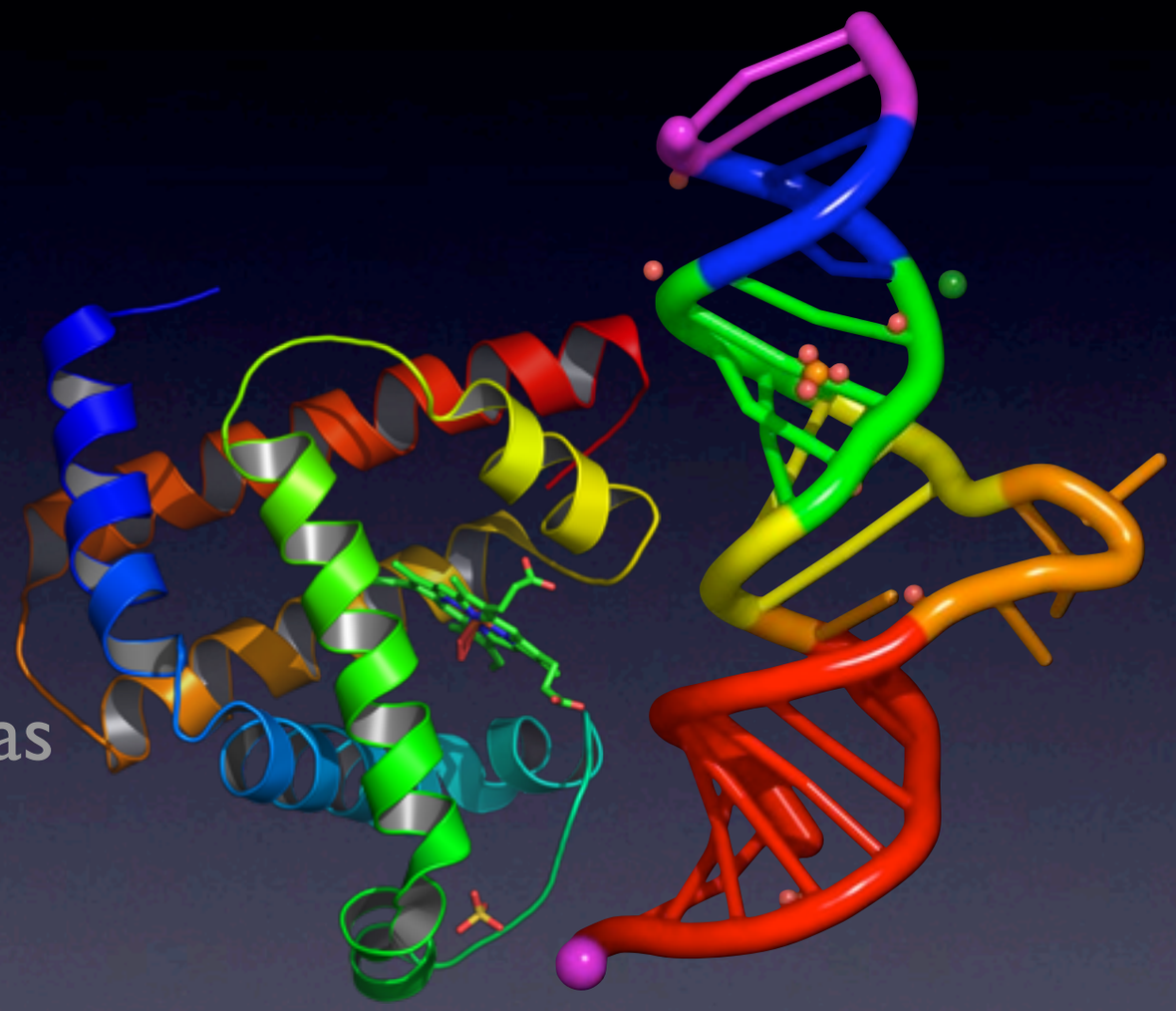
ARN y Proteínas

Estructura Dimensional
Modelos Simplificados de Proteínas



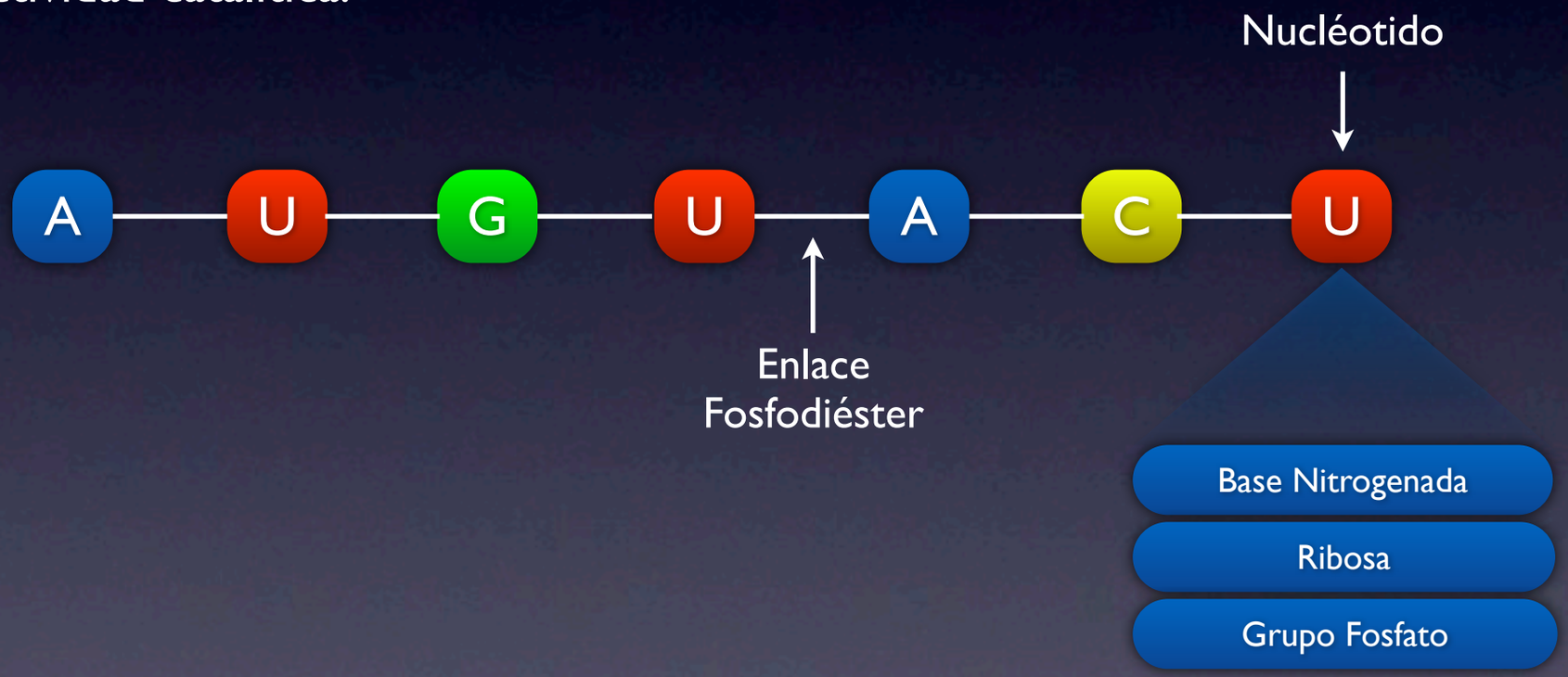
EMILIO HECK OLIVA - IGNACIO MELLA TÉLLEZ

ARN

Introducción

¿Que es el ARN?

- Ácido nucleico formado por una cadena simple de ribonucleótidos.
- Presente tanto en eucariotas como procariotas.
- Dirige etapas intermedias en la síntesis proteica.
- Regulan la expresión genética.
- Actividad catalítica.

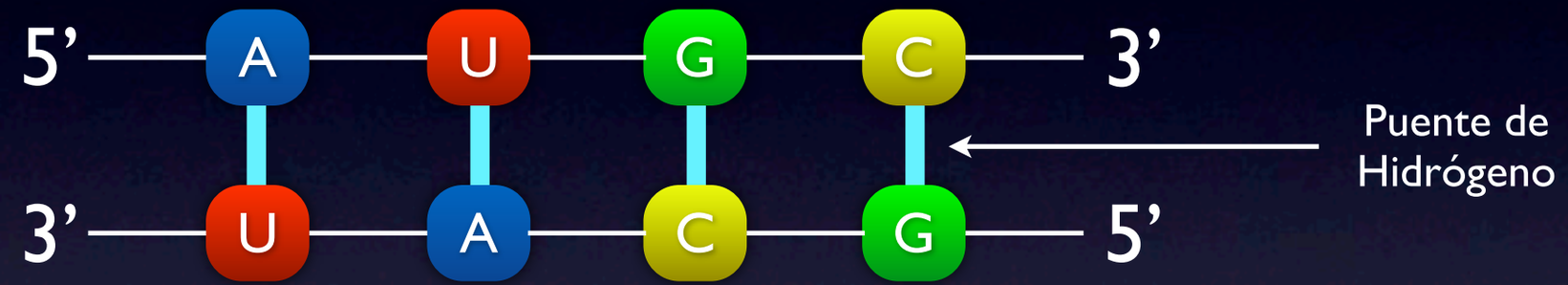


Introducción

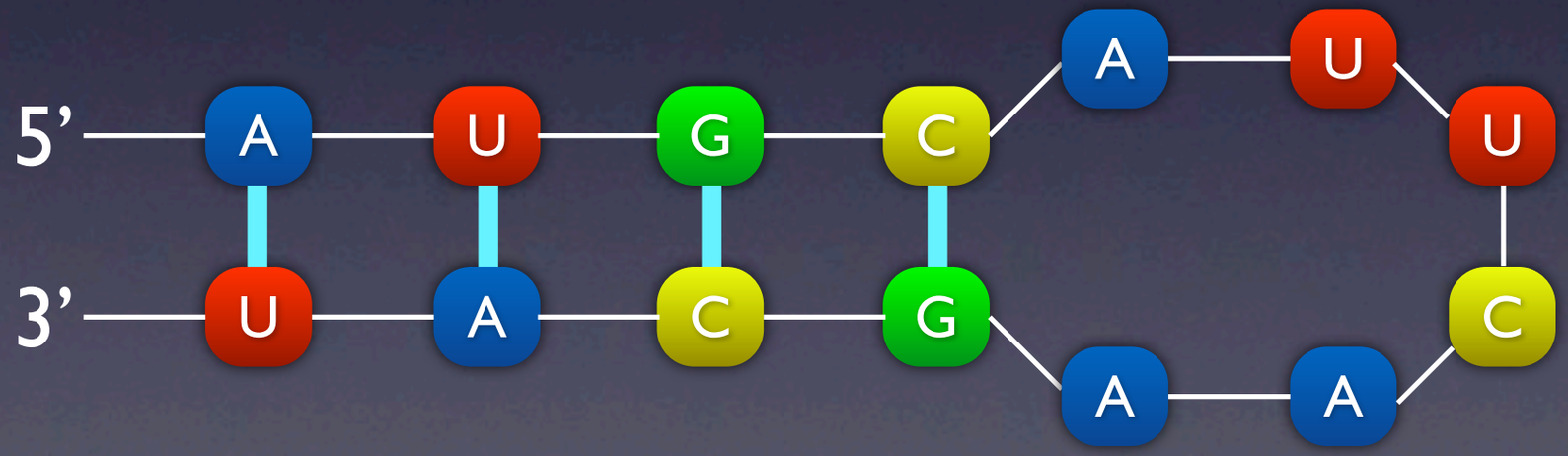
Estructura Secundaria

- Resultado del apareamiento intramolecular de bases.
- Secuencias distantes dentro de la hebra.

Doble Cadena



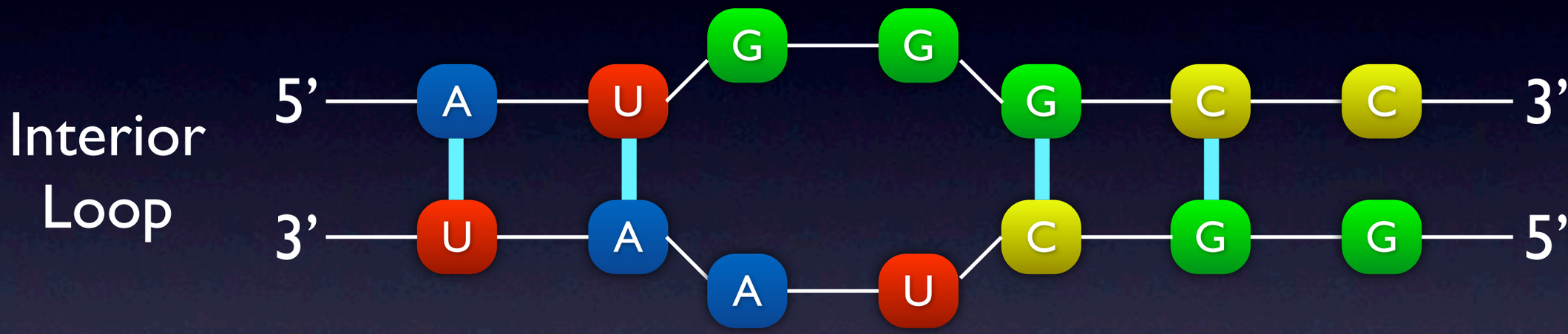
Hairpin Loop

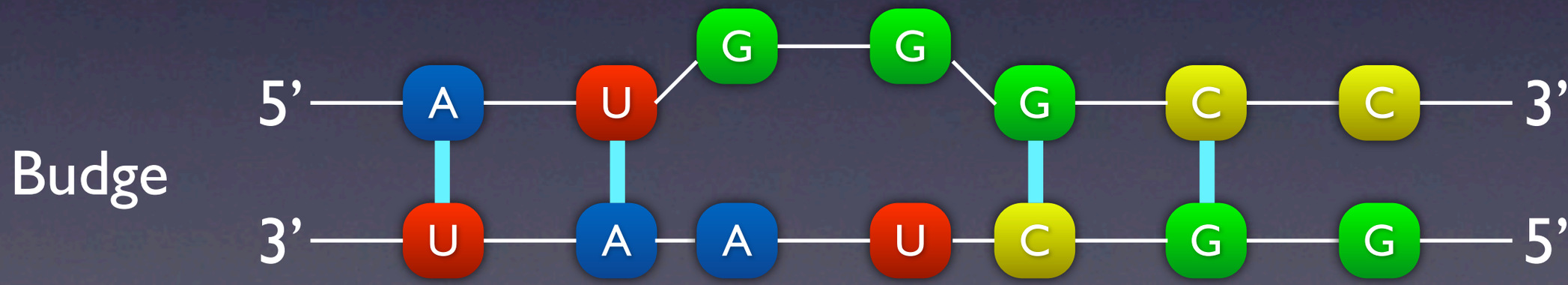


Introducción

Estructura Secundaria

- Resultado del apareamiento intramolecular de bases.
- Secuencias distantes dentro de la hebra.

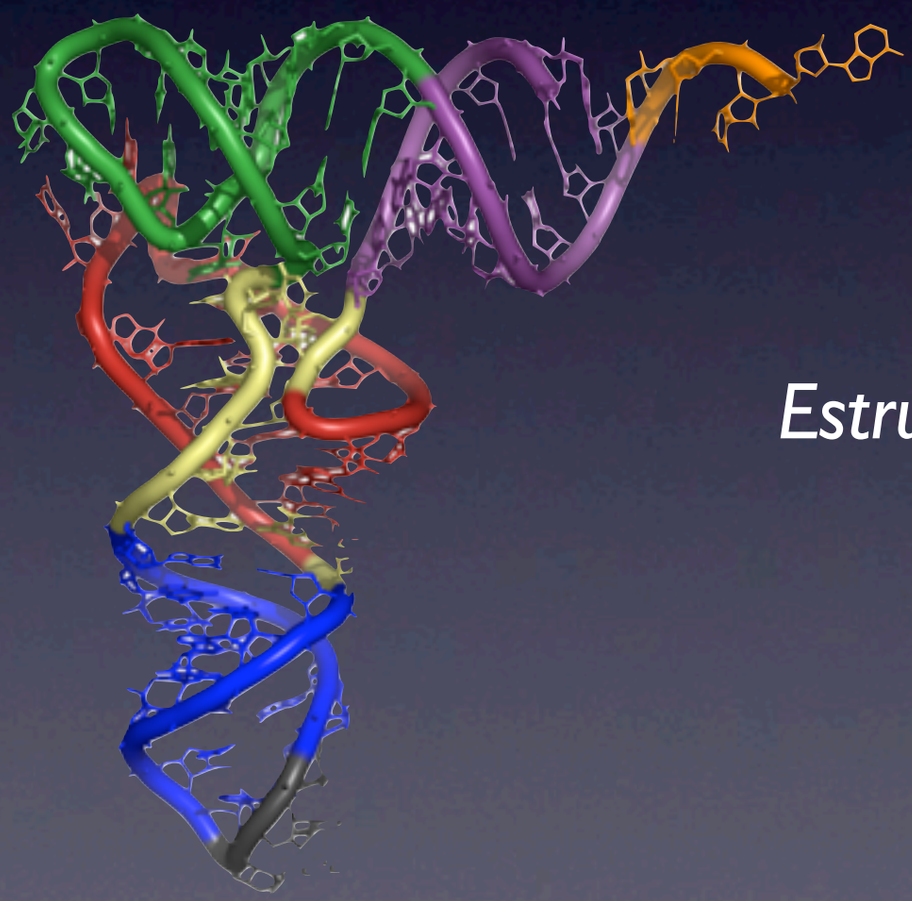




Introducción

Estructura Terciaria

- Hace referencia a la organización que tienen los elementos de la estructura secundaria en el espacio.
- Contactos de Van der Waals.
- Formación de puentes de hidrógeno entre pares de bases (Watson-Crick o inusuales).



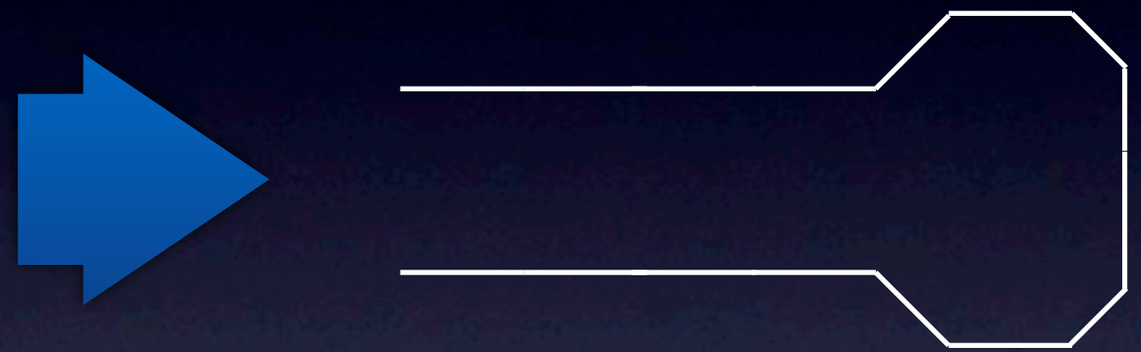
Estructura Terciaria
tRNA

Algoritmos de predicción

Idea

- Predecir la estructura secundaria de una molécula de ARN a partir de su cadena de nucleótidos.

ACUGCUACGT



Algoritmo de Nussinov o Pair Base Maximization

- Algoritmo de programación dinámica.
- Busca la estructura que contenga la mayor cantidad de pares de bases.
- Utiliza una matriz de puntuación.
- Luego realiza un traceback, entregando la estructura secundaria.

Algoritmos de predicción

Algoritmo de Nussinov

- *Sea S una secuencia de RNA de tamaño L*
- *Sea* γ *la matriz de folding correspondiente a la secuencia S*

Inicialización

$$\gamma(i,j) = 0 \quad for\ i = 1\ to\ L$$
$$\gamma(i,i-1) = 0 \quad for\ i = 2\ to\ L$$

Relleno

for $i = 2$ **to** L **do**
  for $j = n$ **to** L **do**
    $Set\ i = j - n + 1$

$$Set \quad \gamma(i,j) = max \begin{cases} \gamma(i+1,j), \\ \gamma(i,j-1), \\ \gamma(i+1,j-1)+1, \\ max_{i \le k \le j}[\gamma(i,k) + \gamma(k+1,j)] \end{cases}$$

Algoritmos de predicción

Algoritmo de Nussinov

Traceback

```
if i < j then
    if γ(i,j) = γ(i+1,j) then
        traceback(i+1,j)
    else if γ(i,j) = γ(i,j-1)  then
        traceback(i+1,j-1)
    else if γ(i,j) = γ(i+1,j-1)+1  then
        traceback(i+1,j-1)
    else for k = i+1 to j-1  do
        if  γ(i,j) = γ(i,k) + γ(k+1,j)  then
            traceback(i,k)
            traceback(k+1,j)
            break
end
```

Algoritmos de predicción

Algoritmo de Nussinov

Cadena Ejemplo : GGGAAAUCC

Inicialización

	G	G	G	A	A	A	U	C	C
G	0								
G	0	0							
G		0	0						
A			0	0					
A				0	0				
A					0	0			
U						0	0		
C							0	0	
C								0	0

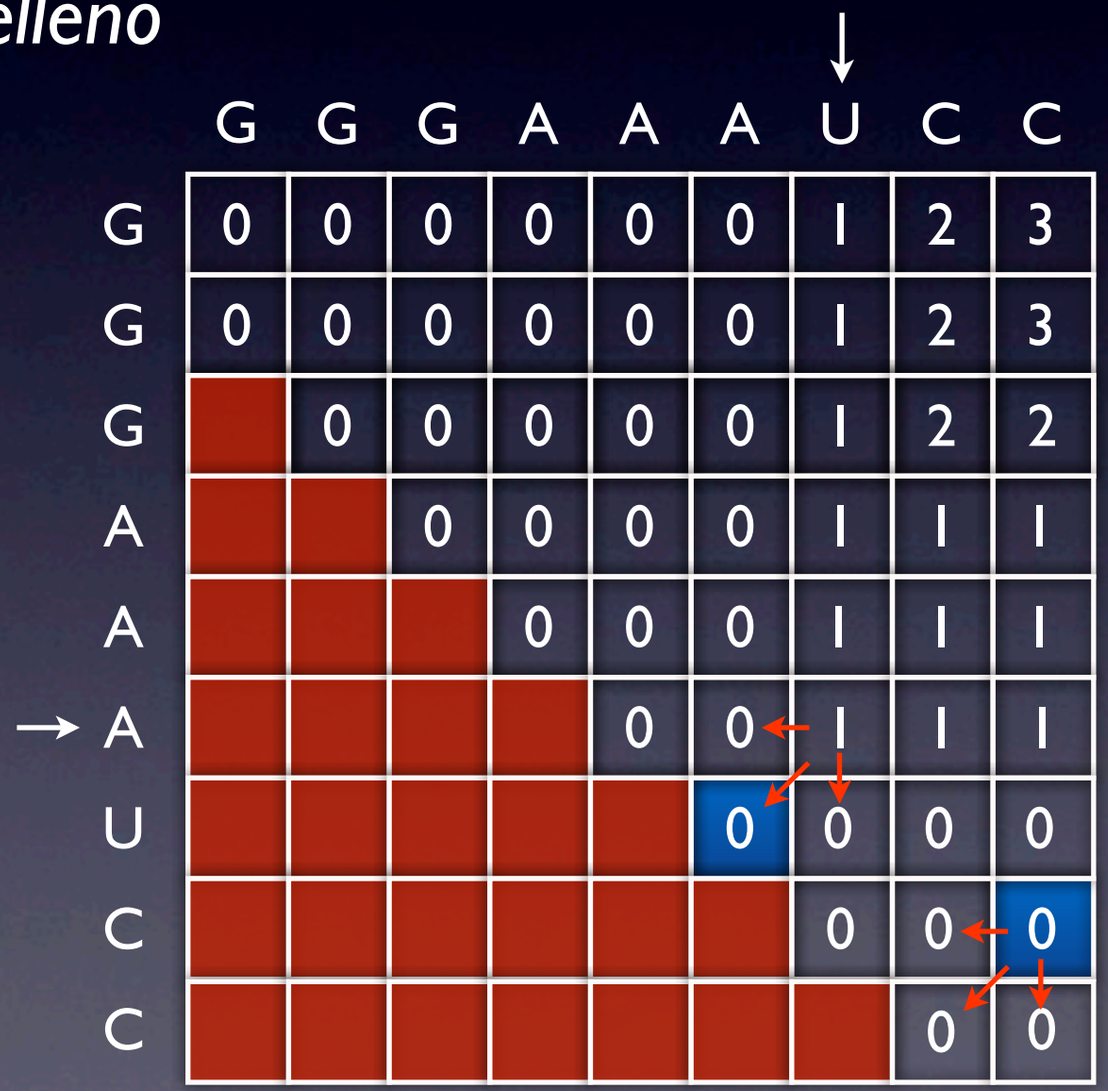
$$\gamma(i,j) = 0 \quad for\ i = 1\ to\ L$$
$$\gamma(i,i-1) = 0 \quad for\ i = 2\ to\ L$$

Algoritmos de predicción

Algoritmo de Nussinov

Cadena Ejemplo : GGGAAAUCC

Relleno



$$Set \quad \gamma(i,j) = max \begin{cases} \gamma(i+1,j), \\ \gamma(i,j-1), \\ \gamma(i+1,j-1)+1, * \\ max_{i \le k \le j}[\gamma(i,k)+\gamma(k+1,j)] \end{cases}$$

* Si las bases son complementarias

Algoritmos de predicción

Algoritmo de Nussinov

Cadena Ejemplo : GGGAAAUCC

Traceback

	G	G	G	A	A	A	U	C	C
G	0	0	0	0	0	0	I	2	3
G	0	0	0	0	0	0	I	2	3
G		0	0	0	0	0	I	2	2
A			0	0	0	0	I	I	I
A				0	0	0	I	I	I
A					0	0	I	I	I
U						0	0	0	0
C							0	0	0
C								0	0

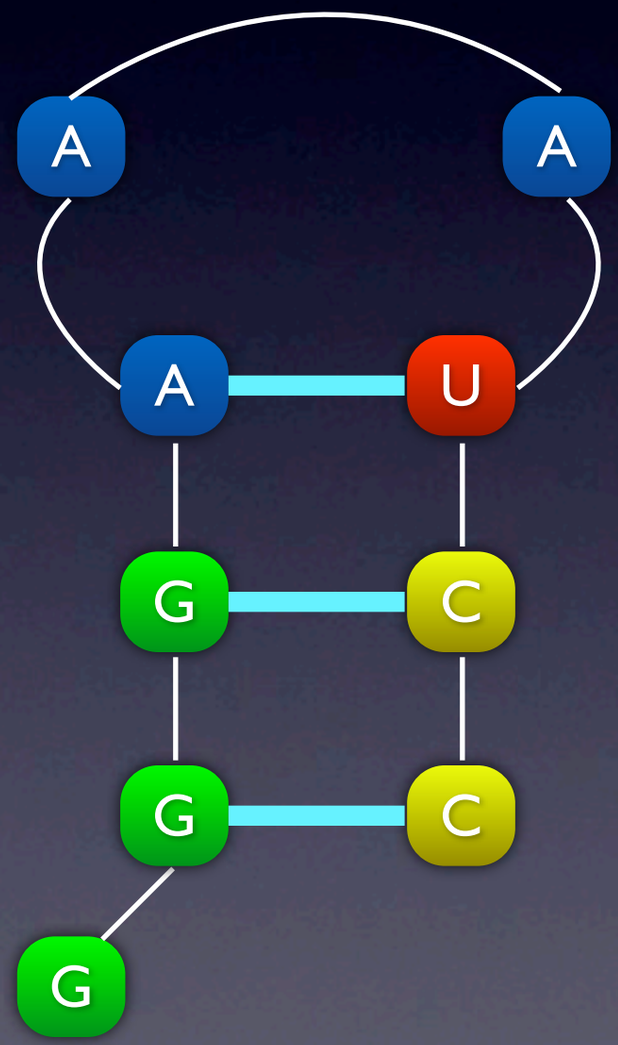
if $i < j$ **then**
  if $\gamma(i,j) = \gamma(i+1,j)$ **then**
    traceback(i+1,j)
  else if $\gamma(i,j) = \gamma(i,j-1)$ **then**
    $\text{traceback}(i+1, j-1)$
  else if $\gamma(i,j) = \gamma(i+1,j-1)+1$ **then**
    traceback(i+1,j-1)
  else for $k = i+1$ to $j-1$ **do**
    if $\gamma(i,j) = \gamma(i,k) + \gamma(k+1,j)$ **then**
      $\text{traceback}(i,k)$
      $\text{traceback}(k+1,j)$
      break
end

Algoritmos de predicción

Algoritmo de Nussinov

Cadena Ejemplo : GGGAAAUCC

Resultado



Algoritmos de predicción

Algoritmo de Nussinov

Inconvenientes

- Este método no necesariamente entregará la estructura más estable. Puede generar estructuras con muchos interior loops o hairpins que son energéticamente desfavorable.

- Tiende a compararse con un alineamiento de secuencias cuando los matches están muy dispersos.

Algoritmos de predicción

Simple Energy Minimization

- Mejores predicciones se obtienen al minimizar la siguiente función de energía para una secuencia de RNA x y un set de bases P

$$E(x, P) = \sum_{(i,j) \in P} e(x_i, x_j)$$

- Donde $e(x_i, x_j)$ es la cantidad de *energía libre* asociada con el par (x_i, x_j)
- Los valores a 37°C son -3, -2 y -1 kcal/mol para los pares C-G , A-U, G-U respectivamente.
- Usando esta función se generaliza el algoritmo de Nussinov, utilizando la cantidad de energía libre en vez de simplemente sumar 1 cuando las bases son complementarias.

$$E(i,j) = min \begin{cases} E(i+1, j) \\ E(i, j-1) \\ E(i+1, j-1) + e(x_i, x_j) \\ min_{i<k<j}[E(i,k) + E(k+1, j)] \end{cases}$$

Algoritmos de predicción

Simple Energy Minimization

Inconvenientes

- Este método no produce buenas predicciones de estructuras ya que no toma en cuenta que los stacks de pares de bases tienen un efecto estabilizador y donde los loops tienen un efecto desestabilizador.

Algoritmos de predicción

Algoritmo de Zuker

- Sea $x = (x_1, x_2, ..., x_L)$ un string en el alfabeto $\Sigma = \{A, G, C, U\}$
- Para $i < j$, donde $W(i, j)$ es la mínima energía de folding de todoslos foldings no-vacios de la subsequencia $x_i, \ldots, x_j$

- Adicionalmente, $V(i, j)$ denota la mínima energía de folding de todos los foldings no-vacios de la subsecuencia $x_i, \ldots, x_j$, *que contienen el par de bases* (i, j)
- Ha de cumplirse

$$W(i, j) \leq V(i, j) \text{ para todo } i, j$$

- Ambas matrices son inicializadas de la siguiente manera

$$W(i, j) = V(i, j) = \infty \text{ para todo } i, j \text{ con } j - 4 < i < j.$$

- Así forzamos dos pares de bases complementarias estén por lo menos 3 posiciones alejadas unas de otras.

Algoritmos de predicción

Algoritmo de Zuker

Definición de Energías

- Es necesario definir funciones de energía relativas a cada tipo de loop
- Sea $eh(i,j)$ la energía asociada a un Hairpin Loop cerrado por el par base
- Sea $es(i,j)$ la energía al par apilado y $(i+1,j-1)$
- Sea $ebi(i,j,i',j')$ la energía de un Bulge o Interior Loop que está cerrado por (i,j) con (i',j') accesible desde (i,j)
- Sea a un término constante asociado a un Multi-Loop.

Algoritmos de predicción

Algoritmo de Zuker

La recursión principal

$$W(i,j) = min \begin{cases} W(i+1,j) \\ W(i,j-1) \\ V(i,j) \\ min_{i<k<j}[W(i,k)+W(k+1,j)] \end{cases}$$

$$V(i,j) = min \begin{cases} eh(i,j) \\ es(i,j)+V(i+1,j-1) \\ VBI(i,j) \\ VM(i,j) \end{cases}$$

$$VBI(i,j) = min_{i<i'<j'<j/i'-i+j-j'>2}[ebi(i,j,i',j')+V(i',j')]$$

$$VM(i,j) = min_{i<k<j-1}[W(i+1,k)+W(k+1,j-1)]+a$$

Algoritmos de predicción

Algoritmo de Zuker

La recursión principal

$$W(i,j) = min \begin{cases} W(i+1,j) & \text{(a)} \\ W(i,j-1) & \text{(b)} \\ V(i,j) & \text{(c)} \\ min_{i<k<j}[W(i,k) + W(k+1,j)] & \text{(d)} \end{cases}$$

- Se consideran 4 posibilidades
  - (a) *i* no está pareado
  - (b) *j* no está pareado
  - (c) *i* y *j* están pareados entre si
  - (d) *i y j* están pareados pero no necesariamente entre si

Algoritmos de predicción

Algoritmo de Zuker

La recursión principal

$$V(i,j) = min \begin{cases} eh(i,j) & \text{(a)} \\ es(i,j) + V(i+1, j-1) & \text{(b)} \\ VBI(i,j) & \text{(c)} \\ VM(i,j) & \text{(d)} \end{cases}$$

- Se consideran distintas situaciones cuando las bases *i* y *j* están pareadas
  - (a) Cerrar un hairpin loop
  - (b) Cerrar una doble cadena
  - (c) Cerrar un bulge o interior loop
  - (d) Cerrar un multi-loop

Algoritmos de predicción

Algoritmo de Zuker

La recursión principal

$$VBI(i,j) = min_{i<i'<j'<j/i'-i+j-j'>2}[ebi(i,j,i',j') + V(i',j')]$$

- Se toman en cuenta todas las formas posibles para definir un bulge o interior loop que involucre al par base *(i ', j ')* y es cerrado por *(i , j)*.
- En cada situación, se tiene una contribución de un bulge o interior loop y una contribución de la estructura que esta al lado opuesto de *(i ', j ')*

Algoritmos de predicción

Algoritmo de Zuker

La recursión principal

$$VM(i,j) = min_{i<k<j-1}[W(i+1,k) + W(k+1,j-1)] + a$$

- Se consideran diferentes formas de obtener un multi-loop a partir de dos estructuras más pequeñas y se añade una contribución de *a* para cerrar el loop.

Algoritmos de predicción

Algoritmo de Zuker

Análisis de Tiempo

- La computación de
  - *W* toma $\Theta(L^3)$
  - *V* toma $\Theta(L^2)$
  - *VBI* toma $\Theta(L^4)$
  - *VM* toma $\Theta(L^3)$
- Por lo tanto el tiempo de ejecución total es del orden de $\Theta(L^4)$

Algoritmos de predicción

Algoritmo de Zuker

Software que lo utiliza

MFold

http://mfold.burnet.edu.au/rna_form

Vienna RNA

http://rna.tbi.univie.ac.at/cgi-bin/RNAfold.cgi

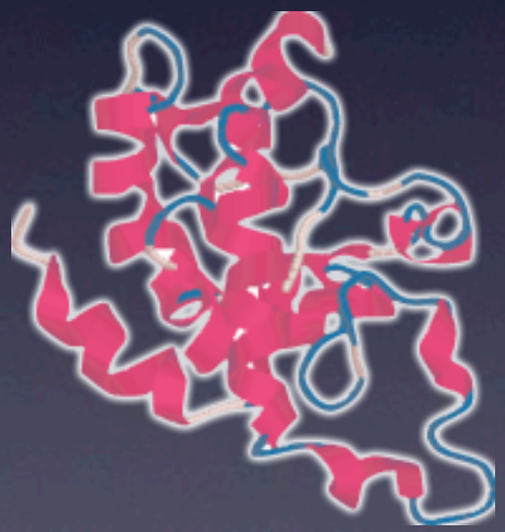
Proteínas

Modelos 3D de Proteínas

Globular

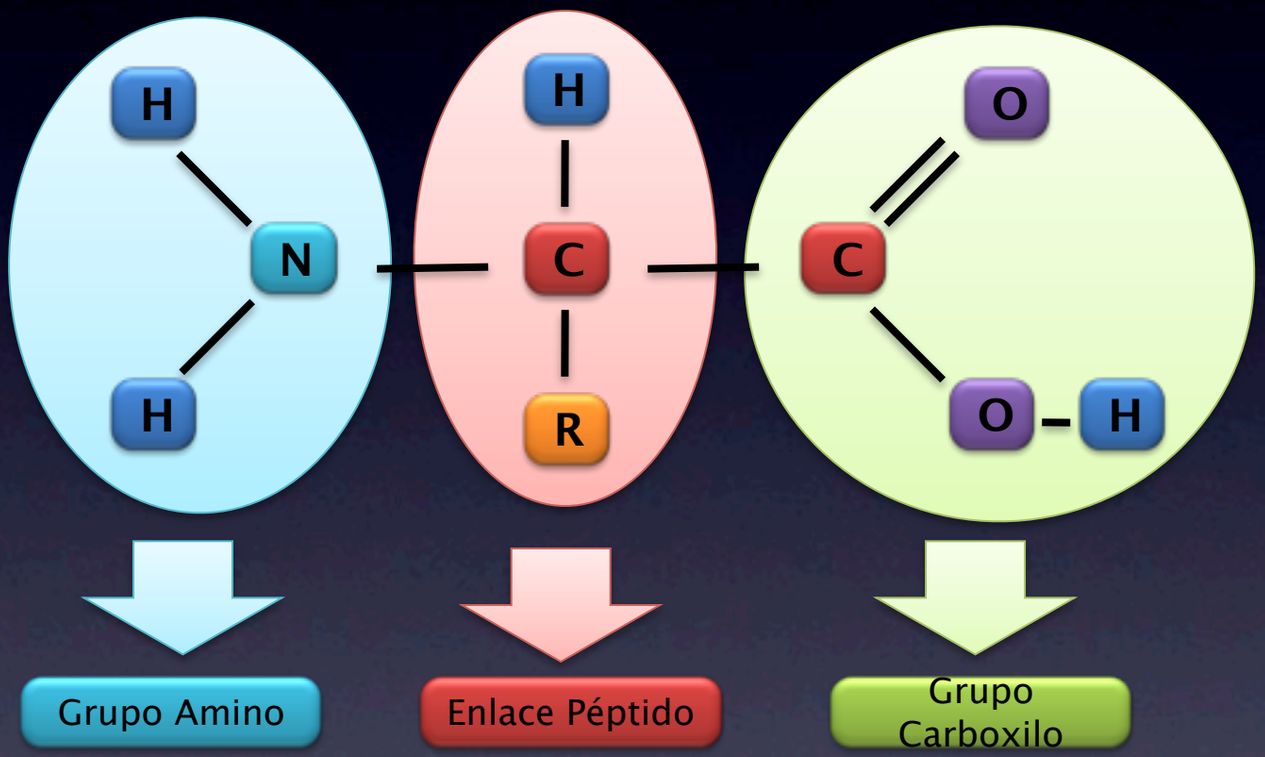
Hélice Alfa

Hoja Beta

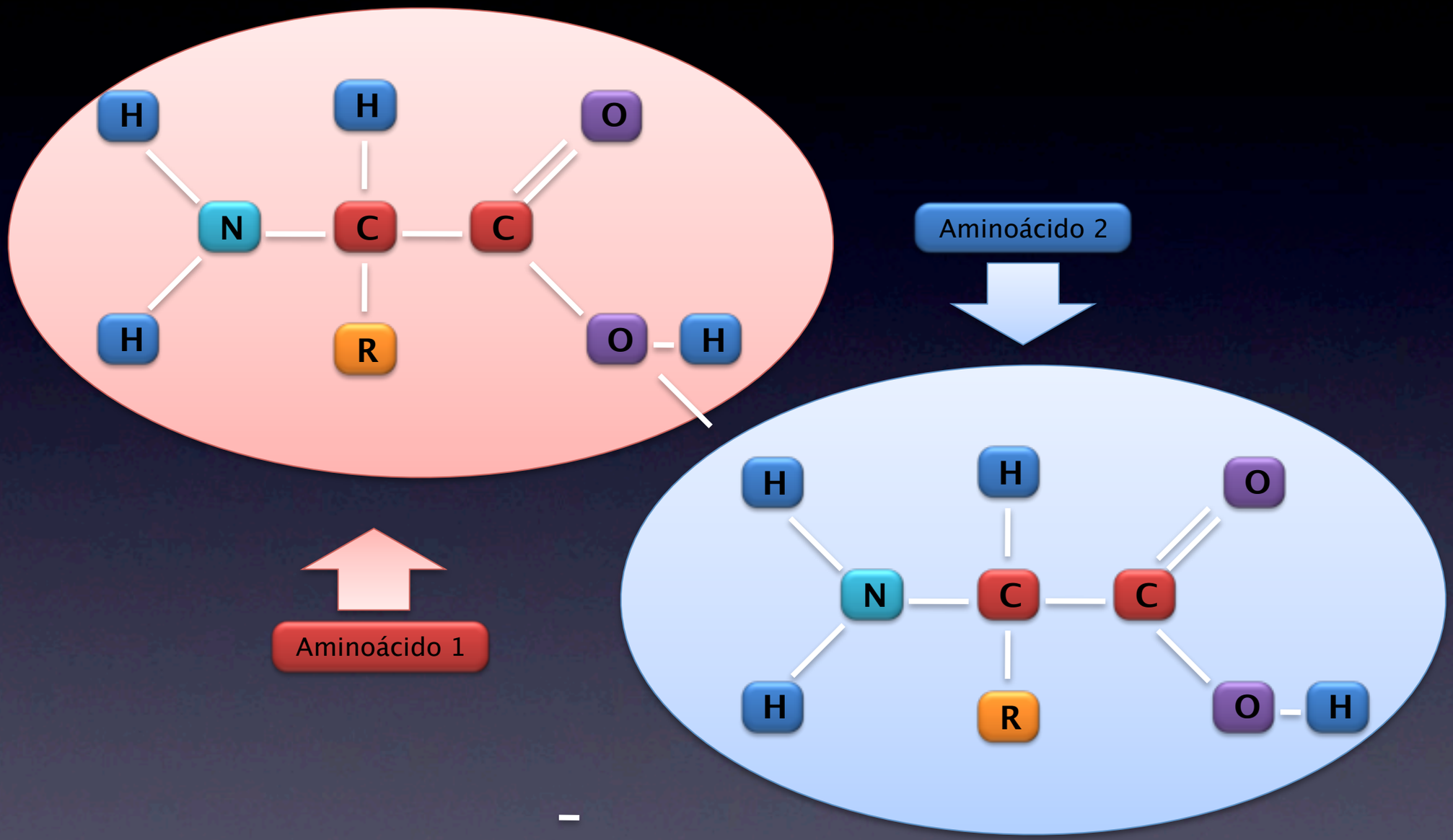


"Real" protein like

Estructura de un aminoácido



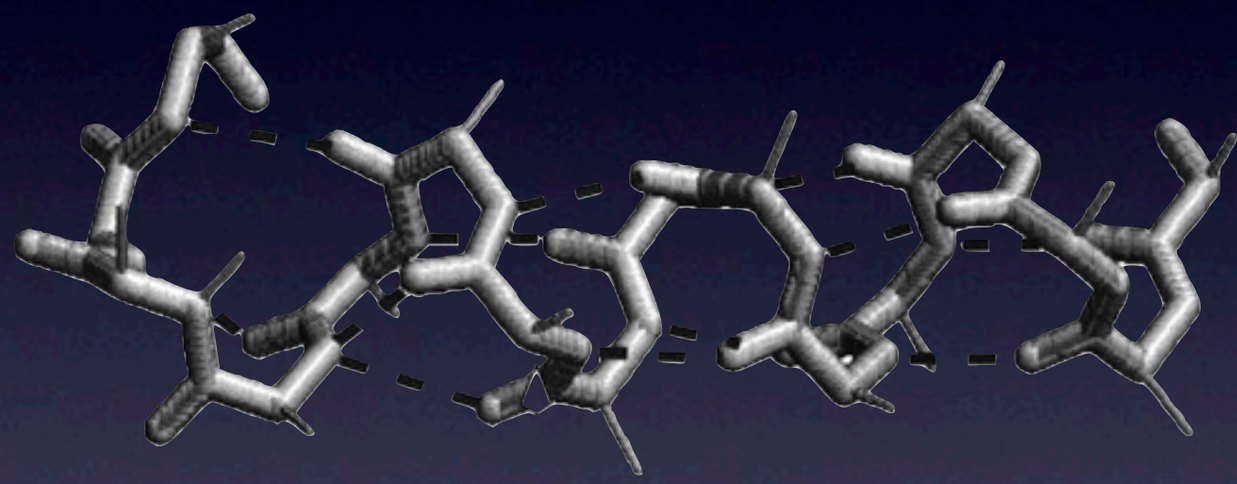
Concatenación de aminoácidos

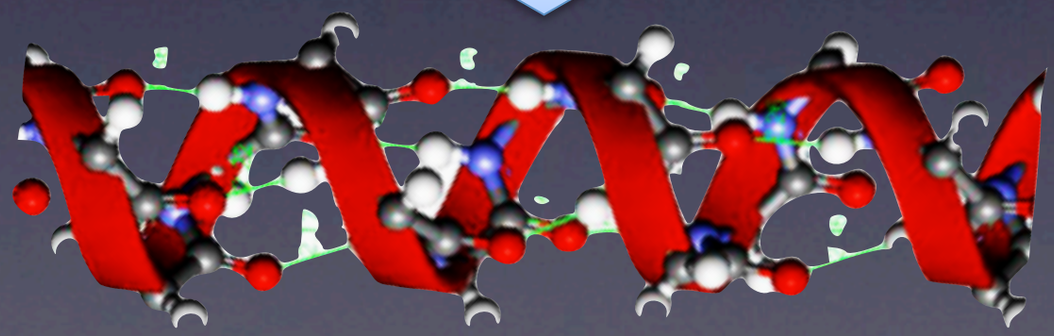


Modelos 3D de Proteínas

Hélice Alfa

- La hélice alfa se genera mediante el apilamiento de aminoácidos en una hélice, formando un cilindro.

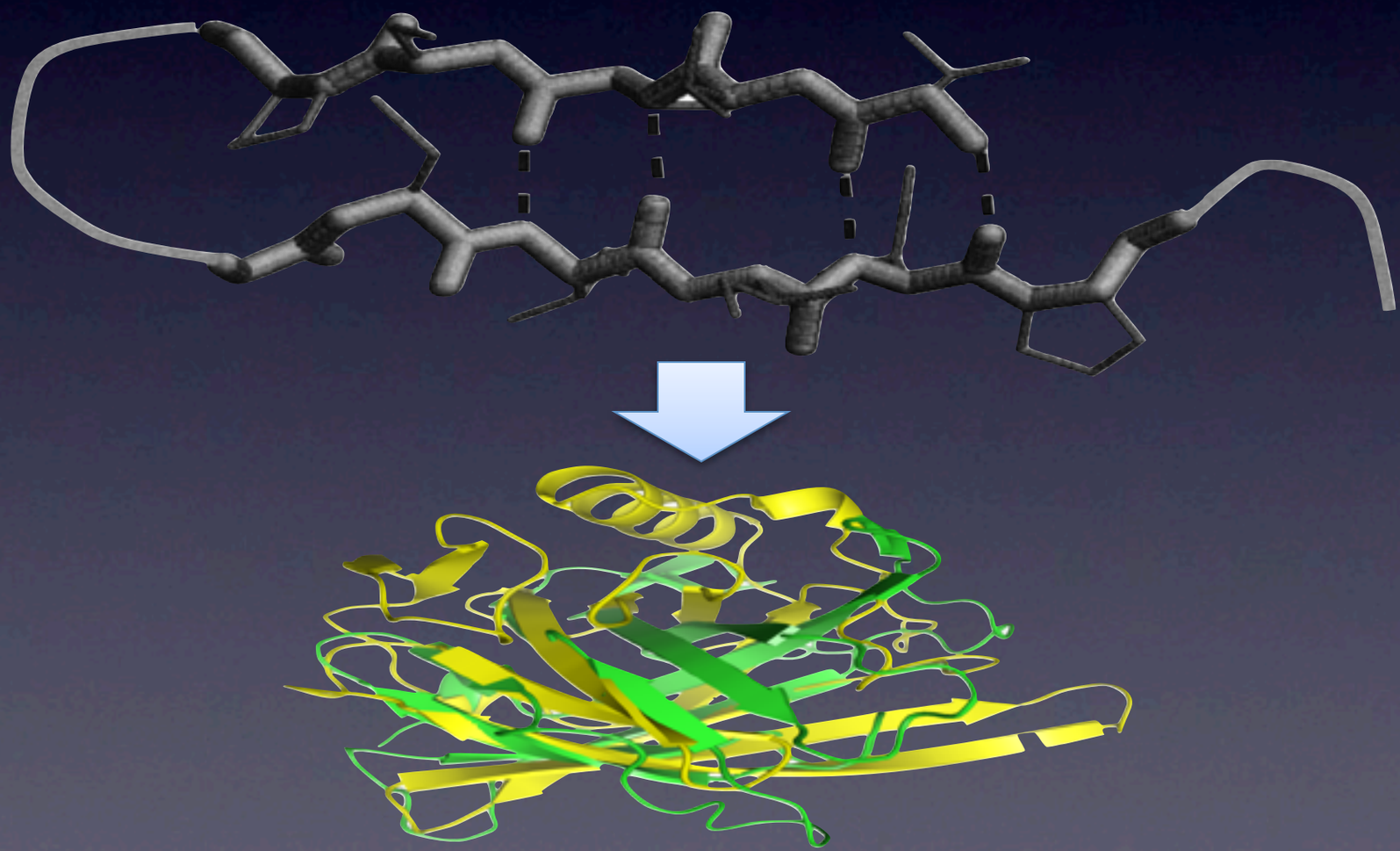




Modelos 3D de Proteínas

Hoja Beta

- La hoja beta es una estructura donde las partes amino de la cadena se apilan una sobre otra.



Modelos 3D de Proteínas

Tipos de Estructuras

- Estructura primaria: la secuencia de aminoácido de la proteína

- Estructura secundaria: se refiere a regiones que pueden ser hélices alfa, formas betas, etc.

- Estructura terciaria: comprende la estructura 3D de la proteína. Si la proteína está compuesta por sub-proteínas, entonces la estructura terciaria describe la estructura de las sub-proteínas.

- Estructura cuaternaria: Describe a toda la estructura tridimensional de la proteína.

Modelos 3D de Proteínas

Modelos de Predicción

- El problema consiste en predecir la estructura terciaria (la forma) a partir de una estructura primaria (secuencia aminoácidos).

- Tipos de aproximaciones de predicción:
  - Namely molecular dynamics.
  - Protein structure prediction.
  - Homologous modeling.

Modelos 3D de Proteínas

Aproximaciones

Namely Molecular Dynamics

Simulación

- Considera los campos de fuerza que actúan sobre los átomos de la cadena y el solvente.

Vectores

- De acuerdo a las fuerzas actuantes (**fuerzas hidrofóbicas**, enlaces covalentes, fuerzas electroestáticas, fuerzas de van der Waals, enlaces de hidrógeno, etc.)
- Son válidas generalmente por alrededor de 10^{-15} segundos.

Iteración

- Hasta que se encuentra una confrontación estable.

Tiempo

- Las proteínas tardan entre milisegundos y segundos en doblarse
- Debe realizarse una gran cantidad de cálculo para una gran cantidad de átomos una gran cantidad de veces.

IMPRACTICABLE

Modelos 3D de Proteínas

Aproximaciones

Protein Structure Prediction

Se considera un modelo arreglado de energía

- E: $\omega \rightarrow \mathbb{R}$, donde ω es el set de todas las conformaciones posibles.

La estructura nativa es la que posee la mínima energía

- ω t.q $E(\omega)$: mínimo

NP-Duro

- Se utilizan modelos simplificados (como **lattice model**)

Modelos 3D de Proteínas

Aproximaciones

Homologous Modeling

Se utilizan estadísticas para calcular pseudo-funciones de energía de acuerdo a la frecuencia de ciertos aminoácidos que se sabe que se encuentran cerca a otras conformaciones de muestras representativas en bases de datos de proteínas.

Se intenta de alinear en paralelo tanto la secuencia como la estructura, mediante la comparación de una nueva proteína P con una proteína conocida Q, asumiendo que P y Q están relacionadas mediante una secuencia de alineamiento.

Modelos 3D de Proteínas

Lattice Models

Experimentos con pequeñas proteínas han sugerido que el estado nativo de las proteínas corresponde a un estado de mínima energía libre.	Esto no ha sido demostrado aún.

Hipótesis ampliamente aceptada y es parte de la base de la predicción de conformaciones de proteínas.

Simplificaciones:

- Los monómeros se representan todos del mismo tamaño.
- Los largos de los enlaces son iguales.
- Las posiciones de los monómeros se restringen a posiciones en una grilla regular.

¡La simplificación más común es el modelo HP!

Modelos 3D de Proteínas

Proteínas HP

- Se realizan las siguiente simplificaciones:

- El alfabeto de las 20 letras de aminoácidos se reduce a un alfabeto de 2 letras (H y P)
- H representa a los aminoácidos hidrofóbicos.
- P representa a los aminoácidos polares o hidrofílicos.
- Esta reducción se puede realizar debido a que la fuerza hidrofóbica es la fuerza predominante en el proceso de doblado de las proteínas.
- Así, la función de energía se puede expresar mediante la siguiente tabla:

	H	P
H	-1	0
P	0	0

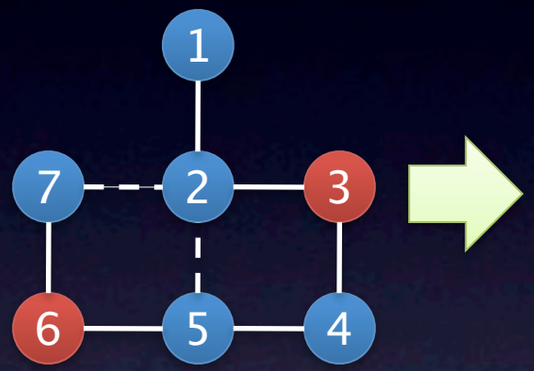
- Si los 2 monómeros son H, su contribución de contacto es −1, 0 en otro caso.

Modelos 3D de Proteínas

Proteínas HP

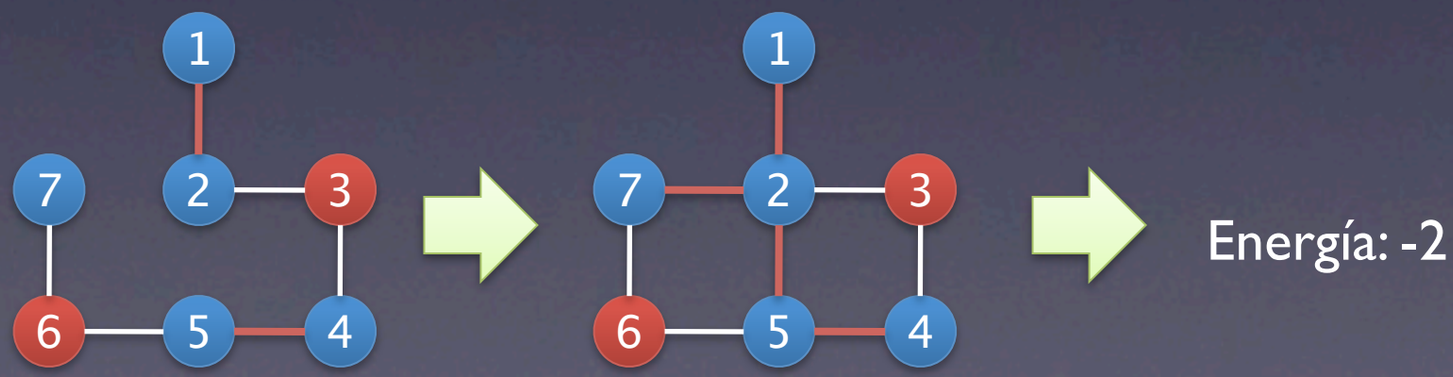
- Dada una confrontación, se distinguen vecinos «conectados» y «topológicos»

H H P H H H P H



Conectados	Topológicos
1 - 2	2- 5
2 - 3	7 - 2
3 - 4	
…	

- La energía del sistema corresponde al negativo de la cantidad de enlaces H ·H
- Entonces se busca la confrontación que contiene la máxima cantidad de estos enlaces:



Modelos 3D de Proteínas

Proteínas HP

Inconvenientes

- La principal desventaja es la degeneración: una secuencia HP dada puede tener distintas conformaciones teniendo un número máximo de enlaces H·H.
- Este es un modelo conceptualmente bastante simple.
- Permite la incorporación de refinamientos (HPNX para incluir fuerzas electroestáticas, etc)
- Computacionalmente hablando, es «menos intratable» que los modelos dinámicos totales.
- Este es un problema NP-Completo

Modelos 3D de Proteínas

Proteínas HP

Restricciones

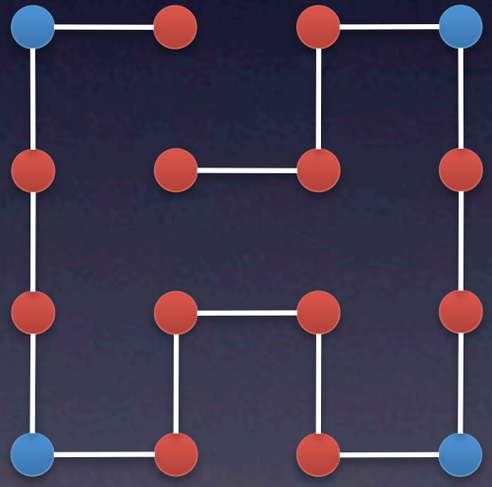
- La idea es establecer restricciones sobre estructuras locales.
- Este método garantiza alcanzar un óptimo global.
- Intenta encontrar la combinación hidrofóbica con una superficie mínima, mediante la introducción sistemática de restricciones geométricas, eliminando así ramas posibles del árbol de búsqueda.

Modelos 3D de Proteínas

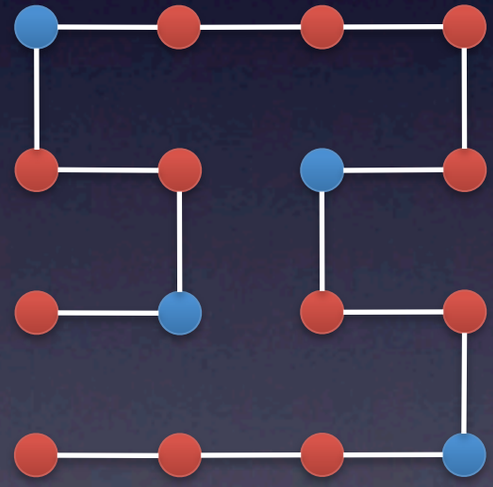
Proteínas HP - Ejemplo

H H H P H H P H H H H P H H P

Estructura nativa

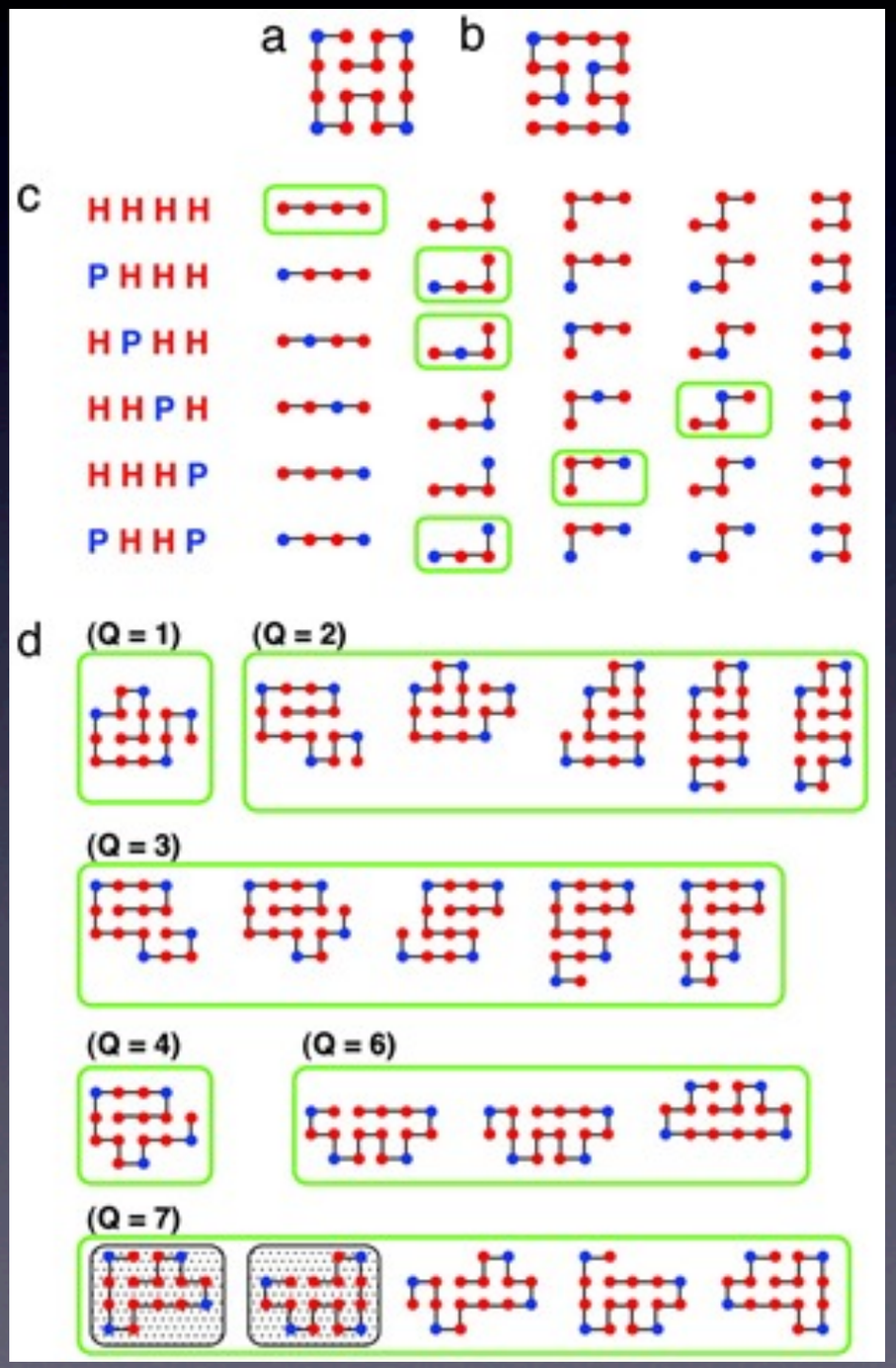


Calculada con HP

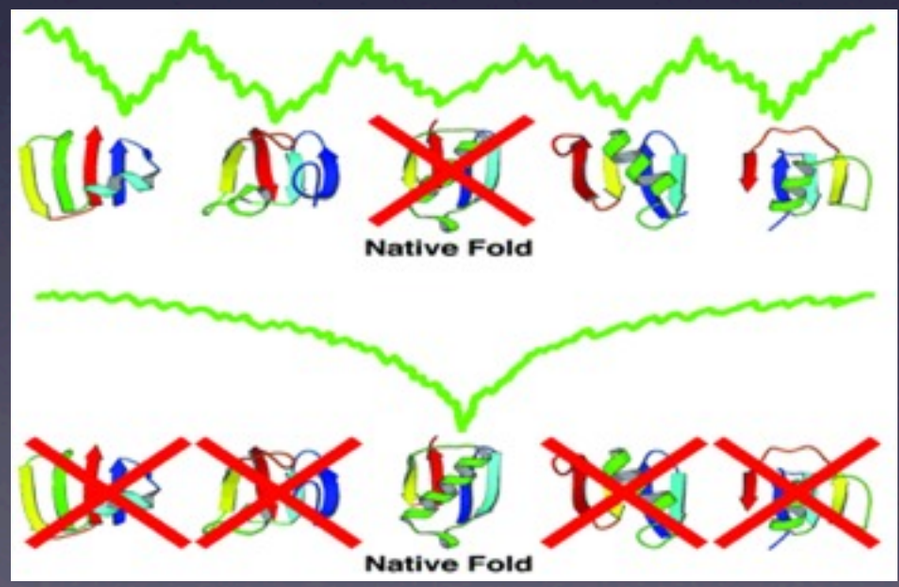


Modelos 3D de Proteínas

Proteínas HP - Ejemplo



a. Estructura nativa del modelo HP.
b. Ejemplo de la estructura que predice el modelo.
c. Los segmentos que se pueden dar en la secuencia y sus posibles estructuras. Las estructuras rodeadas de verde son prohibidas debido a las restricciones impuestas.
d. Las estructuras en las cuales la energía es mínima (E= -8). Solo las estructuras en plomo no están prohibidas por el modelo.



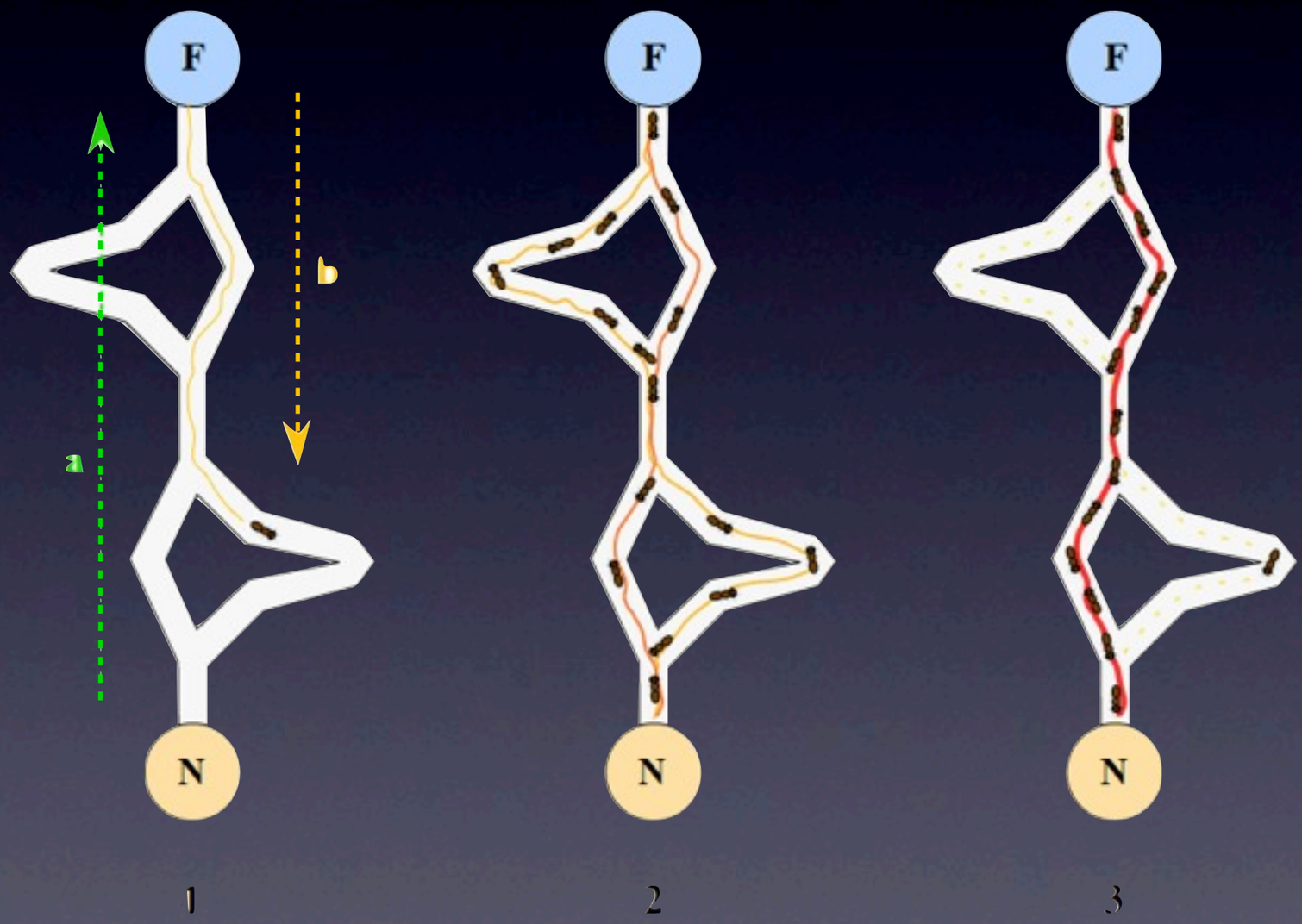
Modelos 3D de Proteínas

Algoritmo Colonia de Hormigas

- En la naturaleza, las hormigas vagan aleatoriamente en su búsqueda de alimento, y a lo largo de su trayectoria de regreso a la colonia depositan feromonas. Si otras hormigas encuentran este rastro, lo más probable es que sigan esta trayectoria para depositar el alimento en la colonia.
- Con el paso del tiempo el rastro de la feromona comienza a evaporarse y se reduce su fuerza atractiva. Las hormigas que siguen el rastro refuerzan la intensidad de la feromona, con lo que logran que perdure por más tiempo.
- Mientras más hormigas recorren una trayectoria, más intenso es el olor del rastro, lo que estimula a más hormigas a seguir esa trayectoria.
- Desde el punto de vista algorítmico, la evaporación de la feromona tiene la ventaja de evitar la convergencia a una solución localmente óptima. Si no hubiera evaporación, todas las trayectorias serían igualmente atractivas para las hormigas. Esta situación haría que la exploración del espacio de la solución fuese demasiado amplia.
- Así, cuando una hormiga encuentra una buena trayectoria de la colonia a una fuente de alimento, es más probable que otras hormigas sigan esa trayectoria, y la regeneración de la feromona provoca que finalmente todas las hormigas sigan una sola trayectoria.
- Este comportamiento es la base para el diseño del algoritmo, donde las "hormigas simuladas" caminan alrededor del gráfico que representa el problema a solucionar.

Modelos 3D de Proteínas

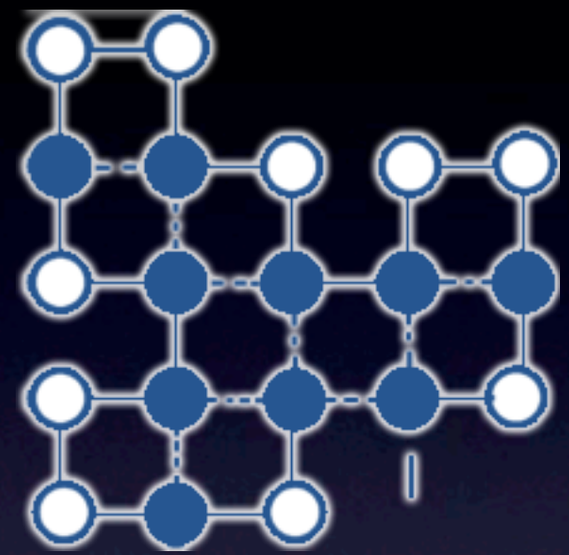
Algoritmo Colonia de Hormigas



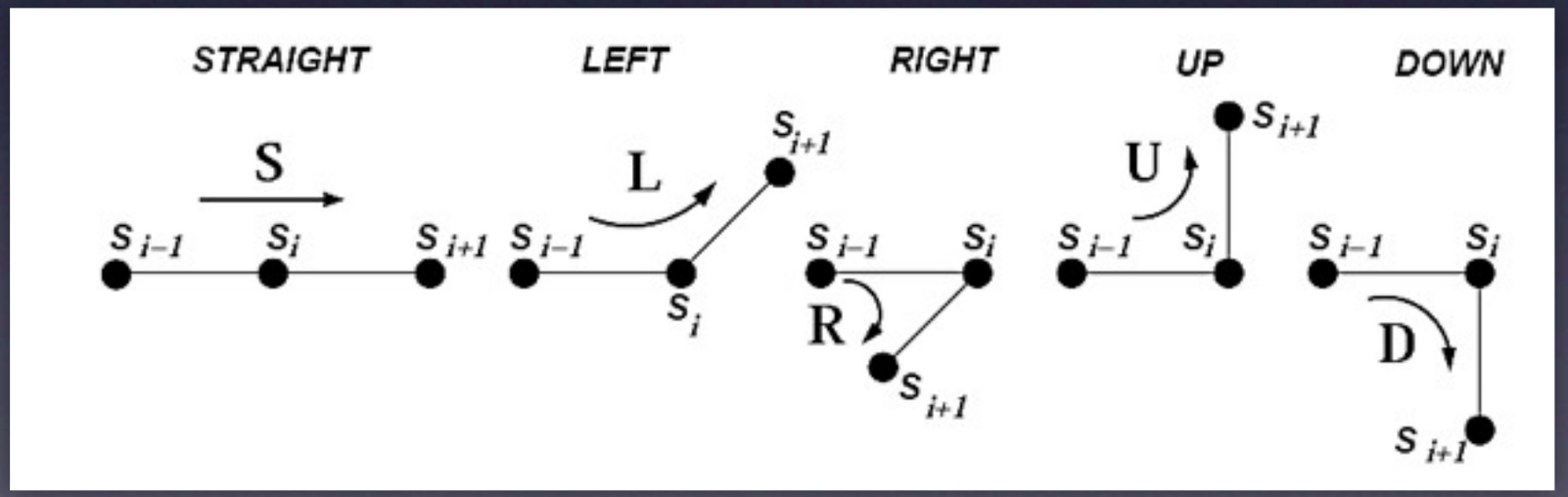
Modelos 3D de Proteínas

Algoritmo Colonia de Hormigas

```
procedure ACO
        initialise pheromone trails;
        while (termination condition not satisfied) do
                construct candidate conformations;
                perform local search;
                update pheromone values;
        end
end
```



L S L L R R L R L L S L R R L L S L

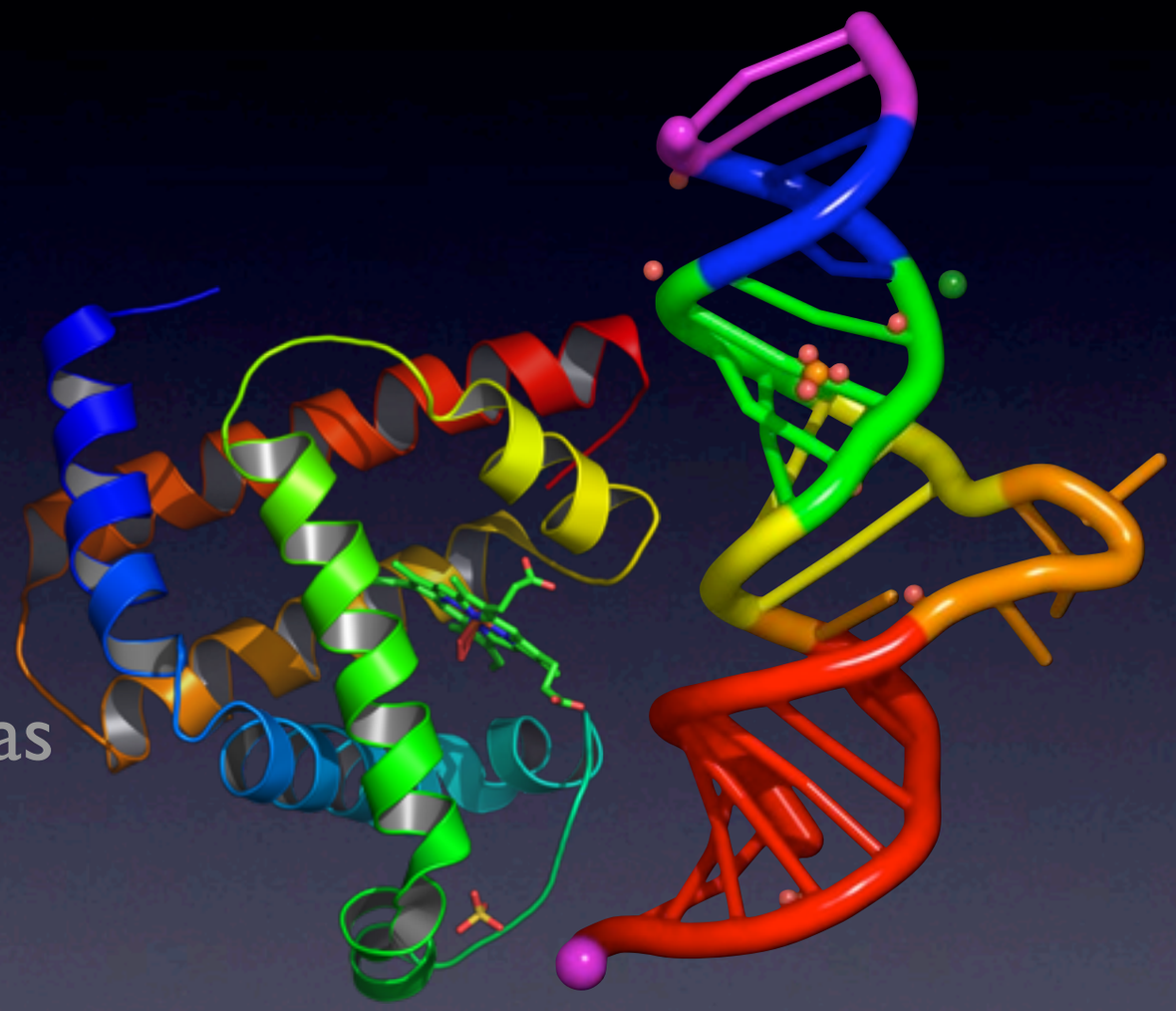


La Tarea

Pronto...

ARN y Proteínas

Estructura Dimensional
Modelos Simplificados de Proteínas



EMILIO HECK OLIVA - IGNACIO MELLA TÉLLEZ