Riconoscimento e recupero dell'informazione per bioinformatica

Dettagli esame

Manuele Bicego

Corso di Laurea in Bioinformatica
Dipartimento di Informatica - Università di Verona

Esame

2 parti:

1. Esame scritto:
   - ⇨ due domande
   - ⇨ 15 punti
2. Seminario o progetto: 15 punti

2

Progetto

- ⇨ Solo per studenti *particolarmente motivati* e autonomi
  - ⇨ suggerimento: fare congiuntamente progetto e tesi (gruppi di due o tre persone)
- ⇨ Requisiti: buona capacità di programmare in MATLAB, voglia di confrontarsi con partner bio-medici
- ⇨ Di cosa si tratta:
  - ⇨ Prima parte: capire un problema, leggere la letteratura, identificare una soluzione, scrivere codice e fare degli esperimenti
  - ⇨ Seconda parte: scrivere una relazioncina -- istruzioni al sito http://profs.sci.univr.it/~swan/Teaching/proposte-tesi.html#SCRITTURARELAZIONE
- ⇨ Argomenti da definire con il docente

3

Seminario

Seminario -- due possibilità:

1. Seminario effettuato l'ultima settimana del corso (in aula)
   - ⇨ seminario fatto da due persone
   - ⇨ seminario di 40-45 minuti
   - ⇨ argomento dato
2. Seminario effettuato quando si vuole nel corso dell'anno (concordando il periodo col docente)
   - ⇨ seminario fatto da una persona singola
   - ⇨ seminario di 40-45 minuti
   - ⇨ identificare un argomento

4

Seminario

- Per preparare il seminario, una volta identificato l'argomento:
  - identificare la relativa letteratura (check con il docente)
  - preparare il seminario (check con il docente)
  - seminario (voto)

5

Letteratura:

- Strumenti per la ricerca
  - google scholar (http://scholar.google.it/) - google in generale
  - ACM digital library (http://portal.acm.org/dl.cfm) – più info
  - IEEE digital library (http://ieeexplore.ieee.org/Xplore/dynhome.jsp) – più info
  - Pubmed (http://www.ncbi.nlm.nih.gov/pubmed/) – più bio
  - Altre: DBLP, Citeseer,...
- Gli articoli hanno diversi valori a seconda di:
  - articoli su rivista
    - classificazione riviste con Impact Factor: Journal Citation Reports (via ISI-Web of knowledge - *www.isiknowledge.com/* -> additional resources)
  - articoli su conferenza
    - conferenze internazionali e revisionate (IEEE, ACM)
    - un possibile ranking (per informatica) si può trovare sul sito del GRIN http://www.di.unipi.it/grin/Classif.Conferenze2000.html

6

Giornali rilevanti:

⇨ Bioinformatics

⇨ Journal of Computational Biology,

⇨ Briefings in Bioinformatics

⇨ BMC Bioinformatics

⇨ Journal of Bioinformatics and Computational Biology

⇨ IEEE/ACM Transactions on Computational Biology and Bioinformatics

⇨ ...

7

Impostazione del seminario

⇨ Seminario su una metodologia:

- ⇨ contesto e obiettivo, motivazioni che portano allo sviluppo della metodologia
- ⇨ descrizione della metodologia
- ⇨ descrizione di una o più applicazioni della metodologia in ambito di bioinformatica
- ⇨ limiti e possibili estensioni (cenni)

⇨ Seminario su una specifica applicazione:

- ⇨ contesto biologico, background per capire
- ⇨ descrizione del problema che si vuole risolvere
- ⇨ descrizione della metodologie tipicamente utilizzate per risolvere il problema
- ⇨ limiti e possibili estensioni (cenni)

8

Esempi di seminari

- ⇨ Filogenesi (2 persone 40 minuti):
  - ⇨ Mattioli-Daipré
- ⇨ Algoritmi genetici in generale (1 persona, 20 minuti)
  - ⇨ Heller
- ⇨ Motif discovery con algoritmi genetici (1 persona, 20 minuti)
  - ⇨ Heller

10

Evoluzione Molecolare: la Filogenesi

Seminario del corso
RECUPERO DELL'INFORMAZIONE
(docente Dott. Bicego Manuele)

Alberto Dai Prè (vr067036)
Veronica Mattioli (vr069402)

Università degli Studi di Verona
Corso di Laurea in Bioinformatica
A.A. 2009/2010

Indice

- **Meccanismi molecolari alla base dei processi evolutivi.**
- **Geni ortologhi e paraloghi.**
- **Alberi filogenetici.**
- **I “passi” della filogenesi.**
- **Metodi basati sulle sequenze: metodi tree-searching.**
- **Determinazione delle distanze genetiche tra sequenze.**
- **Metodi distance-based:**
  - **UPGM**
  - **Neighbor-Joining**
    - **L'algoritmo del Neighbor-Joining**
    - **Esempio di applicazione dell'algoritmo Neighbor-Joining**
    - **Varianti del Neighbor-Joining.**

Meccanismi molecolari alla base dei processi evolutivi

- La trasmissione dell'informazione genetica avviene per mezzo della replicazione del DNA.
- L'apparato di replicazione è molto accurato, ma (con probabilità piccola) si possono verificare degli ”errori” (mutazioni).
- Ecco perché gli organismi viventi, pur discendendo da un unico progenitore comune, possono avere genomi di dimensioni molto diverse.

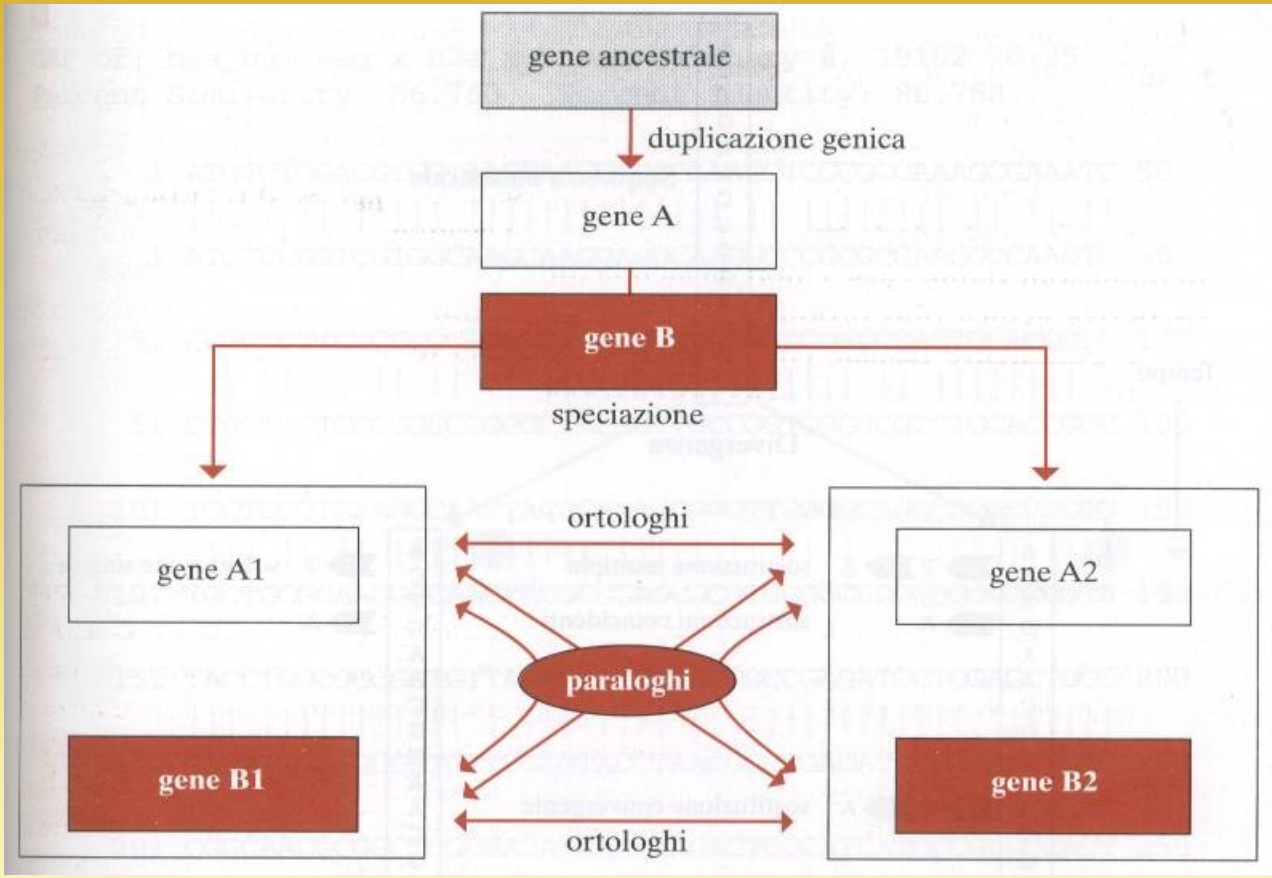
Meccanismi molecolari alla base dei processi evolutivi (2)

- Lo studio della velocità e dei tipi di cambiamenti che avvengono nel materiale genetico (o nei suoi prodotti) è l'oggetto d'indagine degli studi di evoluzione molecolare.
- Tipi di mutazioni: *mutazioni puntiformi*, *delezioni*, *inserzioni* ed *inversioni*.
- Nell'ambito delle mutazioni puntiformi, possiamo avere la *transizione* e la *trasversione*.

Tipi di mutazioni che intervengono nel corso dell'evoluzione molecolare

A) A partire da una sequenza progenitore possono verificarsi:
B) Sostituzioni puntiformi;
C) delezioni;
D) inserzioni;
E) inversioni.
F) Questo fa sì che la sequenza discendente possa risultare differente da quella ancestrale anche nella lunghezza.

Relazione tra geni ortologhi e paraloghi



Relazioni tra geni ortologhi e paraloghi illustrate dall'evoluzione a partire da un gene ancestrale, con una duplicazione genica nella specie 0, che genera i geni A e B, seguita da una speciazione che origina i geni A1 e B1 nella specie 1 e A2 e B2 nella specie 2.

Geni ortologhi e paraloghi

- Omologia: carattere qualitativo che posseggono quelle sequenze che derivano da un progenitore comune in seguito al processo evolutivo.
- Due geni omologhi la cui divergenza evolutiva risale ad un evento di speciazione si definiscono ortologhi.
- Ma due gene possono iniziare a divergere in modo indipendente anche in seguito ad un processo di duplicazione genica. In questo caso, i geni sono definiti paraloghi.

Geni ortologhi e paraloghi (2)

- La similarità che si osserva tra due o più sequenze è il criterio principale per ipotizzare una relazione di omologia.
- Due sequenze che non mostrano un significativo livello di similarità possono essere ugualmente omologhe, ma talmente divergenti da non mostrare un apprezzabile grado di similarità.

Alberi filogenetici

- Le relazioni evolutive tra geni omologhi possono essere rappresentate attraverso *alberi filogenetici*.
- Albero filogenetico: grafico costituito da nodi e da rami, in cui ogni ramo mette in relazione solo due nodi.
- I nodi terminali rappresentano le unità tassonomiche attuali, mentre i nodi interni rappresentano le unità tassonomiche ancestrali.

Alberi filogenetici (2)

- Le unità tassonomiche attuali corrispondenti alle sequenze omologhe oggetto dell'analisi sono definite unità tassonomiche operative (OTUs).
- Se un albero filogenetico descrive esclusivamente le relazioni filogenetiche tra i vari nodi, la lunghezza dei diversi rami non ha alcun significato (l'albero è definito cladogramma).
- Se la lunghezza dei rami è riportata proporzionale alla distanza evolutiva tra i nodi, l'albero è definito filogramma.

Alberi filogenetici (3)

- **Obiettivi dell'analisi filogenetica**: determinare la topologia dell'albero che descrive le relazioni filogenetiche tra le specie in esame e determinare la lunghezza dei vari rami.

Alberi filogenetici (4)

- Non sempre l'albero costruito analizzando i geni corrisponde all'albero che descrive le relazione tra le specie. L'incongruenza tra “albero dei geni” e “albero delle specie” può essere determinata da diverse cause.
- La causa più frequente è che alcuni dei geni considerati non sono realmente ortologhi ma paraloghi, hanno cioè avuto origine da un processo di duplicazione genica e non di speciazione.

I “passi” della filogenesi

- Ricerca per similarità: si devono cercare sequenze ortologhe alla sequenza in esame (*sequenza query*).
- Allineamento multiplo.
- Applicazione di un metodo di ricostruzione filogenetica: metodi tree-searching o metodi distance-based.
- Visualizzazione dell’albero filogenetico.
- Validazione della ricostruzione filogenetica: l'attendibilità di un'ipotesi filogenetica si può valutare misurando la significatività statistica dei vari nodi che compongono l'albero filogenetico in questione.

Metodi basati sulle sequenze: i metodi tree-searching

- I metodi tree-searching ricercano l'albero migliore all'interno dello spazio degli alberi.
- Comprendono: *Maximum Parsimony* e *Maximum Likelihood.*
- Maximum Parsimony: identifica l'albero filogenetico che richiede il minor numero possibile di sostituzioni che spieghino le differenze osservate tra le sequenze in esame.
- La lunghezza di ciascun ramo dell'albero è pari al numero minimo di sostituzioni occorse tra i nodi che esso congiunge.

Metodi basati sulle sequenze: i metodi tree-searching (2)

- Limitazione 1: la lunghezza di ciascun ramo non dell'albero non stima in modo accurato l'effettiva distanza genetica (non tiene conto di sostituzioni multiple o convergenti).
- Limitazione 2: non tiene conto della composizione nucleotidica della sequenza in esame.
- Maximum Likelihood: restituisce un albero che giustifica meglio il set di dati in esame, ovvero il multi-allineamento fornito in input.
- Limitazione: grande quantità di calcolo.

Determinazione delle distanze genetiche tra sequenze

- La distanza genetica tra due sequenze omologhe è determinata dal numero di sostituzioni che hanno avuto luogo nel corso dell'evoluzione nelle sequenze stesse.
- L'unità di misura è data dal numero di sostituzioni per sito.
- A causa della possibilità di sostituzioni multiple sullo stesso sito, di sostituzioni convergenti o di retro-mutazioni, il numero di sostituzioni che è osservato tra una coppia di sequenze è inferiore rispetto al numero di sostituzioni che effettivamente hanno avuto luogo.

Determinazione delle distanze genetiche tra sequenze (2)

- Nello studio dell'evoluzione si possono considerare sia sequenze di acidi nucleici sia di proteine. Le prime sono più informative.
- Assunzioni “a priori” utilizzate per distinguere i vari modelli proposti per la stima delle distanze genetiche:
  - tutti i siti evolvono in modo indipendente;
  - tutti i siti possono mutare con la stessa probabilità;
  - tutti i tipi di sostituzione sono ugualmente probabili;
  - la velocità di sostituzione e costante nel tempo;
  - la composizioni in basi delle sequenze e all'equilibrio.

Determinazione delle distanze genetiche tra sequenze (3)

- 1969 – Jukes e Cantor (JC69) che richiede la stima di un solo parametro pari alla probabilità di una sostituzione.
- 1980 – Kimura (KIM): consente una diversa velocità per transizioni e trasversioni e, quindi, adotta due diversi parametri.
- 1981 – Felsenstein (F81): estende il modello JC69 tenendo conto della reale composizione nucleotidica delle sequenze analizzate.
- 1984 – Lenave et al.: oltre a tener conto della reale composizioni in basi delle sequenze in esame consente una diversa probabilit per i sei tipi di sostituzioni.
- 1985 – Hasegawa et al.: estende il modello F81 assumendo una diversa probabilità per transizioni e trasversioni (come KIM).
- 1992 – Tamura: oltre alla diversa probabilità di transizioni e trasversioni, tiene conto della composizioni in G+C.

Determinazione delle distanza genetiche tra sequenze (4)

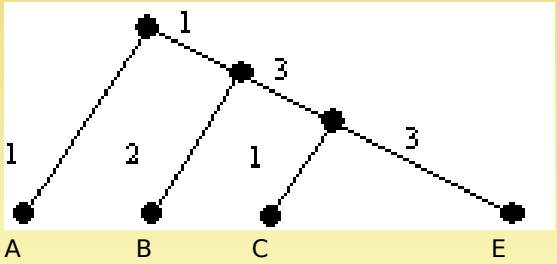
Modello	Matrice delle probabilità delle sostituzioni nucleofile	Composizione in basi nello stato stazionario (f_i^{∞}, i = A, C, G, T)	Numero parametri
Jukes & Cantor (1969)	$\begin{matrix} p_{11} & \alpha & \alpha & \alpha \\ \alpha & p_{22} & \alpha & \alpha \\ \alpha & \alpha & p_{33} & \alpha \\ \alpha & \alpha & \alpha & p_{44} \end{matrix}$	$\left[\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right]$	1
Kimura (1980)	$\begin{matrix} p_{11} & \beta & \alpha & \beta \\ \beta & p_{22} & \beta & \alpha \\ \alpha & \beta & p_{33} & \beta \\ \beta & \alpha & \beta & p_{44} \end{matrix}$	$\left[\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right]$	2
Tamura (1992)	$\begin{matrix} p_{11} & \theta\beta & \theta\alpha & (1-\theta)\beta \\ (1-\theta)\beta & p_{22} & \beta & (1-\theta)\alpha \\ (1-\theta)\alpha & \beta & p_{33} & (1-\theta)\beta \\ (1-\theta)\beta & \alpha & \beta & p_{44} \end{matrix}$	$\left[\frac{1-\theta}{2}, \frac{\theta}{2}, \frac{\theta}{2}, \frac{1-\theta}{2}\right]$	3
Tajima and Nei (1982)	$\begin{matrix} p_{11} & \pi_C\alpha & \pi_G\alpha & \pi_T\alpha \\ \pi_A\alpha & p_{22} & \pi_G\alpha & \pi_T\alpha \\ \pi_A\alpha & \pi_C\alpha & p_{33} & \pi_T\alpha \\ \pi_A\alpha & \pi_C\alpha & \pi_G\alpha & p_{44} \end{matrix}$	$[\pi_A, \pi_C, \pi_G, \pi_T]$	4
Hasegawa et al. (1985)	$\begin{matrix} p_{11} & \pi_C\beta & \pi_G\alpha & \pi_T\beta \\ \pi_A\beta & p_{22} & \pi_G\beta & \pi_T\alpha \\ \pi_A\alpha & \pi_C\beta & p_{33} & \pi_T\beta \\ \pi_A\beta & \pi_C\alpha & \pi_G\beta & p_{44} \end{matrix}$	$[\pi_A, \pi_C, \pi_G, \pi_T]$	5
Lanave et al. (1984) Saccone et al. (1990)	$\begin{matrix} p_{11} & \pi_C\beta_1 & \pi_G\alpha_1 & \pi_T\beta_2 \\ \pi_A\beta_1 & p_{22} & \pi_G\beta_3 & \pi_T\alpha_2 \\ \pi_A\alpha_1 & \pi_C\beta_3 & p_{33} & \pi_T\beta_4 \\ \pi_A\beta_2 & \pi_C\alpha_2 & \pi_G\beta_4 & p_{44} \end{matrix}$	$[\pi_A, \pi_C, \pi_G, \pi_T]$	9

Metodi basati sulle distanze

- I metodi distance-based assumono che ad ogni arco dell'albero evolutivo sia associata una misura (*peso*), della distanza evolutiva tra le due specie che individuano gli estremi dell'arco in questione.
- *Distanza tra due foglie* `X` e `Y` nell'albero evolutivo (`d(X, Y)`): somma dei pesi degli archi che compongono il percorso che congiunge `X` e `Y`.

Metodi basati sulle distanze (2)

- Dato un insieme di specie $S = \{s_1, s_2, \ldots s_n\}$, ed una filogenesi T (con radice o senza radice), associamo a T una matrice $M_{(n \times n)}$, detta matrice delle distanze, che rappresenti la distanza nella filogenesi tra tutte le coppie di specie in S.
- Esempio:



d(A, E) = 1+1+3+3 = 8

$$M = \begin{array}{c|cccc} & A & B & C & E \\ A & 0 & 4 & 6 & 8 \\ B & 4 & 0 & 6 & 8 \\ C & 6 & 6 & 0 & 4 \\ E & 8 & 8 & 4 & 0 \end{array}$$

Metodi distance-based: UPGMA

- Il metodo UPGMA (*Unweighted Pair Group Method with Arithmetic mean*) presuppone la validità che la velocità di evoluzione delle sequenze sia costante lungo tutti i rami dell'albero.
- Utilizza un algoritmo di clusterizzazione iterativo che procede associando via via le sequenze o cluster di sequenze più simili tra loro.
- Guardando la matrice delle distanze genetiche, individua la coppia di OTUs caratterizzata dalla distanza minima.

Metodi Distance-Based: UPGMA (2)

- Tale coppia di OTUs forma un cluster ed è trattata come un'unica OTUs nella successiva iterazione dove è prima ricalcolata la matrice delle distanze genetiche, e quindi selezionata la nuova coppia di OTUs con il valore minimo della distanza genica, che formerà un nuovo cluster.
- Questa procedura è ripetuta finché non rimangono due sole OTUs che sono infine congiunte da una ramo il cui punto medio costituisce la radice dell'albero filogenetico.

Metodi distance-based: Neighbor-Joining

- Il metodo del Neighbor-Joining risolve il problema della costruzione di alberi filogenetici con matrice delle distanze. Dato un insieme di specie `S` ed una matrice delle distanze `M` simmetrica, la questione è quella di trovare la filogenesi `T` che rappresenta l'evoluzione delle specie in `S` e che sia consistente con la matrice `M`.
- L'algoritmo si basa sull'iterazione dei seguenti passi:

  (1) tra tutte le coppie di specie della matrice `M` scegliere quella avente distanza minore (es. `(X, Y)`);

Metodi Distance-Based: Neighbor-Joining (2)

(2) aggiungere all'albero `T`, inizialmente vuoto, un nodo interno `{X, Y}` e i due nodi `X` e `Y` che collego al nodo `{X, Y}`;

(3) assegnare agli archi che collegano le due specie al nodo `{X, Y}` pesi pari a `d(X, Y)/2`;

(4) costruisco la matrice `M'` a partire da `M`, sostituendo agli elementi `X` e `Y` l'elemento `{X, Y}` e definendo le distanze tra `{X, Y}` e gli altri elementi della matrice nel seguente modo:

$$\forall\ z \in S - \{X, Y\},\quad d'(\{X, Y\}, z) = d(X, z) - d(X, \{X, Y\})$$

L'algoritmo del metodo Neighnor-Joining

- Supponiamo che l'albero `T` si possa vedere come una coppia `T = (V, E)` dove `V` è l'insieme dei nodi ed `E` è l'insieme degli archi. Inizialmente gli insiemi `V` ed `E` sono vuoti.
- Poiché consideriamo albero pesati, viene introdotta anche una ipotetica funzione $P: E \rightarrow R$ che consente di associare ad un arco un valore (peso dell'arco).
- L'algoritmo si basa sull'iterazione dei passi definiti in precedenza:

L'algoritmo del metodo Neighbor-Joining (2)

```
1. Neighbor-Joining():
2.   E = {};
3.   V = {};
4.   REPEAT
5.     dxy = Min(d(x, y)) per ogni x, y appartenente a
   M; /* Corrisponde al passo (1) */
6.     V = V + {X, Y} + x + y; /* Aggiunge all'insieme
   dei nodi di T i nodi {X, Y}, x , y */
7.     E = E + ({X, Y}, x) + ({X, Y}, y); /* Aggiunge
   all'insieme degli archi di T gli archi ({X, Y}, x) e
   ({X, Y}, y) */
8.    P({X, Y}, x) = dxy/2; /* Corrisponde al passo (3)
  */
```

L'algoritmo del metodo Neighbor-Joining (3)

```
9.    P({X, Y}, y) = dxy/2; /*Corrisponde al passo
  (3)*/
10.   FOR z appartenente ad S { /*Costruisce M'(n-1, n-
  1) lasciando inalterate le distanze che non
  coinvolgono X e Y}*/
11.     IF (z != x && z != y)
12.       M'({X, Y}, z) = M(X, z) - M(X, {X, Y});
  /*Corrisponde la passo (4)*/
13.   }
14.   M = M'; /*Alla prossima iterazione occorrerà
  considerare la nuova matrice*/
15.  UNTIL M' è una matrice costituita da un solo
  elemento;
```

Esempio di applicazione dell'algoritmo del metodo Neighbor-Joining

Applichiamo l'algoritmo del Neighbor-Joining con $S=\{A,B,C,D\}$ e consideriamo la matrice

$$M = \begin{array}{c|cccc} & A & B & C & D \\ A & 0 & 2 & 6 & 6 \\ B & & 0 & 6 & 6 \\ C & & & 0 & 2 \\ D & & & & 0 \end{array}$$

Prima iterazione:

`Min (d(x, y)) = d(A, B);`

Aggiungere il nodo `{A, B}`;

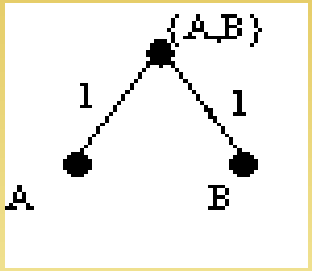
Costruire `M'`: `d({A, B}, C) =`

`d(A, C) – d(A, {A, B}) = 6 – 1 = 5`

e `d({A, B}, D) =`

`d(A, D) – d(A, {A, B}) = 6 – 1 = 5;`

Ripetere il ciclo con `M = M'`, dove



$$M' = \begin{array}{c|ccc} & \{A,B\} & C & D \\ \{A,B\} & 0 & 5 & 5 \\ C & & 0 & 2 \\ D & & & 0 \end{array}$$

Esempio di applicazione dell'algoritmo Neighbor-Joining (2)

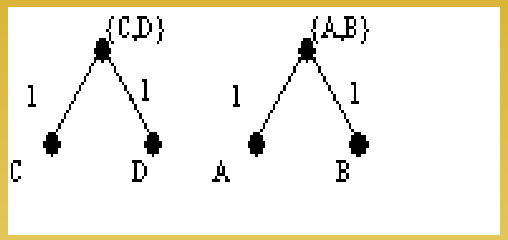
Seconda iterazione:

`Min (d(x, y)) = d(C, D);`

Aggiungere il nodo `{C, D}`;

Costruire `M'`: `d({C, D}, {A, B}) = d(C, {A, B}) – d(C, {C, D}) = 5 – 1 = 4;`

Ripetere il ciclo con `M = M'`, dove



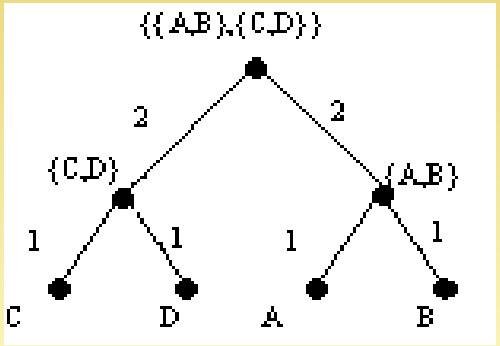
$$M' = \begin{array}{c|cc} & \{A,B\} & \{C,D\} \\ \{A,B\} & 0 & 4 \\ \{C,D\} & & 0 \end{array}$$

Terza iterazione:

`Min (d(x, y)) = d({A, B}, {C, D});`

Aggiungere il nodo `{{A, B}, {C, D}};`

`M'` è vuota ----> l'algoritmo termina



Varianti del metodo Neighbor-Joining

- Problema: il Neighbor-Joining ad ogni passo prende una decisione sola.
- Soluzione: approccio Multi-Neighbor-Joining
  - Considera ad ogni iterazione possibilità multiple.
- Problema: il Neighbor-Joining ha una complessità, in termini di tempo, di $O(n^3)$.
- Soluzione: approccio Fast-Neighbor-Joining.
  - Esegue le sue operazione in tempo $O(n^2)$.

Varianti del metodo Neighbor-Joining (2)

- Problema: il Neighbor-Joining, nel momento in cui si calcola le nuove distanze dei cluster creati, non prende in considerazione le dimensioni dei cluster che sono stati fusi insieme.
- Soluzione: approccio Avarage Linkage.
  - La misura della distanza tra due cluster è considerata come la media della distanza di ogni membro del cluster da ogni membro dell'altro.

Conclusioni

- La filogenesi è un importante strumento per rappresentare le relazioni evolutive tra gli organismi.
- Diversi approcci per la filogenesi: il più utilizzato è il distance-based.
- Tra l'approccio distance-based spicca il Neihgbor- joining perché:
  - abbastanza rapido;
  - tiene conto delle diverse velocità di evoluzione lungo i diversi rami dell'albero;
  - piuttosto affidabile.
- Il Neighbor-Joining ha dei problemi.
  - Ci sono delle varianti.

Bibliografia

- Giorgio Valle, Manuela Helmer Citterich, Marcella Attimoli e Graziano Pesole, *Introduzione Alla Bioinformatica*, Zanichelli (2003)
- Articolo *Fast Neighbor-Joining*, di Isaac Elias e Lens Lagergren
- Articolo *A multi-neighbor-joining approach for phylogenetic tree reconstruction and visualization,* di Ana Estela A. da Silva, Wilfredo J.P. Villanueva, Helder Knidel, Vinícius Bonato, Sérgio F. dos Reis e Fernando J. Von Zuben
- Alcune dispense e siti internet.

Genetic Algorithms

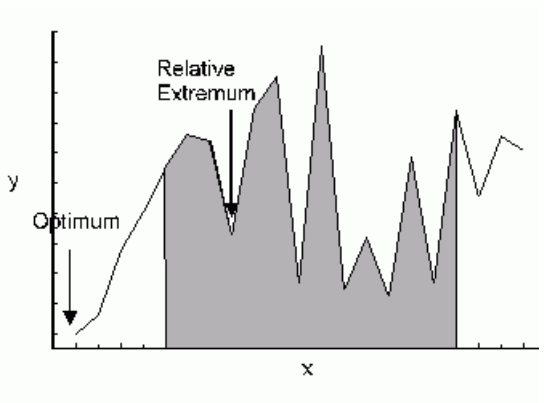
Davide Martin Heller, VR070256

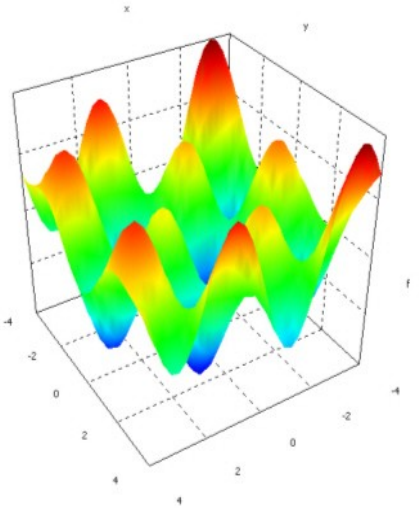
Outline

- Definition & History
- Main concepts
- Simple Example
- Applications in Bioinformatics
- Final words

What is it?

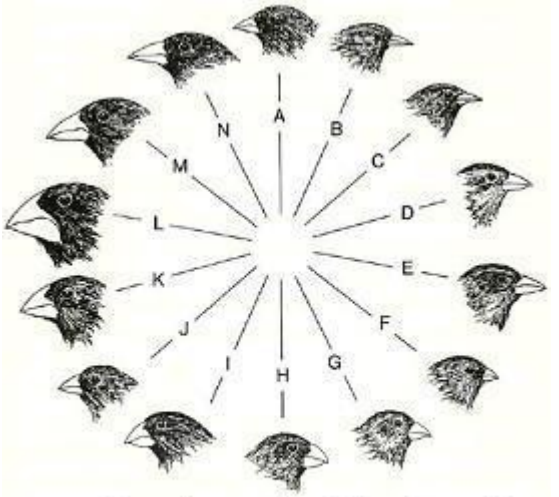
- A search algorithm based on evolution theory
- First mathematical formulation by John Henry Holland in ground-breaking *"Adaptation in natural and Artificial systems"* (1975, University of Michigan Press. Ann Arbor, MA.)
- Used in optimization problems and global search



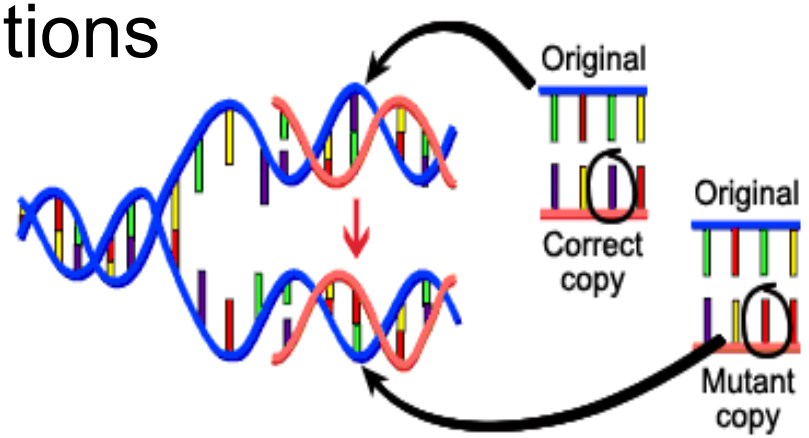


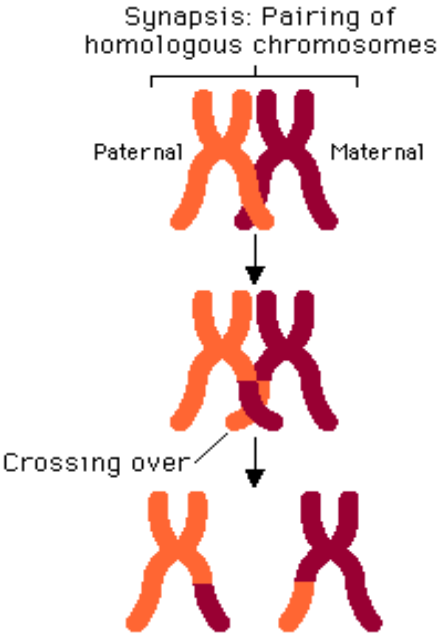
Foundations/Basics

- Evolution Theories
  - Natural selection
    - "Survival of the fittest"
  - Inheritance
- Concepts of modern genetics
  - Crossing over
  - Mutations

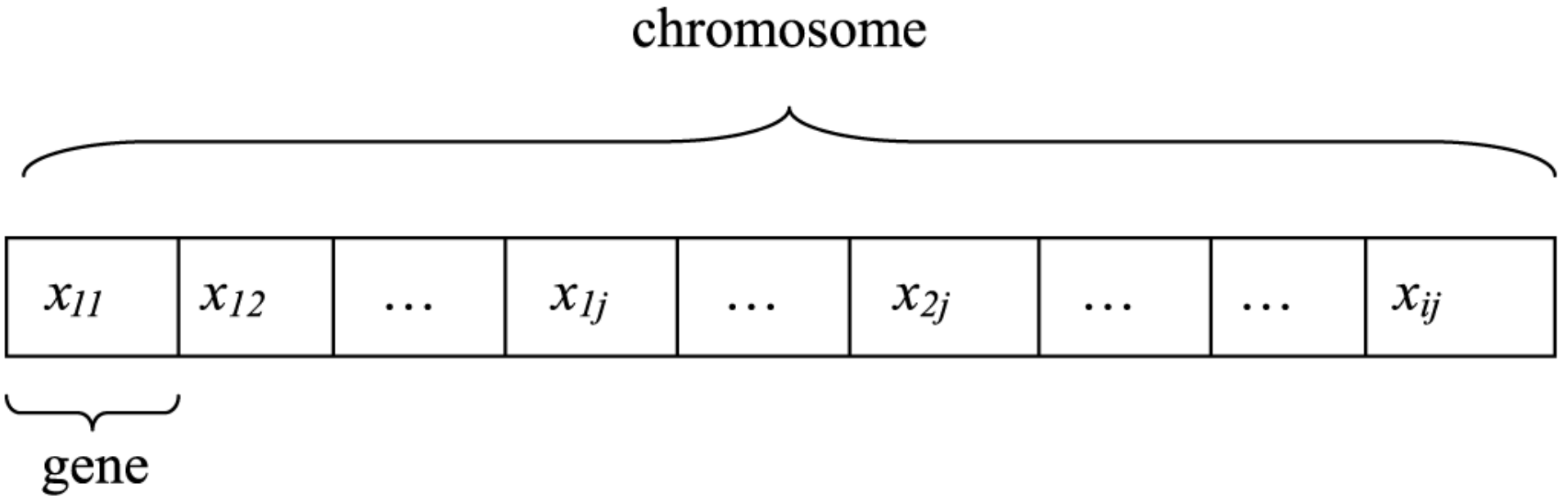


From Fig. 8-1 in Icons of Evolution by Jonathan Wells, page 161. Figure by Jody Sjogren.





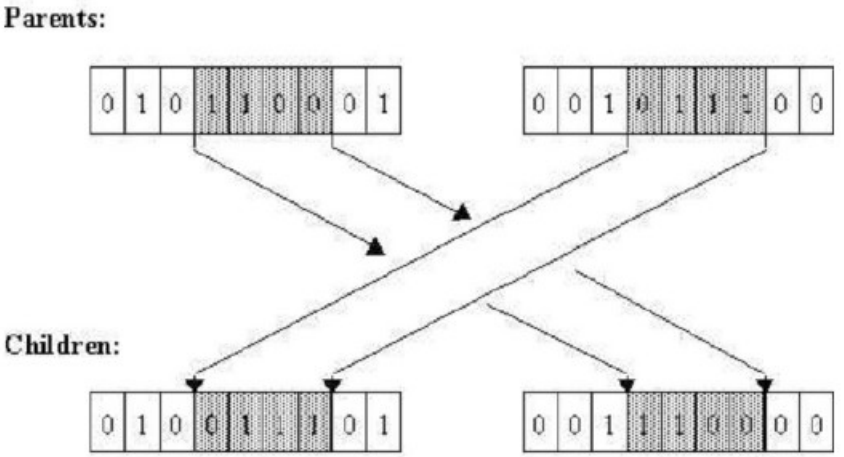
Individual / Chromosome



Is a solution to our problem

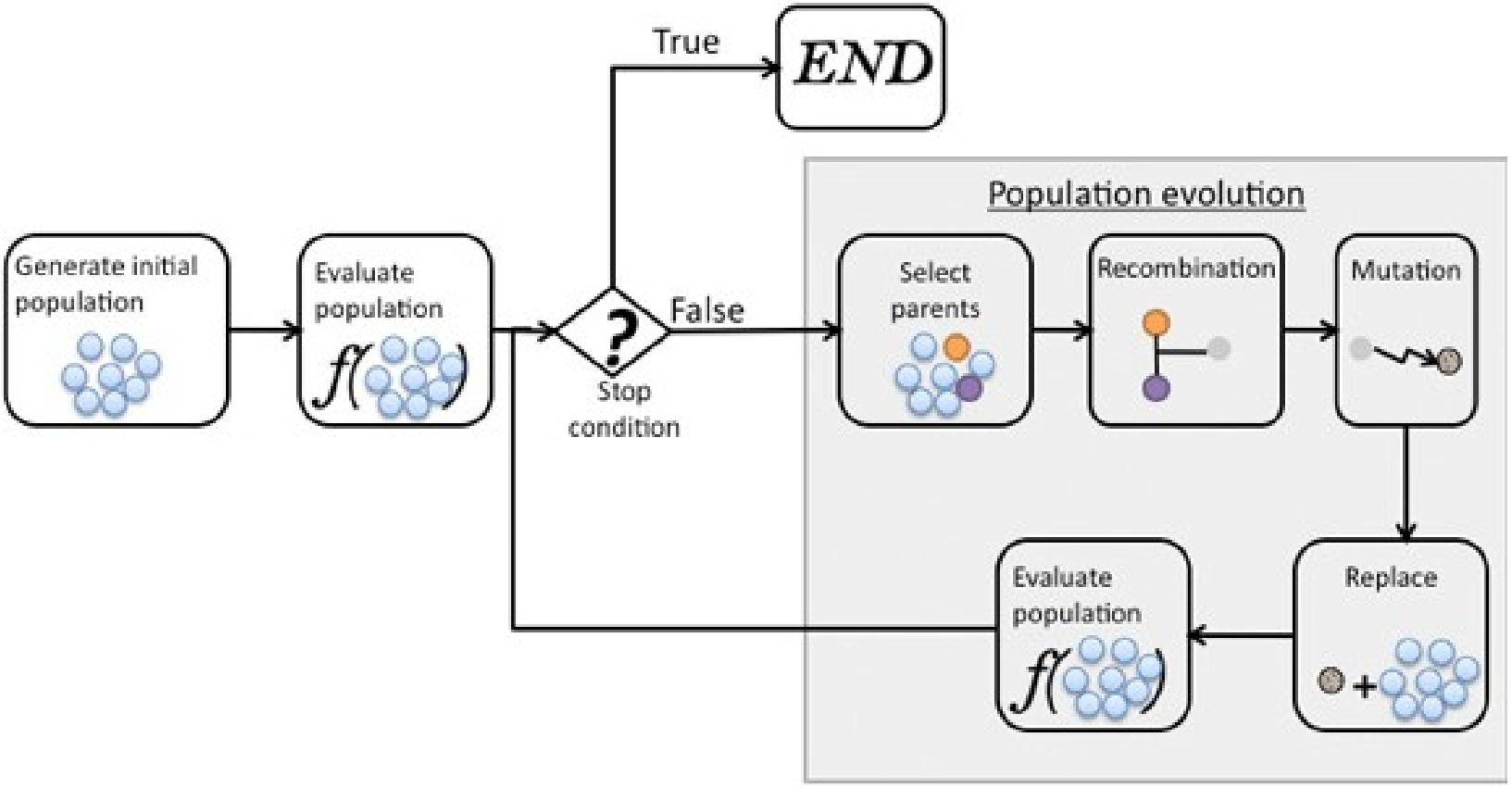
Metaphore

- Nature
  - Population
  - Selection
  - Genetic operations
- Computer Science
  - Possible solutions
  - Fitness Function
  - Binary operations



0 ⇒ 1

Steps


True
END
Population evolution
Generate initial population
Evaluate population
f(
)
?
Stop condition
False
Select parents
Recombination
Mutation
Evaluate population
f(
)
Replace
+

Simple example

- Maximization of a Binary String
  - Given a string of length N
  - Maximise number of 1
  - Solution is logic but let's go with GA

Example:

N =10

1	1	1	1	1	1	1	1	1	1

Initial Population

- Initial set of indivuals/chromosomes
- Often randomly choosen

- Example:
  - S1 = 1 1 1 1 0 1 0 1 0 1
  - S2 = 0 1 1 1 0 0 0 0 0 0
  - S3 = 1 0 1 0 0 1 1 1 0 1

Fitness function

- Principle of “survival of the fittest”
- Search Direction for the Algorithm
- Optimal solution maximises Fitness function

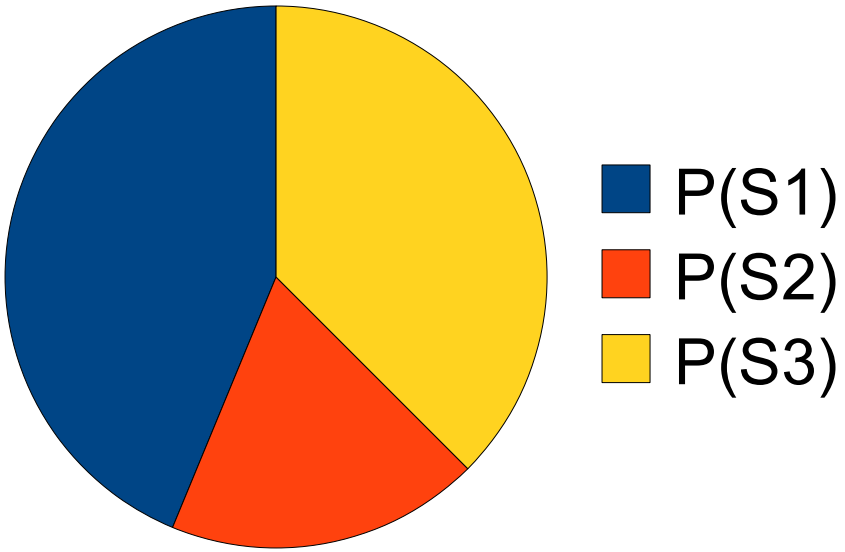
$$f(S) = \sum_{1}^{N} S[i]$$

- Example:
  - S1 = 1 1 1 1 0 1 0 1 0 1 → f (S1) = 7
  - S2 = 0 1 1 1 0 0 0 0 0 0 → f (S2) = 3
  - S3 = 1 0 1 0 0 1 1 1 0 1 → f (S3) = 6

Selection

- Based on fitness function(f) given value
  - ex. Roulette wheel

$$P(S) = \frac{f(S)}{\sum_{i=1}^{3} f(S_i)}$$

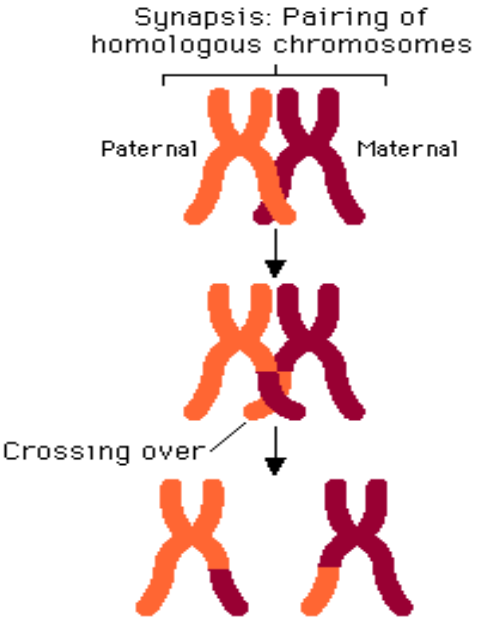


Parents chosen:

- S'1 = S3
- S'2 = S1

Crossing over

- Recombination
- ex.1 Point crossing over



- Example:

S'1 = 10100 11101 → S'1 = 10100 10101

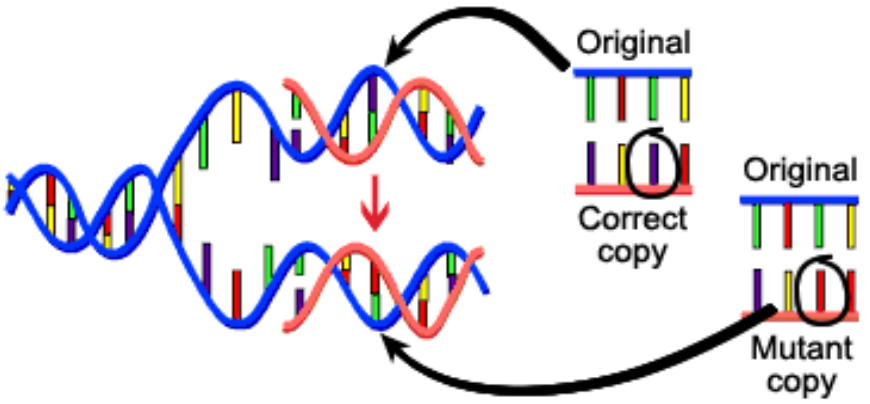
S'2 = 11110 10101 → S'2 = 11110 11101

Mutation

- Event that changes genetic structure
- ex. SNP(Single Nucleotide Polimorphism)

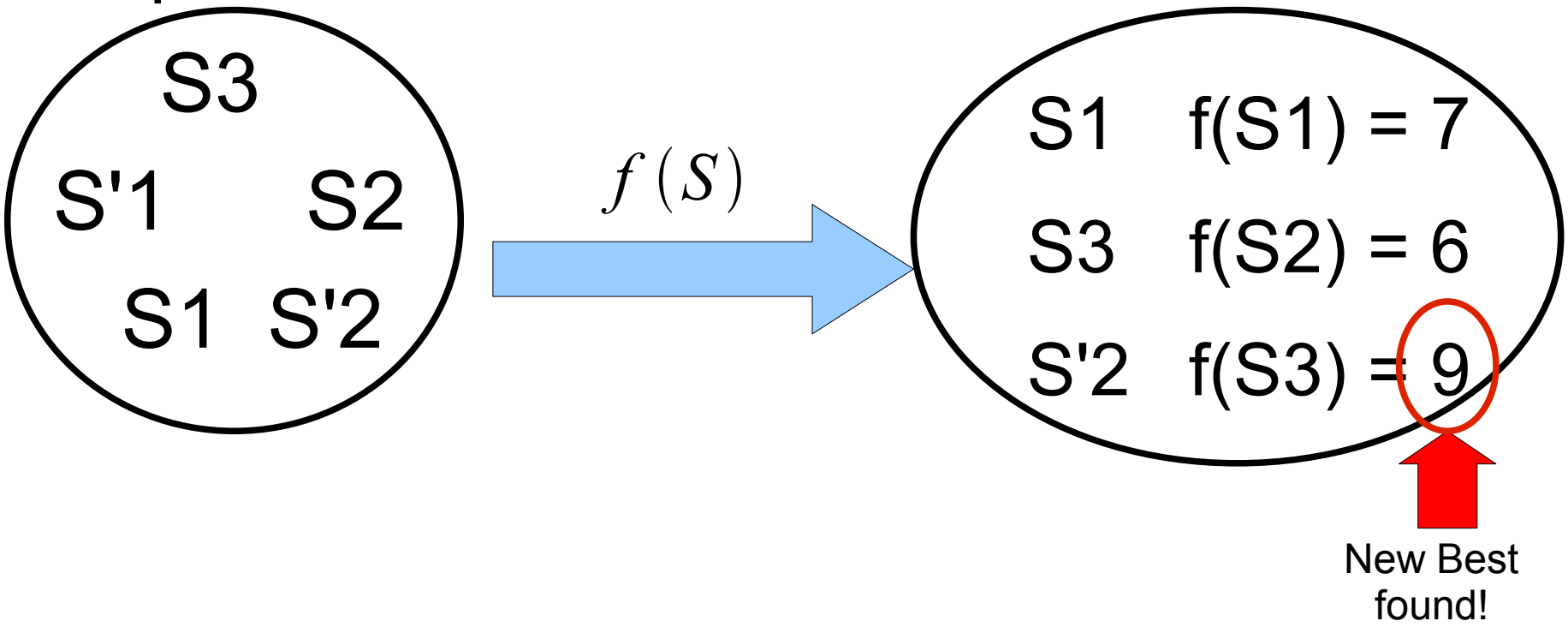
- Example

S'2 = 1111 0 101110 → S'2 = 1111 1 101110

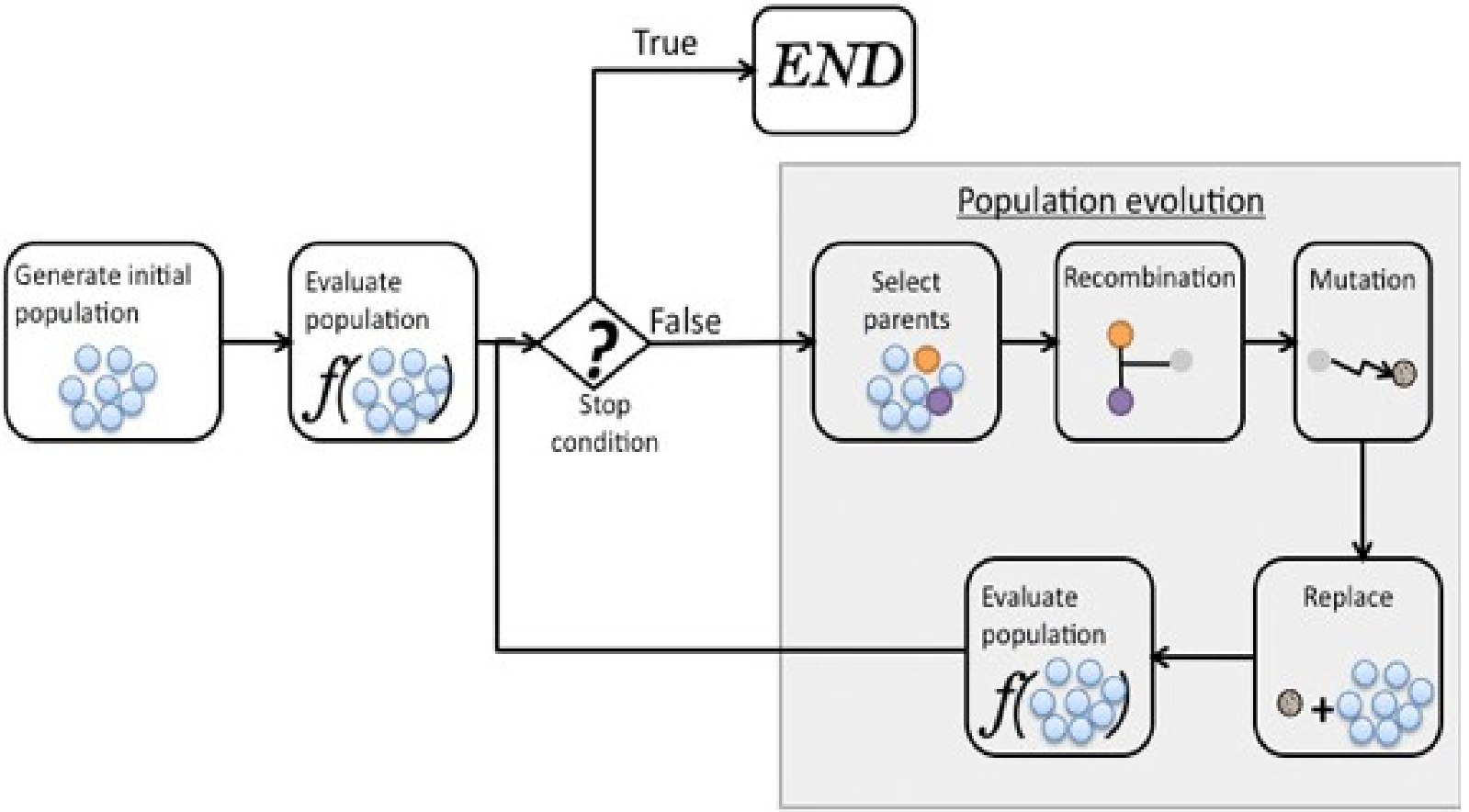


New Population

- Who will survive to the new Population?
  - Union
  - Elitism
- es.Top 3



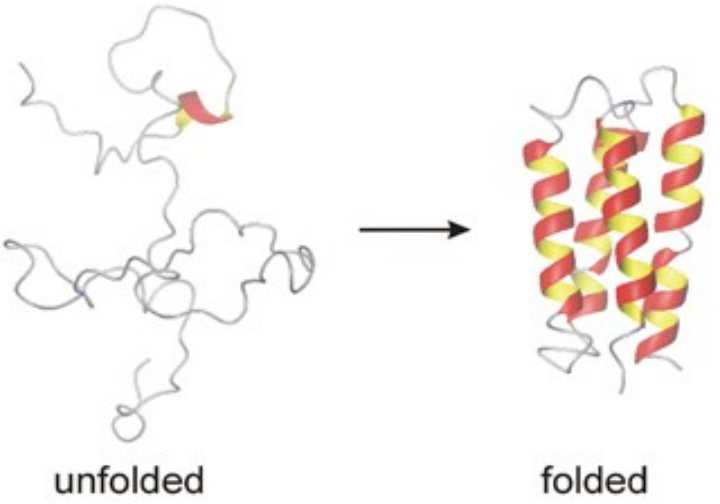
Summary Recap



Application in Bioinformatics

- Many fields such as
  - Multiple Sequence Alignment
  - Protein Folding
  - Ligand Docking

```
 Score =  399 bits (1025), Expect = e-111
 Identities = 198/290 (68%), Positives = 241/290 (82%), Gaps = 1/290

Query: 57  MENFQKVEKIGEGTYGVVYKARNKLTGEVVALKKIRLDTETEGVPSTAIREISLLKELNH 116
           ME ++KVEKIGEGTYGVVYKA +K T E +ALKKIRL+ E EGVPSTAIREISLLKE+NH
Sbjct: 1   MEQYEKVEKIGEGTYGVVYKALDKATNETIALKKIRLEQEDEGVPSTAIREISLLKEMNH 60

Query: 117 PNIVKLLDVIHTENKLYLVFEFLHQDLKKFMDASALTGIPLPLIKSYLFQLLQGLAFCHS 176
            NIV+L  DV+H+E ++YLVFE+L   DLKKFMD+          LIKSYL+Q+L  G+A+CHS
Sbjct: 61  GNIVRLHDVVHSEKRIYLVFEYLDLDLKKFMDSCPEFAKNPTLIKSYLYQILHGVAYCHS 120

Query: 177 HRVLHRDLKPQNLLINTE-GAIKLADFGLARAFGVPVRTYTHEVVTLWYRAPEILLGCKY 235
           HRVLHRDLKPQNLLI+     A+KLADFGLARAFG+PVRT+THEVVTLWYRAPEILLG +
Sbjct: 121 HRVLHRDLKPQNLLIDRRTNALKLADFGLARAFGIPVRTFTHEVVTLWYRAPEILLGARQ 180

Query: 236 YSTAVDIWSLGCIFAEMVTRRALFPGDSEIDQLFRIFRTLGTPDEVVWPGVTSMPDYKPS 295
           YST VD+WS+GCIFAEMV ++ LFPGDSEID+LF+IFR LGTP+E  WPGV+ +PD+K +
Sbjct: 181 YSTPVDVWSVGCIFAEMVNQKPLFPGDSEIDELFKIFRILGTPNEQSWPGVSCLPDFKTA 240

Query: 296 FPKWARQDFSKVVPPLDEDGRSLLSQMLHYDPNKRISAKAALAHPFFQDV 345
           FP+W  QD + VVP LD  G  LLS+ML Y+P+KRI+A+ AL H +F+D+
Sbjct: 241 FPRWQAQDLATVVPNLDPAGLDLLSKMLRYEPSKRITARQALEHEYFKDL 290
```



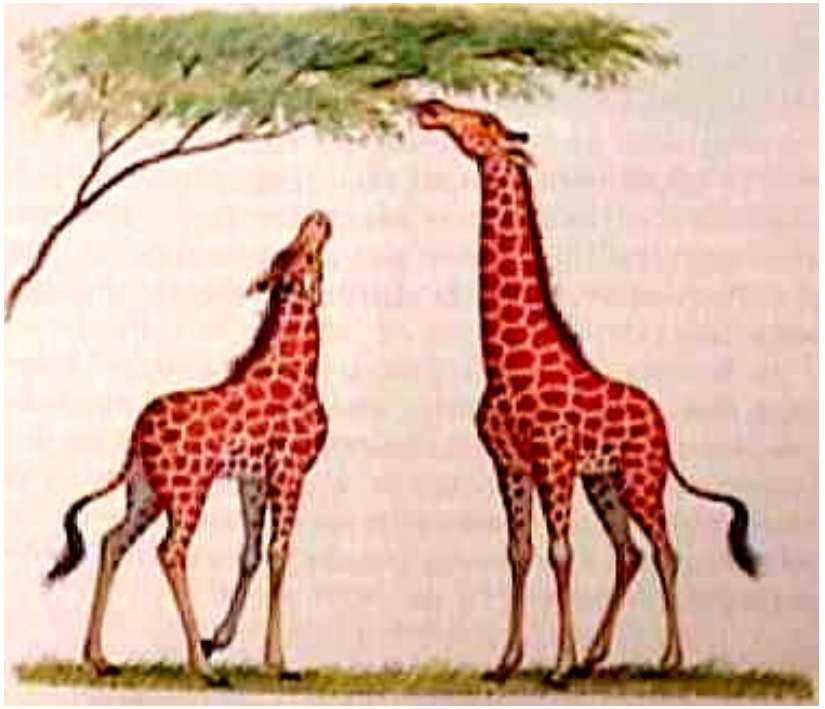
LIGAND DOCKING CASE

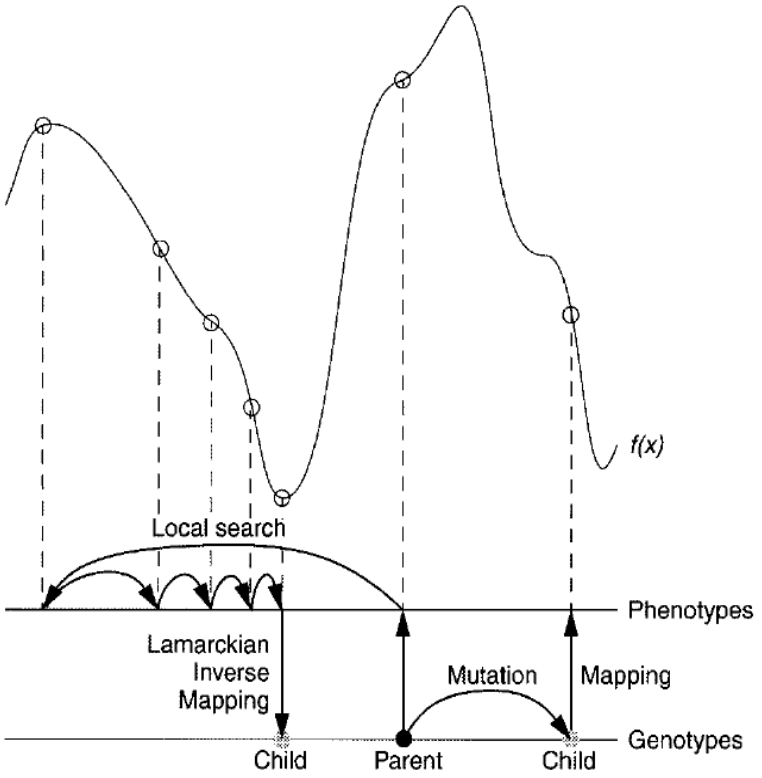
- Problem: Which is the best docking between a Ligand and his Protein?
- Difficult to optimize(entire range of positional, orientational, and conformational possibilities)
- Energy-based Fitness Function

AUTO - DOCK[1]

Lamarckian Genetic Algorithm(LGA)

- Union of traditional GA with Local Search
- Coherent with Lamarck's Evolution Theory





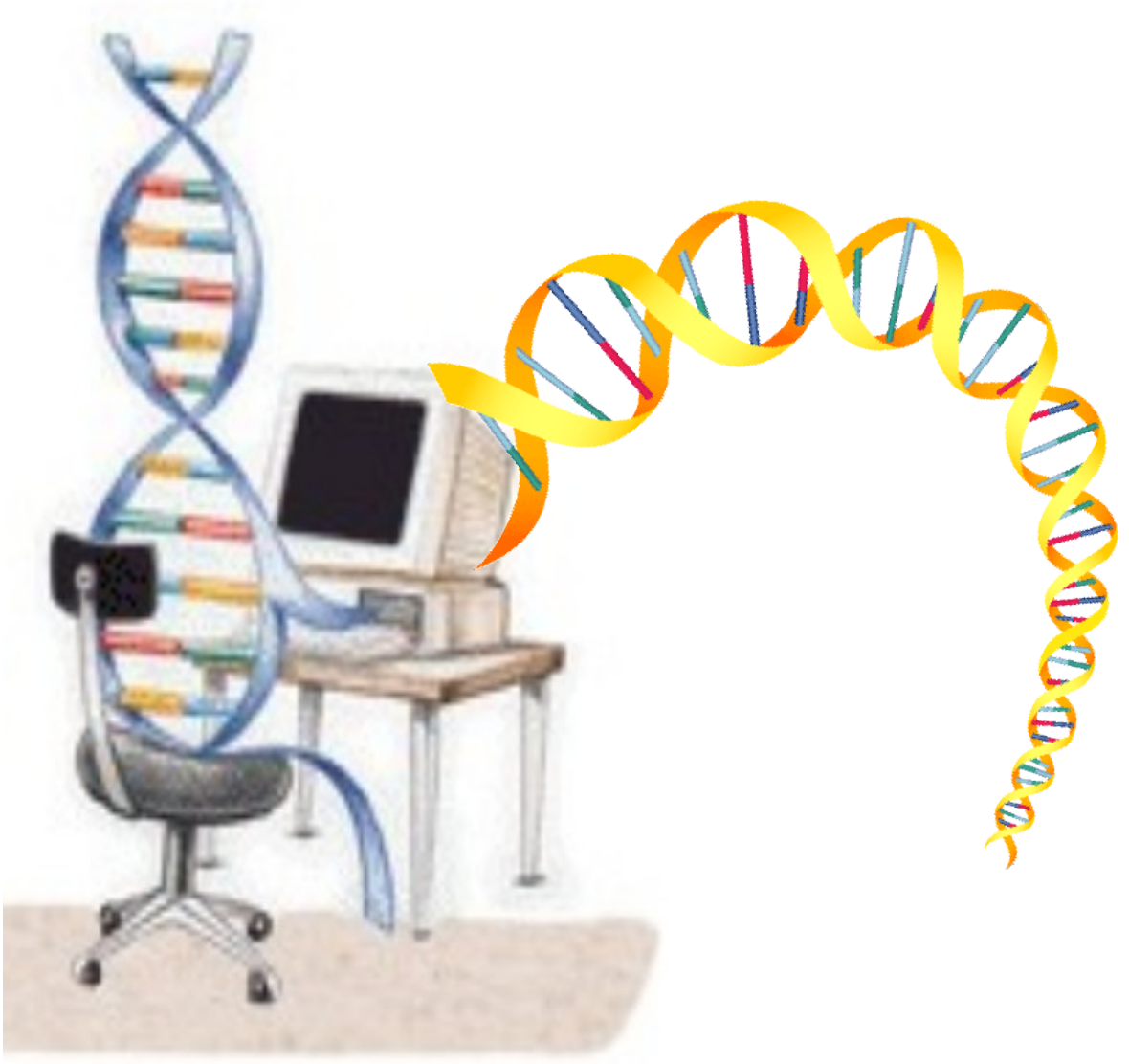
Final Words

- GA solution to every problem? No!
- Hard Tuning(Many parameters to set)
- Like dealing with a “Blind watchmaker”(Dawkins)[2]
- When to use?
  - Huge State Space
  - Independent parell solution search needed
  - Good either than Optimal solution acceptable
  - Last chance:-)

References

- [1] Journal of Computational Chemistry, Vol.19, 1639-1662 (1998) *“Automated Docking Using a Lamarckian Genetic Algorithm and an Empirical Binding Free Energy Function”*
- [2] Richard Dawkins,Norton & Company, Inc(1986) *“The Blind Watchmaker”*

Motif Discovery using Genetic Algorithms

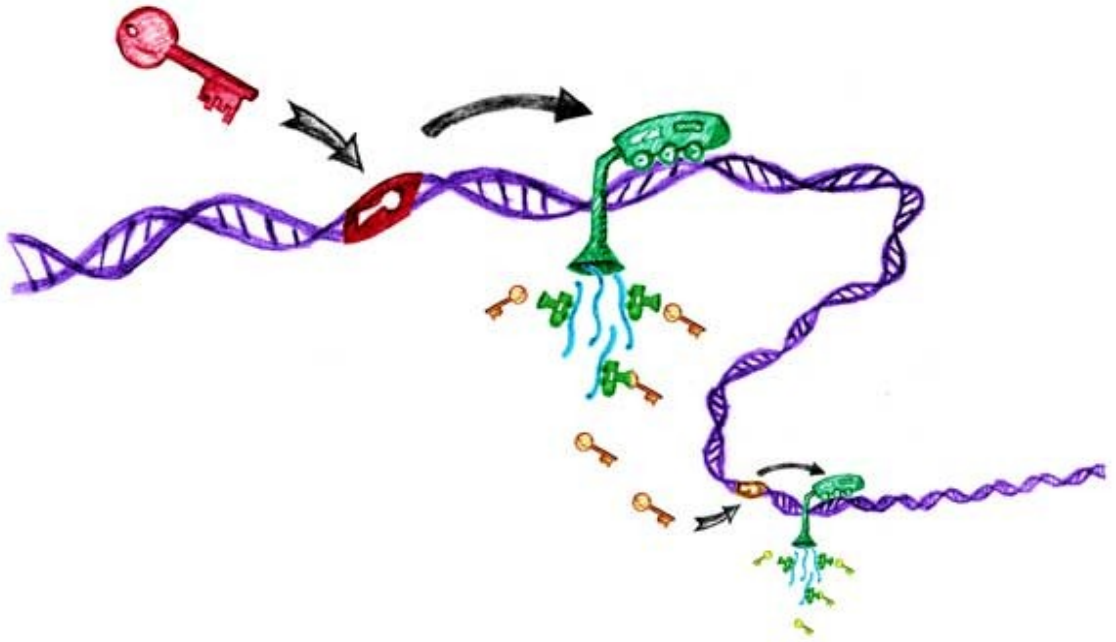


Outline

- Biological Problem of Motif Discovery (MD)
- Short review about Genetic Algorithms (GA)
- Presentation of MDGA
  - Aim
  - Groundings
  - Comparison
- Conclusion

What's a Sequence Motif?

- A nucleotide or amino-acid sequence pattern with biological relevance.
- E.g. A binding site for regulation of transcription

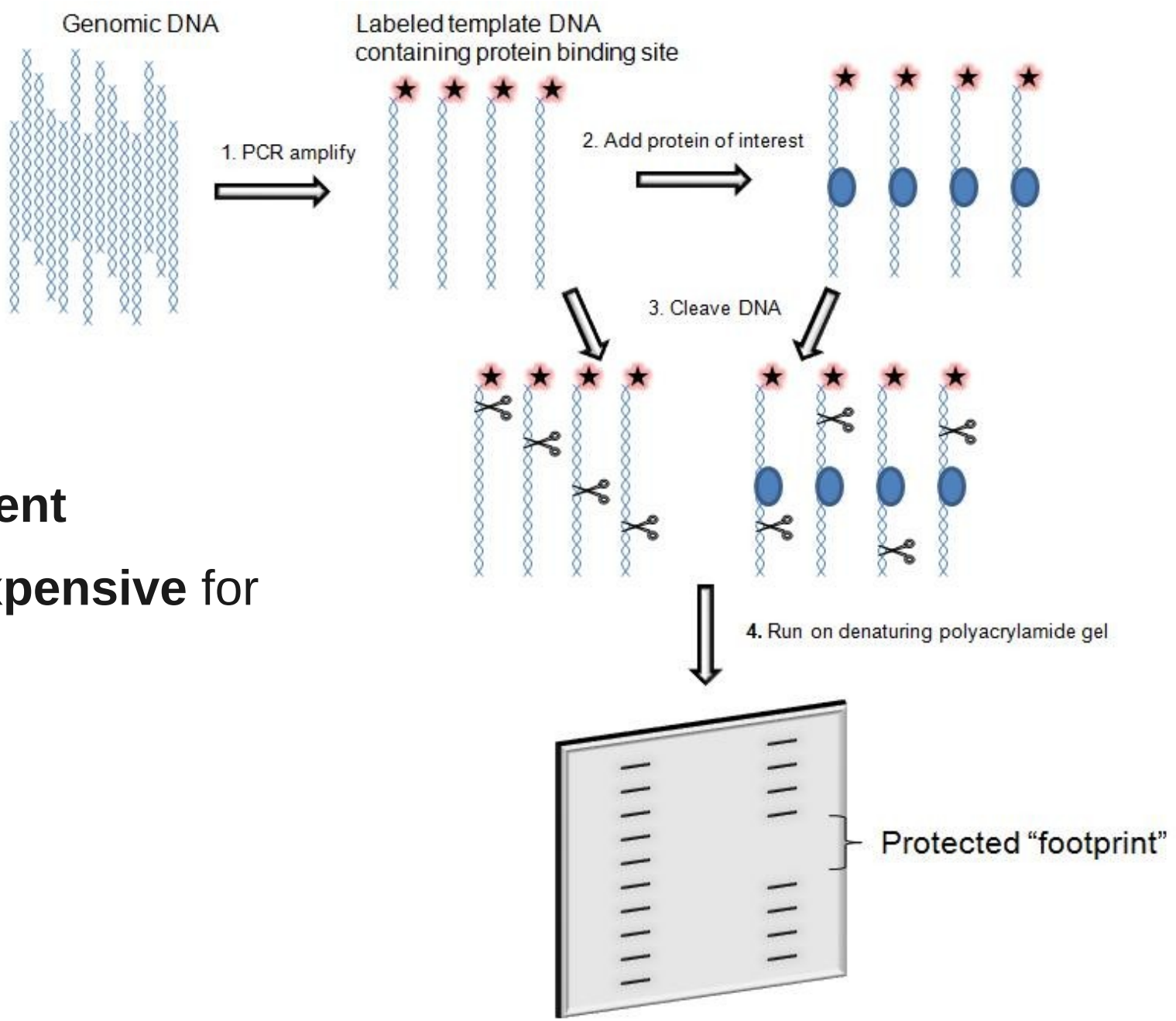


What are we looking for?

- **Exact Sequence Motif**
- **Starting location**
- **Consensus sequences**
  e.g. between homologous sequences which have often similar motifs

Traditional Methods

e.g.
DNA Footprinting technique:



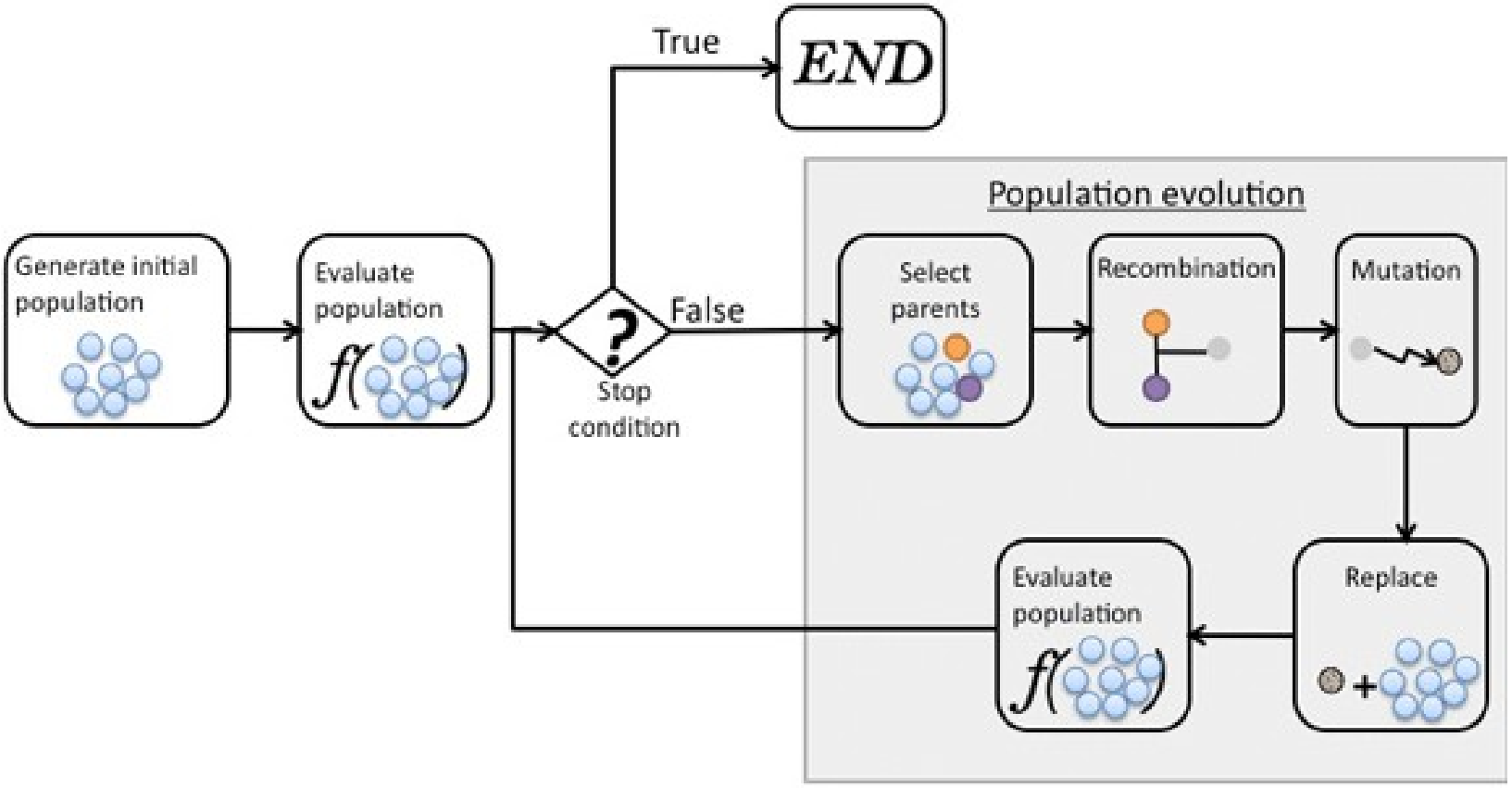
- *Pro*: **Very Efficient**
- *Contra*: **Very Expensive** for large application!

Computational Approach

- Inexpensive BUT complex
- Difficulty arises from the fact that the locations of the binding site can vary significantly on the upstream regions of different homologous genes

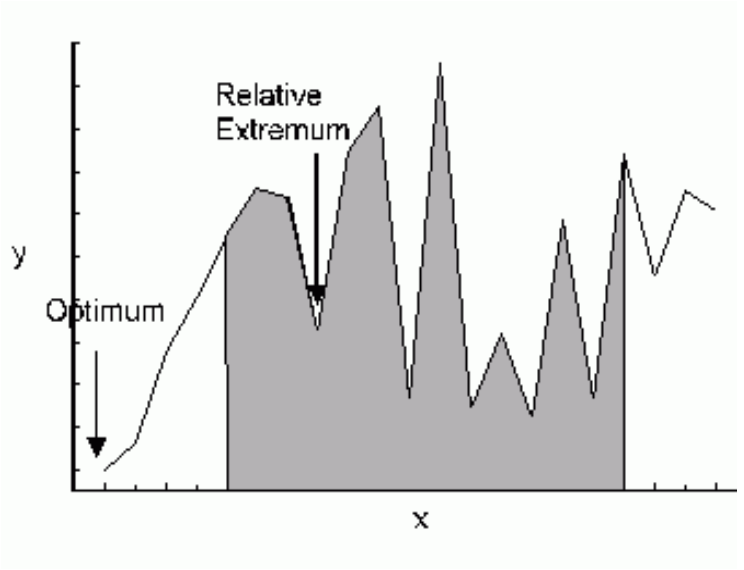
Sequences
S1 : atcATCCGTgtagctcaaaa
S2 : agATCCGTaacgaagtttac
S3 : ccccATCCGTaattacctat
S4 : ggccgacttagccaatcga
S5 : tATCCGTtagATACGTgccga

Short explain GA



Why use GA

- Good for searching through big spaces
- Won't stop easily on local maxim/minim
- Independent parallel solution search



Motif Discovery using a Genetic Algorithm (MDGA)

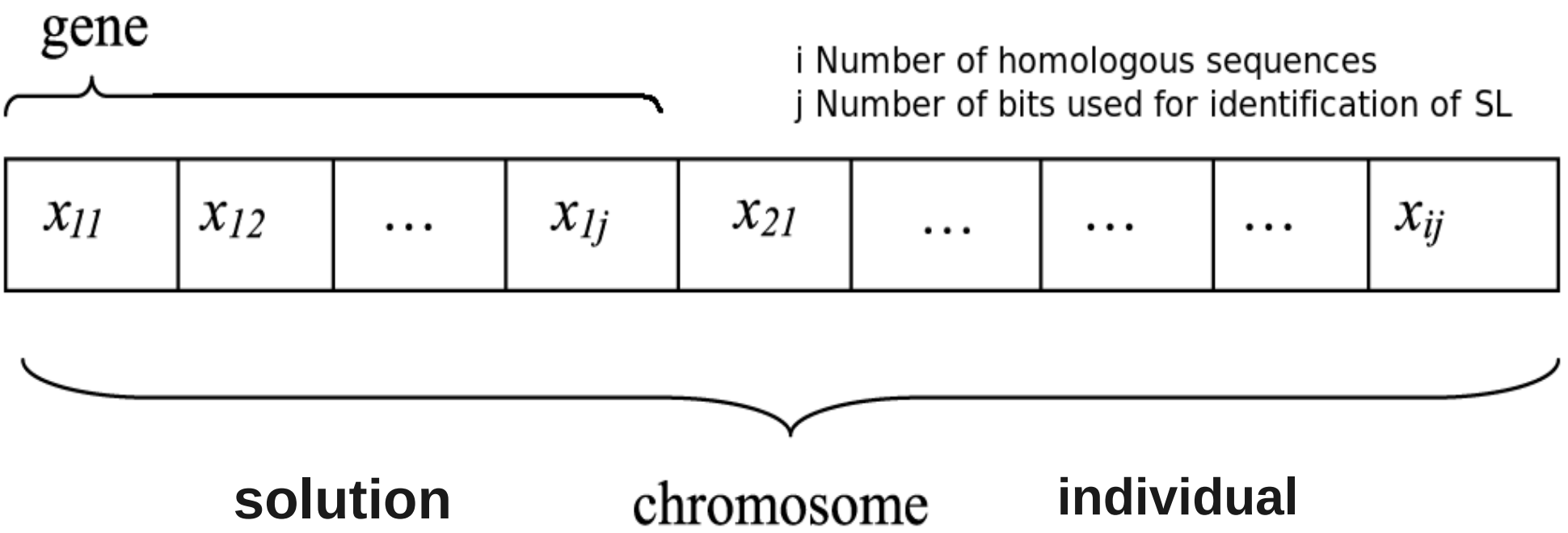
- *Aim:* Identification of Binding Sites in nucleotidic sequences(DNA or RNA) through comparing homologues

- *How:* Exploring the search space of all possible starting locations with a population that undergoes evolution

MDGA

- Representation
  - An individual as set of starting locations(SL)
    e.g SL = 3 → integer(16 bits): 0000000000000011

gene



EVALUATION

- Initial Population randomly selected (e.g. 100)
- Fitness Function should provide a measure of similarity among all motifs defined in an individual

$$fitness = \sum_{i=1}^{W} IC_i$$

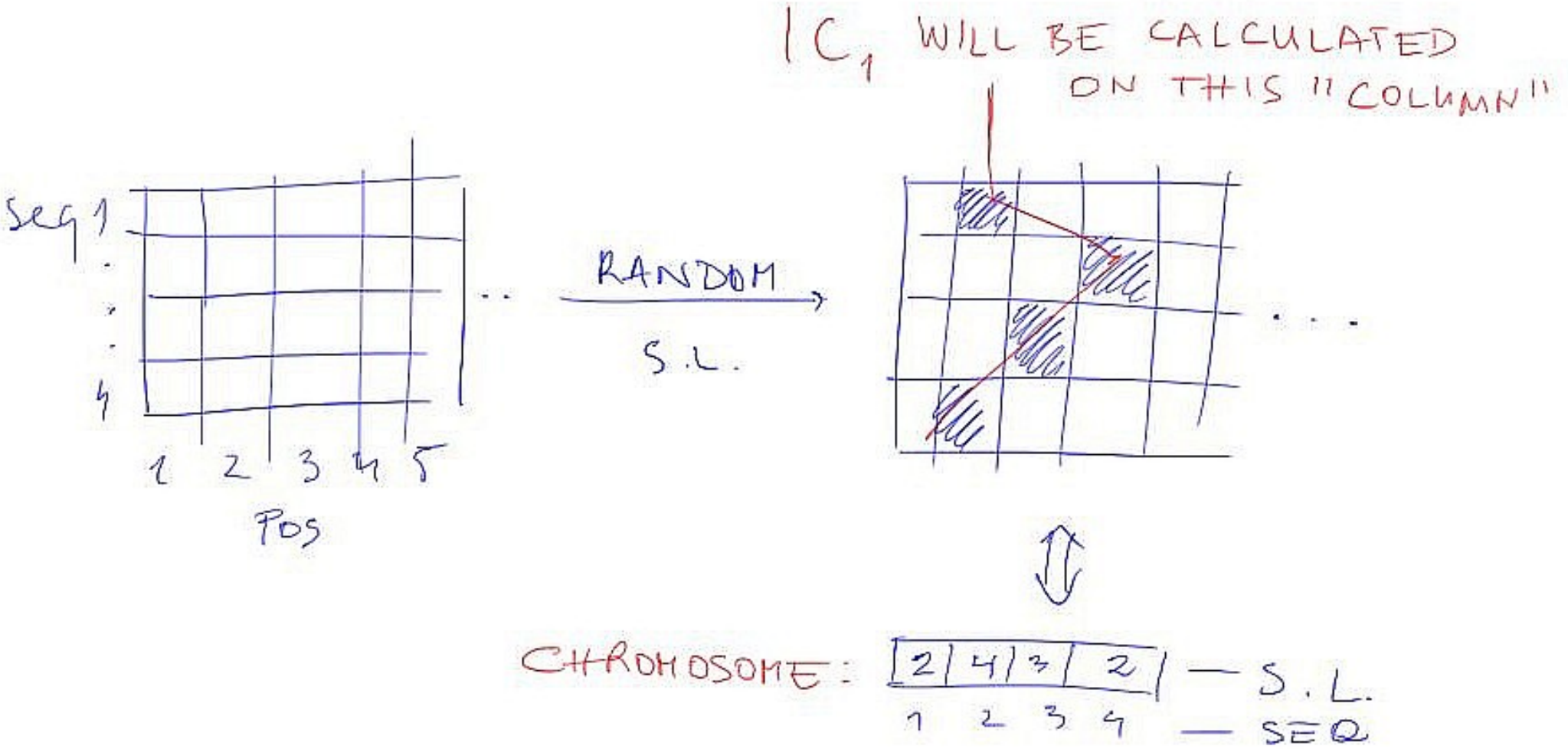
Information Content for single column:

$$IC = \sum f_b \log_2 \frac{f_b}{p_b}$$

Frequency inside column of base b

Background frequency of b

Information Content



ALGORITHM (simplified)

1. *SET UP Parameters*
   SET W = motif width;
   SET G = maximum iteration number;
   SET S = shift range;
   SELECT a crossover strategy;

2. *INITIALISE population with random candidates (vectors of start positions);*

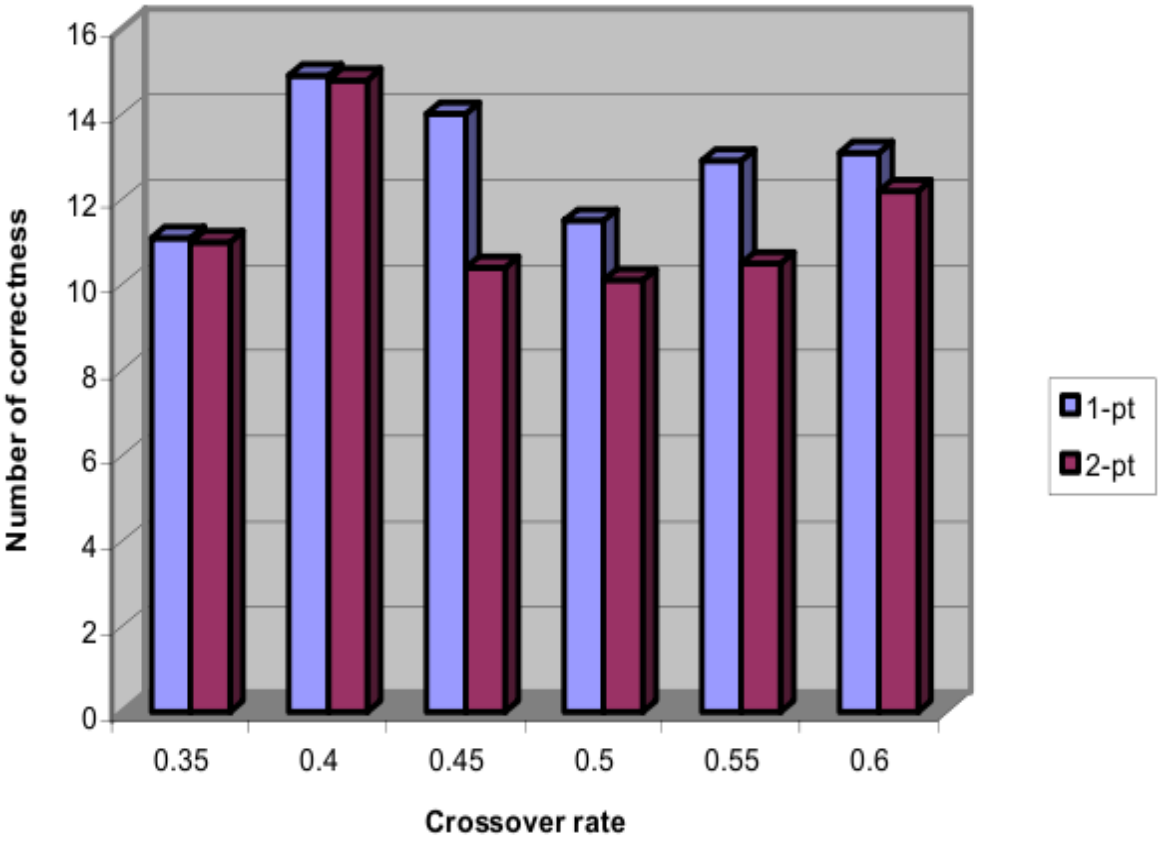
3. *EVALUATE each candidate*

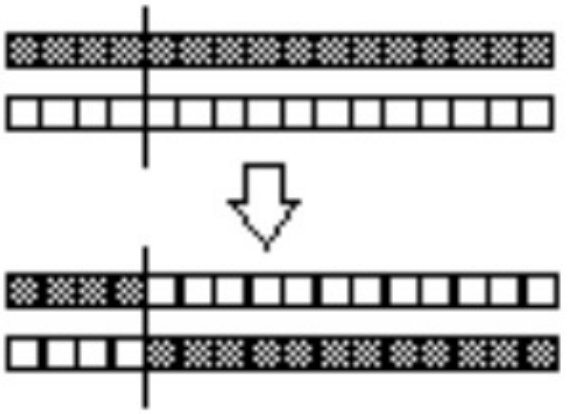
Algorithm(continued)

REPEAT UNTIL (iteration number = G)
{
SELECT parents by roulette wheel selection;
CROSSOVER parents based on crossover strategy picked;
MUTATE the resulting offspring;
EVALUATE new candidates;
REPLACE worst individuals;
}

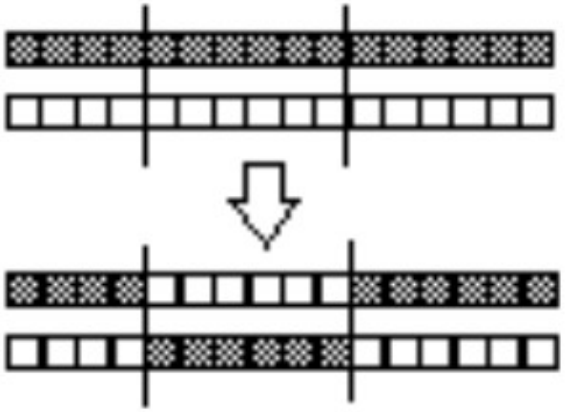
OUTPUT predicted motifs and their consensus motif

Parameters refinement





array one point crossover



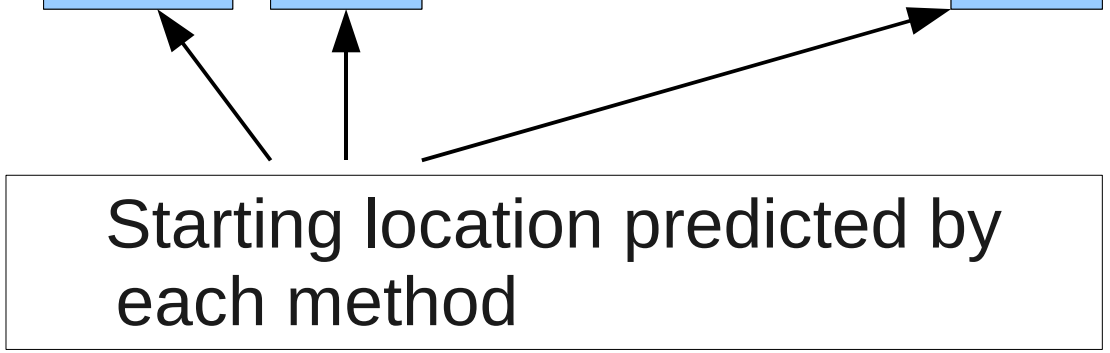
array two point crossover

Experimental evaluation

- CRP(Cyclic-AMP Receptor Protein) Binding site
- Dataset of 18 homologous sequences each of 105bps
- Twenty-three binding sites determined by DNA Footprinting(FP) method. MDGA(GA) compared to BioProspector(BP) (based on Gibbs sampling algorithm)
- Fixed motif width, W = 22

EXPERIMENTAL RESULTS

Sequence	FP	BP	ER	GA	ER
1	17, 61	63	2	62	1
2	17, 55	57	2	56	1
3	76	78	2	77	1
4	63	65	2	64	1
5	50	52	2	51	1
6	7, 60	9	2	8	1
7	42	26	-16	43	1
8	39	41	2	40	1
9	9, 80	11	2	10	1
.					
.					
.					
.					
.					

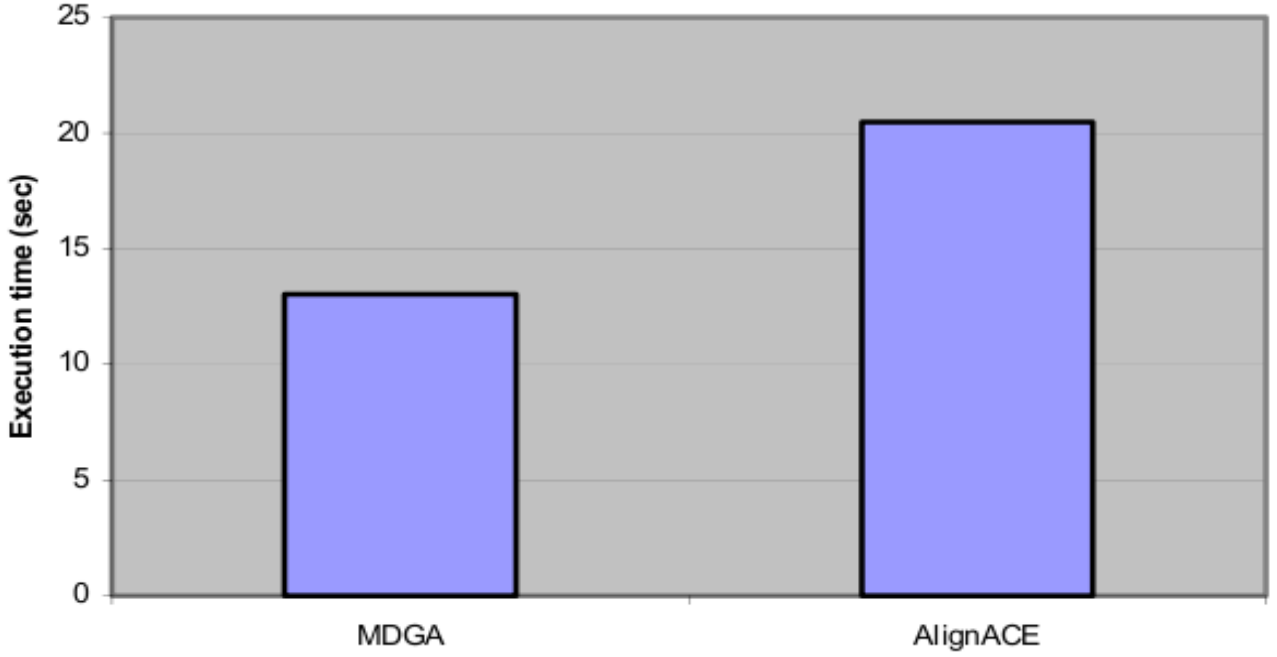


YDR02c Binding Site

Motif-finding program	Predicted motif
AlignACE	TCCGGGTAAA
BioProspector	TACCGGGTAA
Consensus	CCGGGTAAAA
Gibbs Sampler	TATTTTGATG
MEME	GTCCGGGTAA
MDGA	TCCGGGTAAA

Performance

- Time gain up to 50% (t-test)
- Good to Optimal solutions
- Less dependent to size of sequences submitted due to independence of solution search

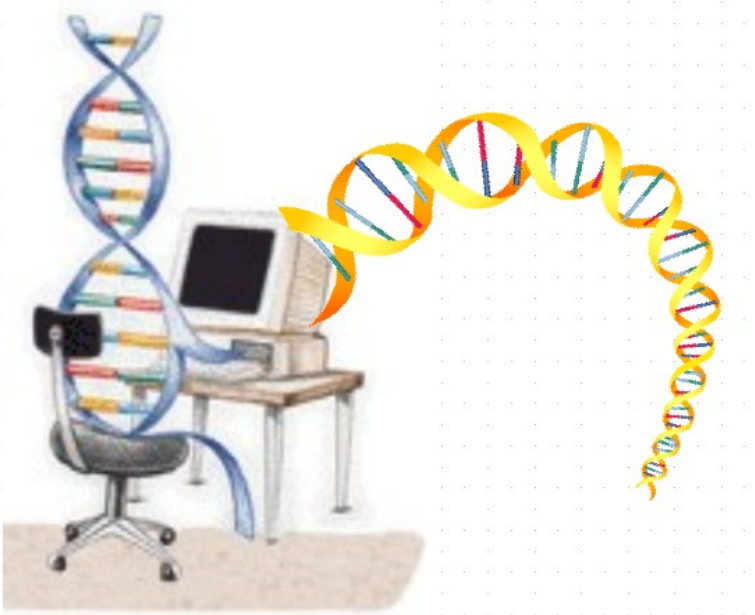


Problems

- Assumption: Every sequence contains A motif
- But in reality even between homologous there can be sequences that have no motif as well as more than one
- Crossing-over and mutation strategies might not be designed for great amount of sequences

Conclusion

- New and effective approach to Motif Discovery followed by many others(FMGA,GAME, MOGAMOD..)
- Curious parallelism between search technique and Object of analysis



References

- Che,Song,Rasheed “*MDGA: Motif Discovery Using A Genetic Algorithm*” 2005, GECCO'05