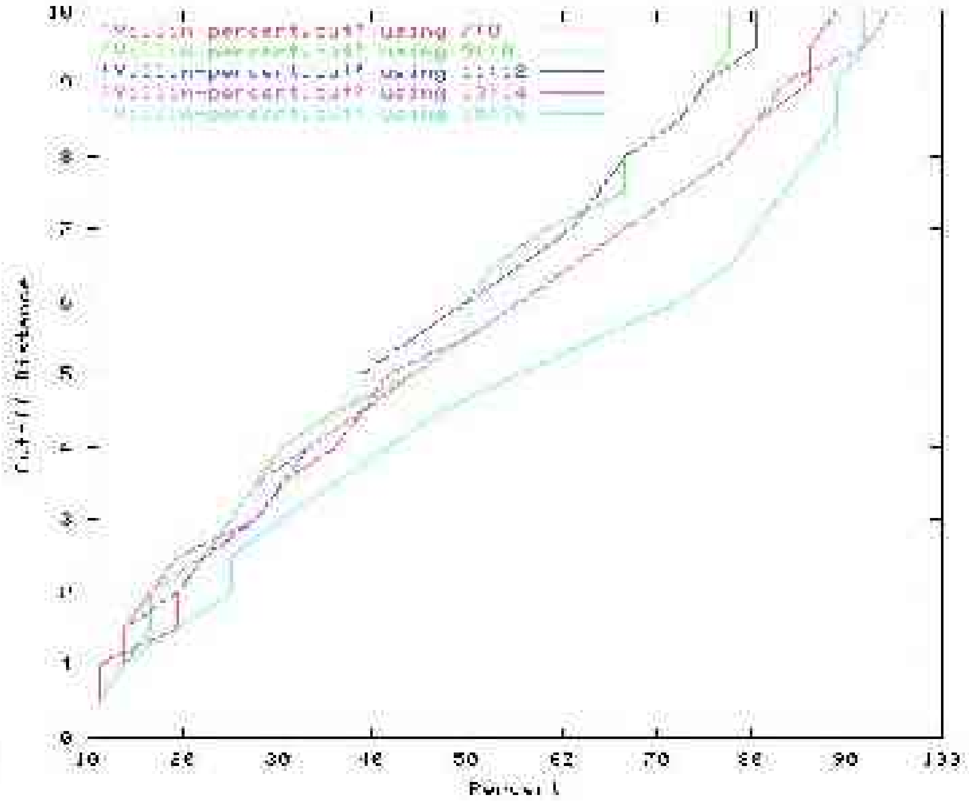
Villin



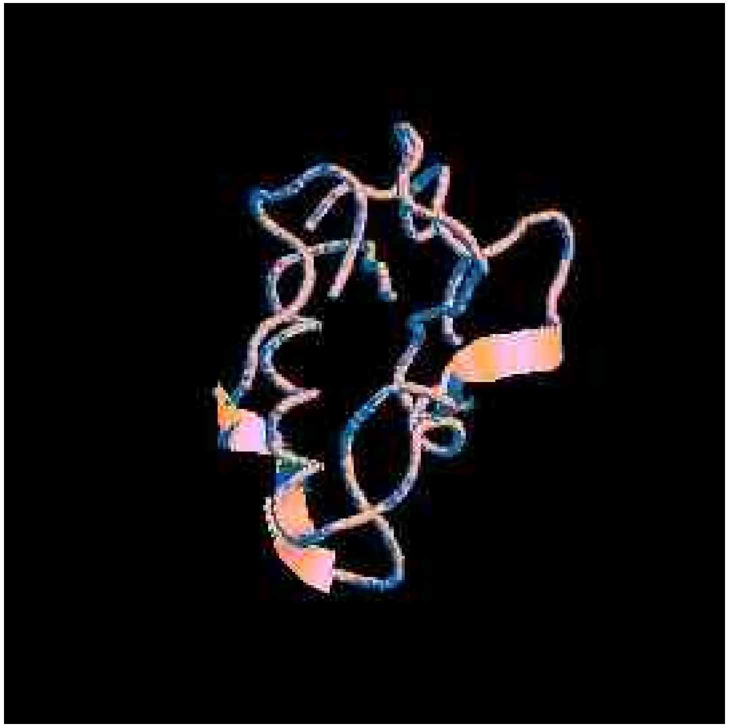


RMSD(Villin)

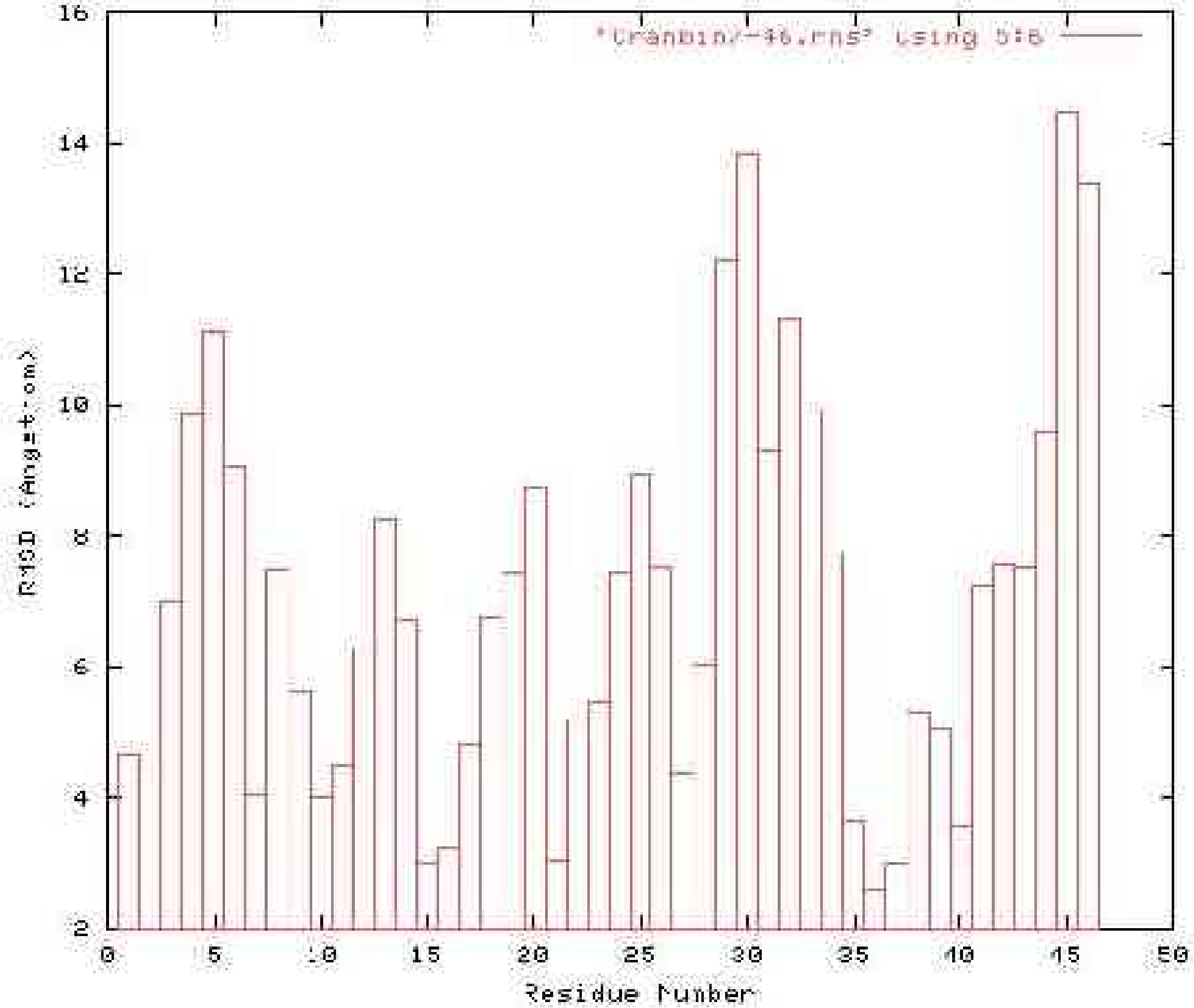

'villin-35-36c.rms' using 5:6
'villin-37-36c.rms' using 5:6
'villin-35-36c.rms' using 5:6
'villin-37-36c.rms' using 5:6
'villin-35-36c.rms' using 5:6
RMSD (Angstroms)
Residue Number
0
5
10
15
20
25
40
45
50
55
60
65
70
75
80


'villin-percent.cut' using 7:8
'villin-percent.cut' using 5:6
'villin-percent.cut' using 1:2
'villin-percent.cut' using 3:4
'villin-percent.cut' using 9:10
Cut-off Distance
Percent
0
1
2
3
4
5
6
7
8
9
10
10
20
30
40
50
60
70
80
90
100

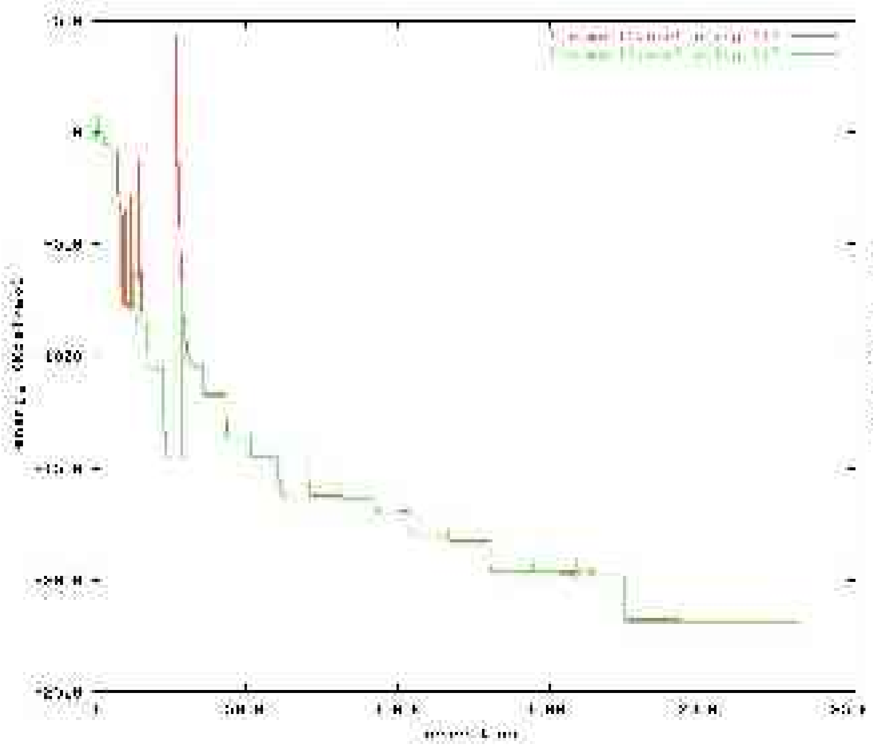
Crambin

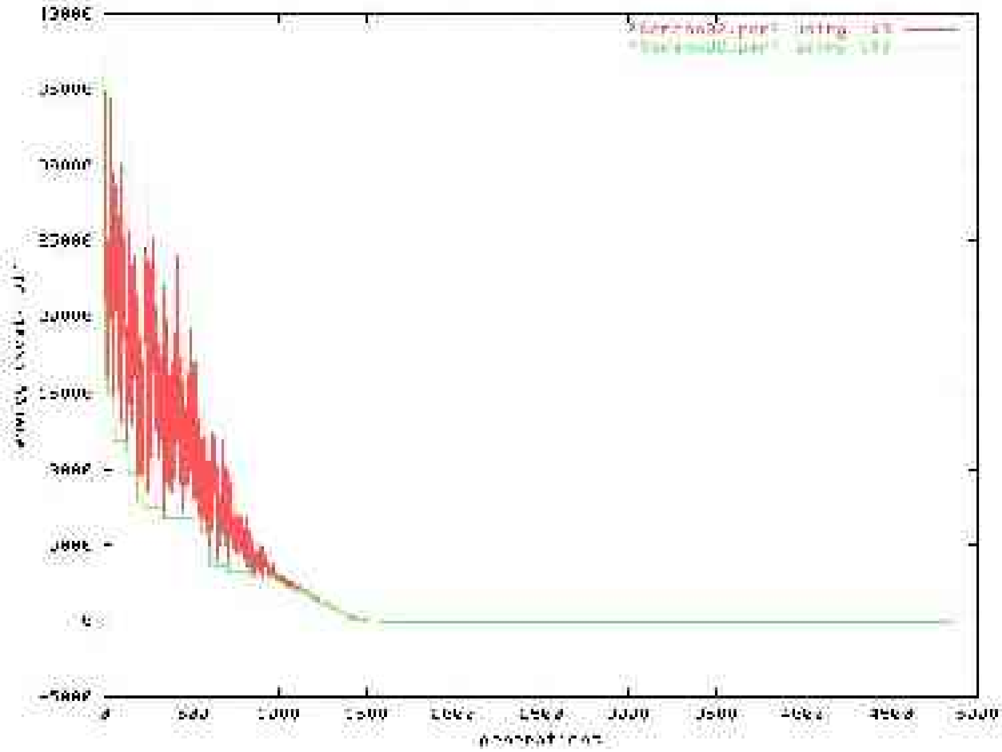


RMSD


'Crambin/~46.rms' using 5:6
RMSD (Angstrom)
16
14
12
10
8
6
4
2
0
5
10
15
20
25
30
35
40
45
50
Residue Number

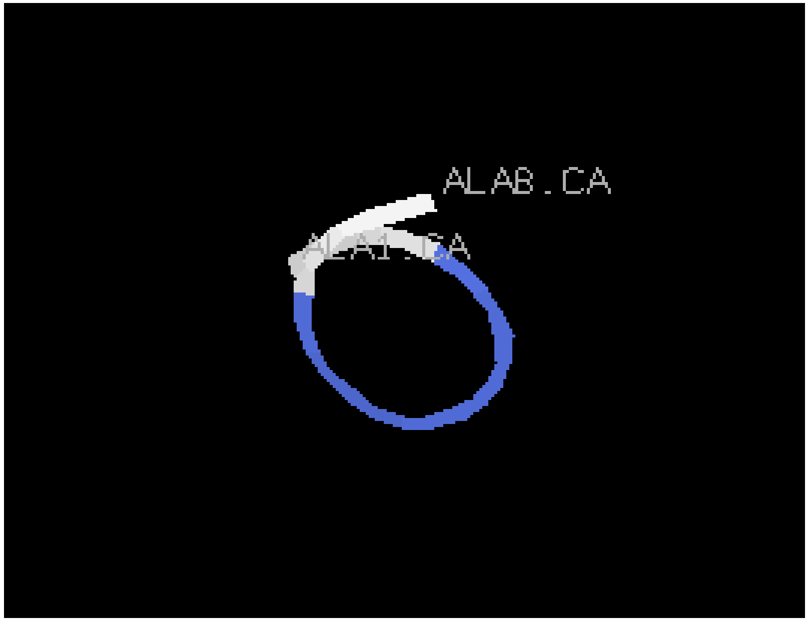
Performance



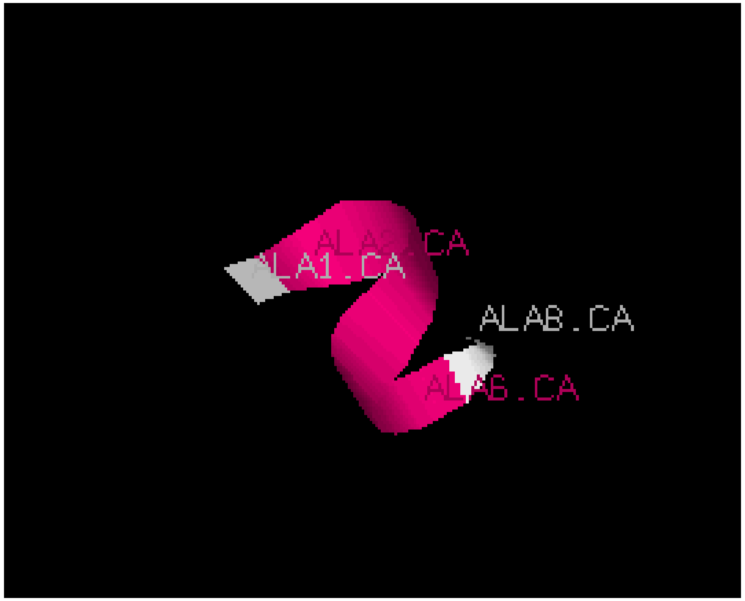


RMSD using the current method = 5.87 to 6.52 Angstrom
RMSD by Simple GA (S.Kremer) = 9 to 10 Angstrom
V. Sundararajan & R. Eils (HPC Asia 2002, IICAI 2003)

Octa-alanine

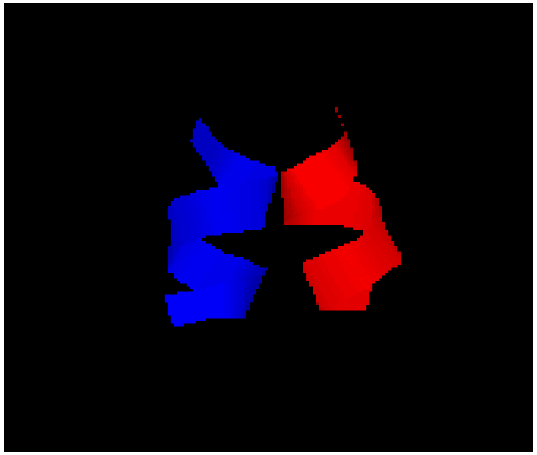


Minimum energy Structure (3)



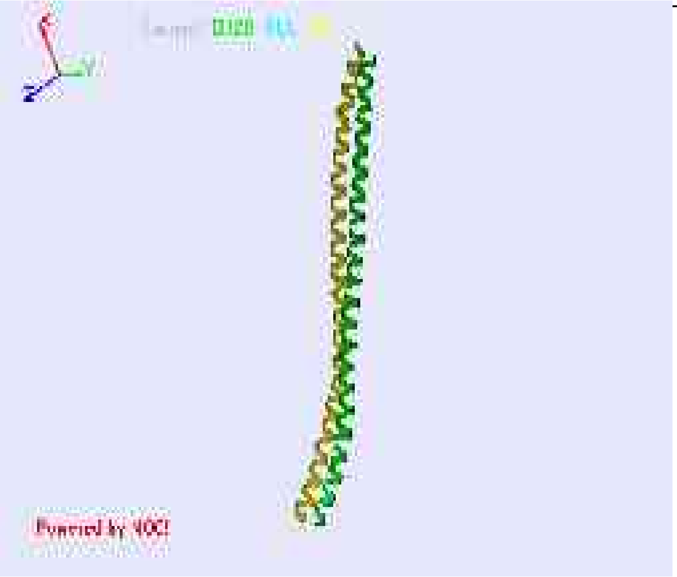
Average Structure (3)

Octaalanine polymerization



Octaalanine chains with one Glycine as linker molecule using both MCS and SA.

Comparison of vimentin with our model with experimental structure



Theoretical model for 5000 MC steps.

Experimental Structure

Rmsd value 2.17 using LGA

Not covered

1. 2D structure optimization using GA
2. Monte Carlo Methods (John Moult)
3. Lattice Models
4. Multi-copy Simulated annealing

Future Work

- Include rotamer library so that the errors due to side chains can be minimised
- Using secondary structures as building blocks may lead to structures more close to native ones
- Developing new models on parallel machines using micro GA as well as parallel GA with local search methods combined with multi-objective optimization

Conclusions

- GA converged to energies which are lower than that for the experimentally determined structure
- Major problem lies with the fitness since there are no clues to know about the native structures from the single force field function
- These methods are also highly applicable to other application areas:
  - Multiple sequence alignment
  - Molecular evolution

Next Hypothesis ?

Hypothesis 2: Protein folding takes place first in bits and pieces (in segments) and these pieces combine to arrive at the native folded structure.

What Strategies ?

Strategy 2: Island Genetic Algorithms is used to evolve the segments first and these evolved segments are combined to evolve the full protein structure.

Computational Demand

Lot of scope for parallelism and the use of Grid would be appropriate

References

- **Proteins**

1. Molecular Biology of the Cell, Alberts et al
2. Introduction to Protein Structure, Branden & Tooze
3. Introduction to Protein Architecture, A.M. Lesk
4. Protein Structure Prediction, (Ed) M.J.E. Sternberg
5. Protein Structure Prediction, (Ed) D.M. Webster (Methods in Molecular Biology, vol 143)
6. W.A. Thomasson, Unravelling the mystery of protein folding (an article for general audience)

References

- **Genetic Algorithms**

1. **J. Holland**, *Adaptation in Natural and Artificial Systems*, Univ of Michigan Press, Ann Arbor, Mich. 1975 Revised edition by MIT Press 1989
2. **Richard Dawkins**, *The Blind Watchmaker*, W.W. Norton and Company, NY 1986; *The Selfish Gene*, Oxford Univ Press, NY 1989.
3. **D.E. Goldberg**, Genetic Algorithms in Search Optimisation and Machine Learning, Addison Wiely 1989.
4. **Z. Michalewicz**, Genetic Algorithms + Data Structures = Evolutionary Programs , Springer-Verlag (1996)
5. **Melanie Mitchell**, An Introduction to Genetic Algorithms, Prentice Hall of India 1998.
6. Parallel Genetic Algorithms, (Ed) Stender

References

- **Application to Protein Structure Prediction**

1. Schulze Kremer in *Protein Structure Prediction* (Ed) D.M. Webster (2000) p175

2. V. Sundararajan and A.S. Kolaskar *Distributed Genetic Algorithms on PARAM for conformational search*, in *Computer Modeling and Simulation of complex biological systems*, Editor S.S. Iyengar, CRC Press, 1998.

3. V. Sundararajan, Predicting protein structure using genetic algorithms: A Review IICAI-03 Dec 2003.