Métodos Computacionales para la Comparación de Proteínas

David Pelta
http://decsai.ugr.es/~dpelta

Grupo de Trabajo en Modelos de Decisión y Optimización (MODO)
http://decsai.ugr.es/modo
Departamento de CC e IA
E.T.S. Ingeniería Informática
Universidad de Granada

Grupo de Trabajo en Modelos de Decisión y Optimización

Líneas de Investigación en Bioinformática (en colaboracíón con el Dr. N. Krasnogor)

- Comparación de estructuras de proteínas
- Diseño e implementación de un sistema gestor de modelos avanzado para SAD
- Problema de plegamiento de Proteínas

Momento Publicitario: el nro especial de Fuzzy Sets & Systems en Bioinformática está en prensa. Las versiones preliminares están disponibles on-line

Conceptos Básicos

- Las proteínas son cadenas de aminoacidos (20 diferentes)
- Esta secuencia es la *estructura primaria* (representada como un string de 20 simbolos diferentes)
- La secuencia primaria forma *estructuras secundarias*
- Las *estructuras secundarias* forman *estructuras* terciarias

La estructura terciaria es la disposición 3D de los residuos que determina la función biológica de la proteína.

Para qué sirven las proteínas:

Proteínas Estructurales: son los "ladrillos" del organismo.
Enzimas: Promueven/evitan reacciones reacciones químicas.
Proteínas Transmembrana: son los "guardianes" de la célula.

¿Por que es necesario comparar estructuras proteicas?

- Agrupar proteínas en terminos de similaridad estructural
- Determinar el impacto de residuos inviduales en la estructura de la proteina
- Identificar homologos distantes de familias de proteínas
- Predecir la función de proteínas con bajo grado de similaridad en secuencia con otras proteínas
- "Fabricar" nuevas proteínas que realicen funciones especificas
- Verificar predicciones ab-initio

Proceso de Comparación

El biologo debe decidir **que** es lo que se compara
(es decir, el significado de *similaridad*)

Heuristico, dependiente del dominio

Construir un modelo de similaridad

a través de

Una Medida

Una Metrica

Exactos
Aproximados
Heuristicos

Métodos

Enfoques Existentes

Implementados como programas /servidores:

SSAP (Orengo & Taylor, 96), ProSup (Feng & Sippl, 96), DALI (Holm & Sander, 93), CE (Shindyalov & Bourne, 98), LGA (Zemla, 2003), SCOP (Murzin, Brenner, Hubbard & Chothia, 95), CATH (Orengo, Mithie, Jones, Jones, Swindells & Thornton, 97)

Estos métodos estan basados en:

Programacion Dinamica (Taylor, 99), Comparación de matrices de distancia (Holm & Sander, 93,96), Detección del subgrafo común maximal (Artimiuk, Poirrette, Rice & Willet, 95), Matching geometrico (Wu, Schmidler, Hastie & Brutlag, 98), enfoques basados en RMSD (Maiorov & Crippen, 94 – Cohen & Sternberg,80)

Un resumen de medidas de similaridad (hasta 37) aparecen en (May, 99)

Debe notarse que:

- No existe consenso sobre cual de los métodos es el mejor.
- Existen dificultades asociadas a cada uno de los metodos.
- Todos asumen que se puede definir una función adecuada de puntuacion (score) tal que los valores optimos se correspondan con el mejor matching estructural de ambas estructuras.
- Esquemas basados en RMSD (*root mean square deviation*) pueden tener problemas de inestabilidad numérica.
- Algunos métodos no pueden producir un ranking adecuado debido a:
  - definiciones ambiguas de la medida de similaridad
  - niegan la existencia de soluciones alternativas con valores de similaridad equivalentes

Similaridad de Estructuras de Proteínas mediante *Universal Similarity Metric*

(Krasnogor & Pelta, 2004 in *Bioinformatics*)

USM aproxima cada posible medida de similaridad. No se necesita decidir a *a priory* cual es el modelo biológico subyacente (el *QUE*)

USM se presento en (Li, Badger, Chen, Kwon, Kearney & Zhang, 2001) y se refino en (Li, Chen, Li, Ma & Vitanyi, 2003)

En el centro de USM está el concepto de *Complejidad de Kolmogorov*

La complejidad de Kolmogorov K(.) de un objeto ***o*** se define como la longitud del programa más corto para que una maquina de Turing universal U produzca el objeto ***o*** como salida:

K(o) = min {|P|, P es un programa y U(P)= o} (1)

Una medida relacionada es la complejidad condicional de Kolmogorov de *o_1* dado *o_2*:

K(*o_1*|*o_2*) = min{|P|,P un programa U(P,*o_2*) = *o_1* } (2)

mide cuanta información se necesita para producir el objeto 1 dado que conocemos el objeto 2.

Se puede demostrar que la Distancia en Informacion entre dos objetos es equivalente (considerando un termino aditivo logaritmico) a:

ID(o_1,o_2) = max { K(o_1|o_2), K(o_2|o_1)} (3)

La *Universal Similarity Measure*, es una metrica en el sentido matematico, es normalizada y universal.

Formalmente, se define como

$$d(o\_1,o\_2) = \frac{\max \{ K(o\_1|o\_2^*), K(o\_2|o\_1^*)\}}{\max\{ K(o\_1),K(o\_2)\}} \quad (4)$$

donde ***o_1* ,o_2**** indican los programas más cortos que computan ***o_1 , o_2*** respectivamente.

A partir de la Eq. (4) podemos calcular una matriz con la distancia USM considerando ***estructuras de proteínas*** *como los* ***objetos o_i*** *y* ***o_j*** *para todos los i,j de un conjunto.*

Pero cómo se calcula d(.,.)?

Desafortunadamente, la universalidad de USM se "paga" con la no computabilidad: la complejidad de Kolmogorov es no-computable, solo "upper-semi computable".

Como aproximar d(.,.) calculando aproximadamente K(.)??

Cada proteina se codifica como un string s y el valor K(s) se aproxima con el tamaño (nro de bytes) del string comprimido zip(s)

Es decir: $K(s) \sim |zip(s)|$ (5)

En (Li & Vitanyi, 97) se muestra que la informacion algoritmica es simetrica, por lo tanto podemos aproximar K(o_1|o_2) como K(o_1 + o_2) - K(o_2) donde + es la concatenacion de strings y K(.) se estima como antes.

¿Como transformamos una proteína a un string?

Para calcular la USM, solo usaremos una parte de la información de una proteína (extraída de PDB), a partir de la cual construiremos un mapa de contactos:

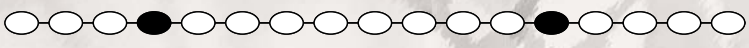
Un MC es una representación compacta de la estructura 3D de una proteina. Se especifica con una matriz binaria S, donde los índices se corresponden con residuos de la proteina

$$S_{i,j} = \begin{cases} 1 \text{ } si\ el\ residuo\ i\ y\ el\ j\ estan\ en\ contacto \\ 0 \text{ } caso\ contrario \end{cases}$$

Dos residuos i y j se dice que estan *en contacto* si su distancia en la estructura 3D es menor a *R* Angstroms.

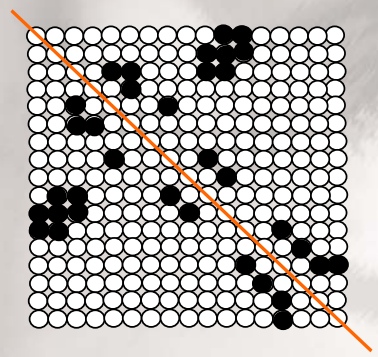
El valor *R* se denomina el *threshold (umbral)* del mapa de contacto.

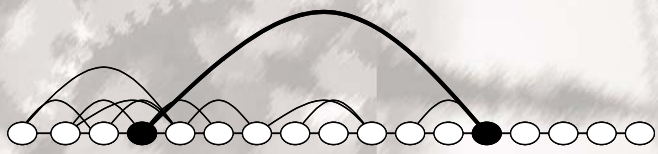
Una proteina:

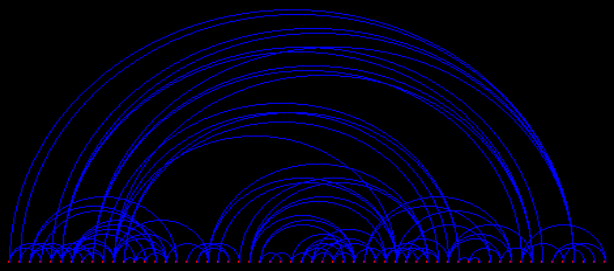


Su estructura:

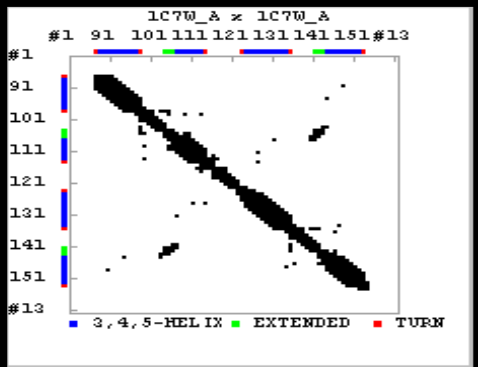
El mapa de contacto asociado:







1C7W.PDB



Ejemplo de Aplicación

• Conjunto de datos usado en (Chew & Kedem, 2002) para evaluar un método destinado a medir "formas de consenso".

• Esta formado por 36 proteínas de tamaño medio, correspondientes a 5 familias diferentes

globins: *1eca, 5mbn, 1hlb, 1hlm, 1babA, 1babB, 1ithA, 1mba, 2hbg, 2lhb, 3sdhA, 1ash, 1flp, 1myt, 1lh2, 2vhbA, 2vhb*

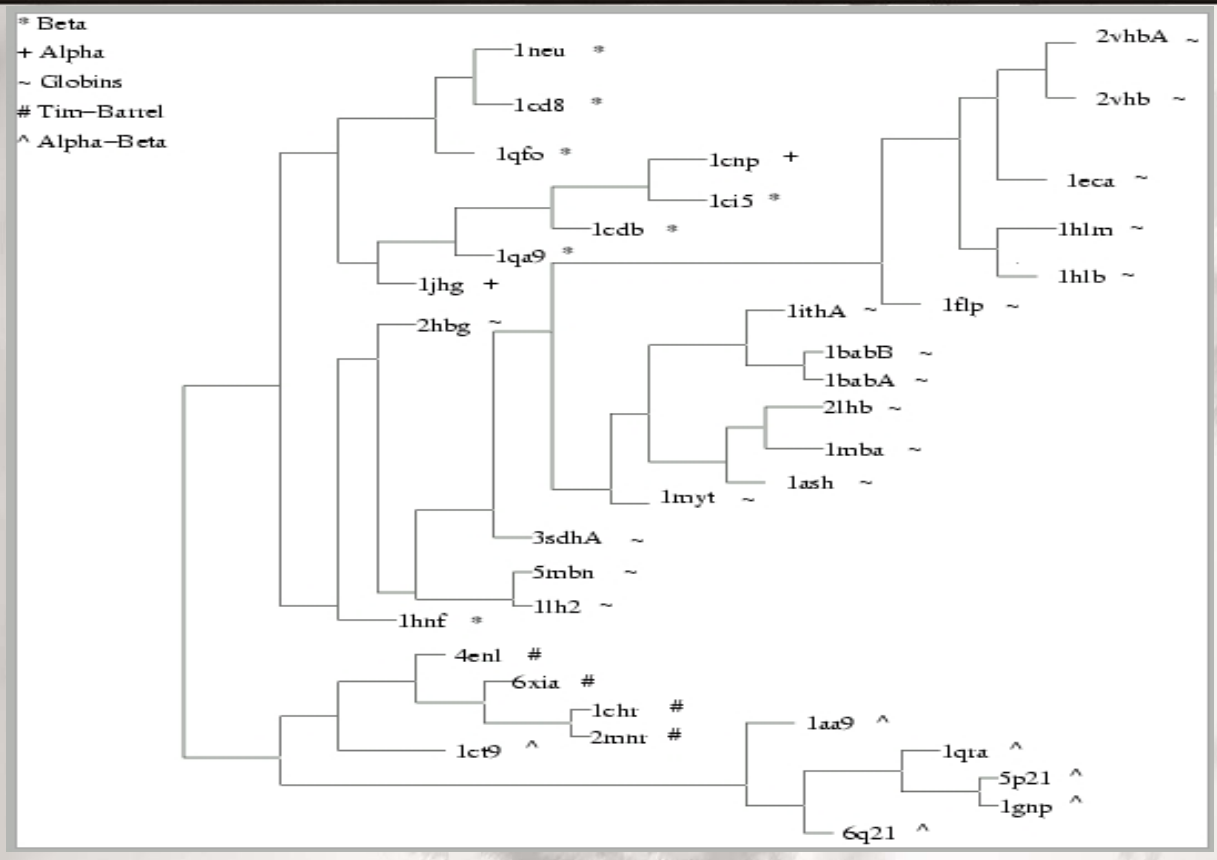
alpha-beta: *1aa9, 1gnp, 6q21, 1ct9, 1qra, 5p21*

tim-barrels: *6xia, 2mnr, 1chr, 4enl*

all beta: *1cd8, 1ci5, 1qa9, 1cdb, 1neu, 1qfo, 1hnf*

all alpha: *1cnp,1jhg*

• La proteina 2vhb aparece dos veces (como 2vhb and 2vhbA) para chequear si USM es capaz de detectarlo e induce un cluster donde ambas aparecen juntas



Salvo casos puntuales, el agrupamiento es casi perfecto

Por lo tanto, USM nos permite medir la similaridad de estructuras de proteínas sin responder a la pregunta “QUE?”

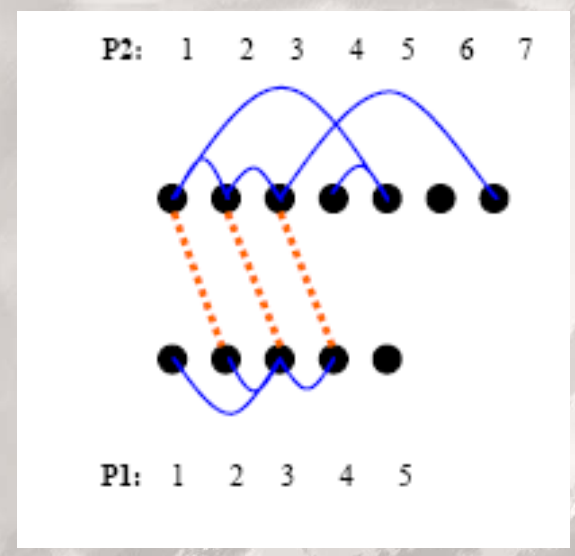
Pero…

no nos dice como o donde dichas estructuras son (di)similares

Para eso resolvemos el problema de buscar la
Máxima Superposición de Mapas de Contacto

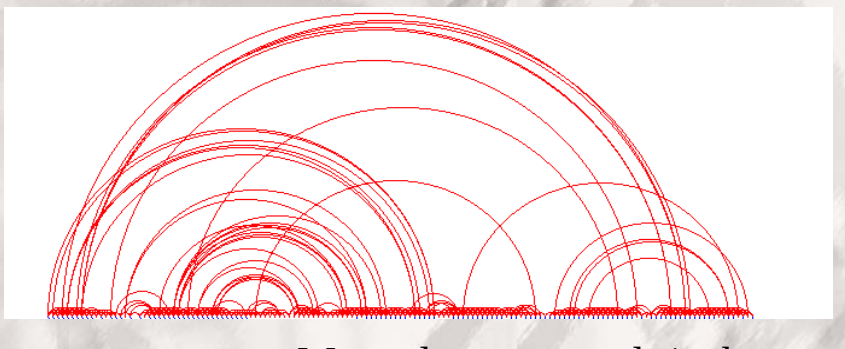
Métodos para el Calculo de la Máxima Superposición de Mapas de Contacto (MAX-CMO)

- La similaridad de dos proteínas se puede calcular a través del "alineamiento" de sus mapas de contacto.
- Un alineamiento de dos proteínas es un emparejamiento entre los aminoacidos correspondientes
- Los Mapas de Contacto se pueden ver como grafos donde los nodos son residuos y los arcos indican que estan en contacto.



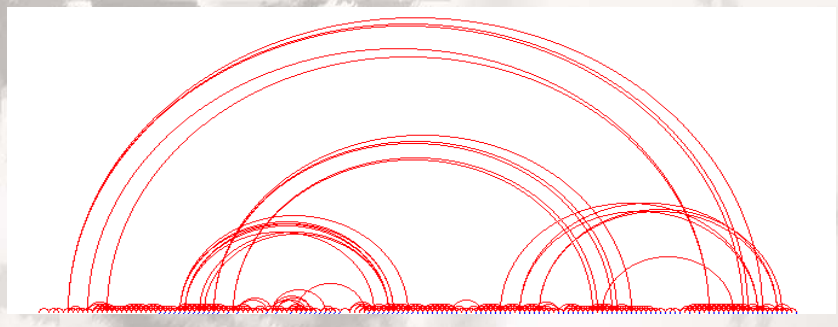
- Las lineas rojas indican la correspondencia entre los residuos.
- MAX_CMO consiste en buscar la correspondencia que maximice la cantidad de ciclos de longitud 4.
- ***Este problema es NP-Completo***

1ash



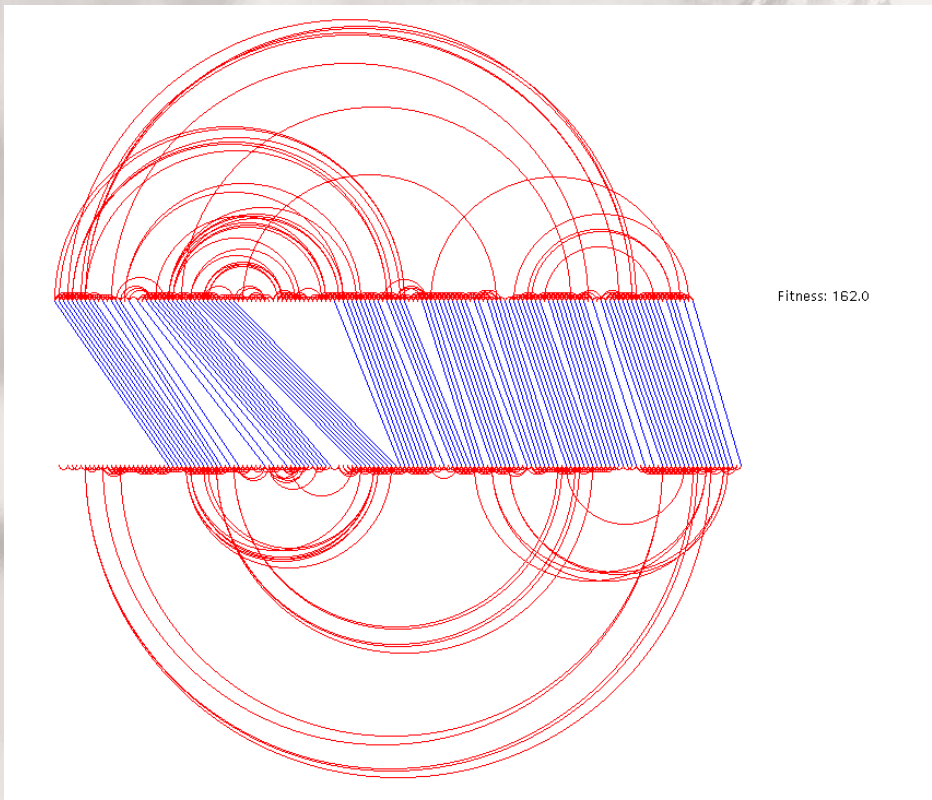
Mapa de contacto de1ash

1hlm



Mapa de contacto de1hlm

Un alineamiento posible entre ambos mapas:



- La Máxima Superposición de Mapas de Contacto se puede formular como un problema de programación entera (Caprara & Lancia, 2002)
- Características especiales de los mapas de contacto construidos a partir de estructuras de proteínas, derivan en instancias del problema que permiten la aplicación de relajación lagrangeana para resolverlo.
- MAX-CMO también se ha abordado mediante heurísticas, como algoritmos memeticos (MA).
- LR permite obtener las mejores soluciones conocidas del problema y en muchos casos, las optimas.
- MA's permiten obtener soluciones sub-optimas, pero muchas de ellas. Por tanto, el usuario final puede elegir aquella con mas significado biológico.

Sin Embargo...

El modelo de mapas de contacto es un enfoque que maximiza una relación puramente geométrica entre dos proteínas sin tener en cuenta la cuestión biológica.

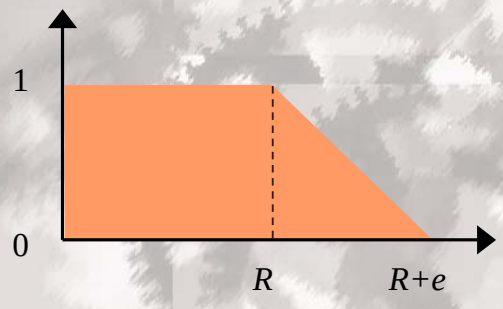
No permite modelizar, por ejemplo, errores producidos en las determinación de las coordenadas cartesianas de los átomos en los procedimientos experimentales (NMR, cristalografía de rayos X).

Los mapas de contacto se basan en un único umbral, perdiendo por lo tanto información de contactos que se producen a umbrales alternativos en forma simultanea.

Propuesta: Fuzzy Contact Maps

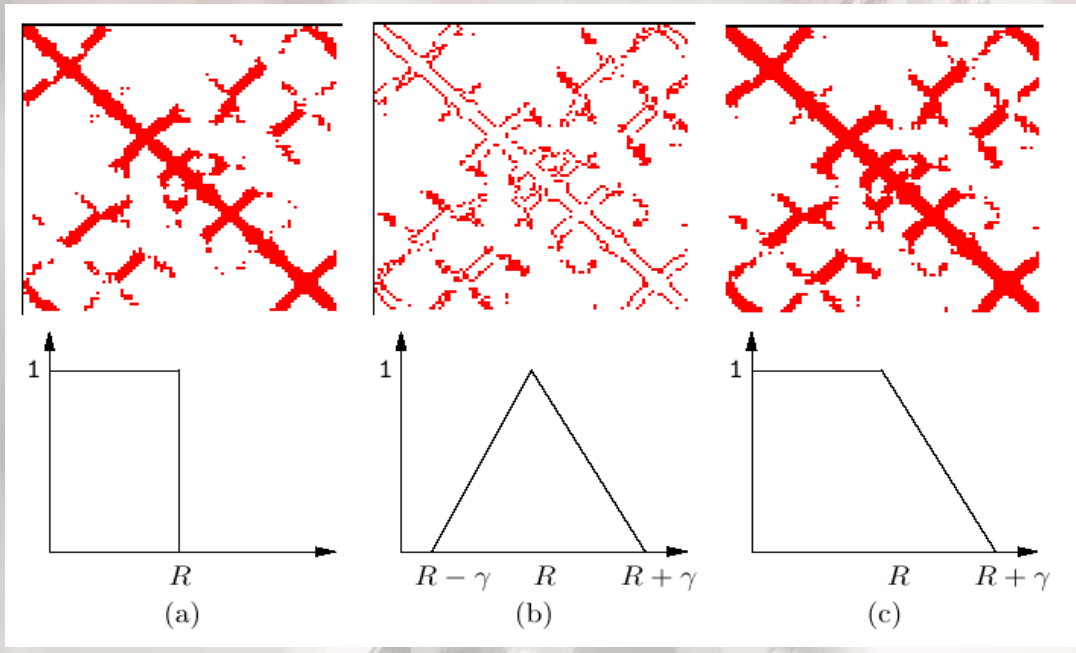
Proponemos una versión "difusa" de los mapas de contacto teniendo como base la noción de "umbral difuso".

Usaremos conjuntos difusos para modelizar los contactos que se producen a una distancia "*aproximadamente R*" con la siguiente función de pertenencia $\mu(x)$



Ahora, cada entrada en la matriz de contactos es un valor en [0,1], que indica la "fuerza" del contacto.

Ejemplos



Diferentes definiciones de "aproximadamente R".

Cada punto, denota un contacto entre dos átomos cuya distancia *d* verifica que $\mu(d) > 0$

Fuzzy Contact Maps Generalizados

1 umbral $\Rightarrow$ *n umbrales:* para detectar patrones que surgen simultáneamente a diferentes escalas

1 función de pertenencia $\Rightarrow$ *m funciones de pertenencia:* puede ser necesario disponer de varias definiciones simultaneas de contacto

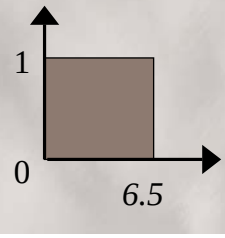
El nivel de un contacto entre elementos i,j se define como:

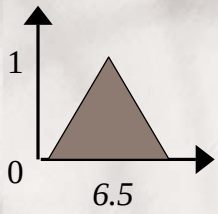
$$F_{i,j} = max\{\mu_1(\overline{[i,j]}, \Re_1), \mu_2(\overline{[i,j]}, \Re_2), \ldots, \mu_m(\overline{[i,j]}, \Re_n)\}$$

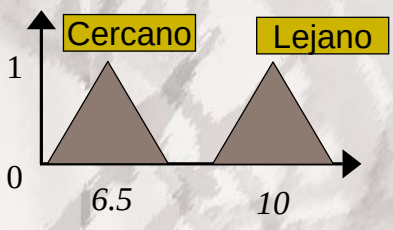
El mapa de contactos es (siendo *r* la cantidad de elementos)

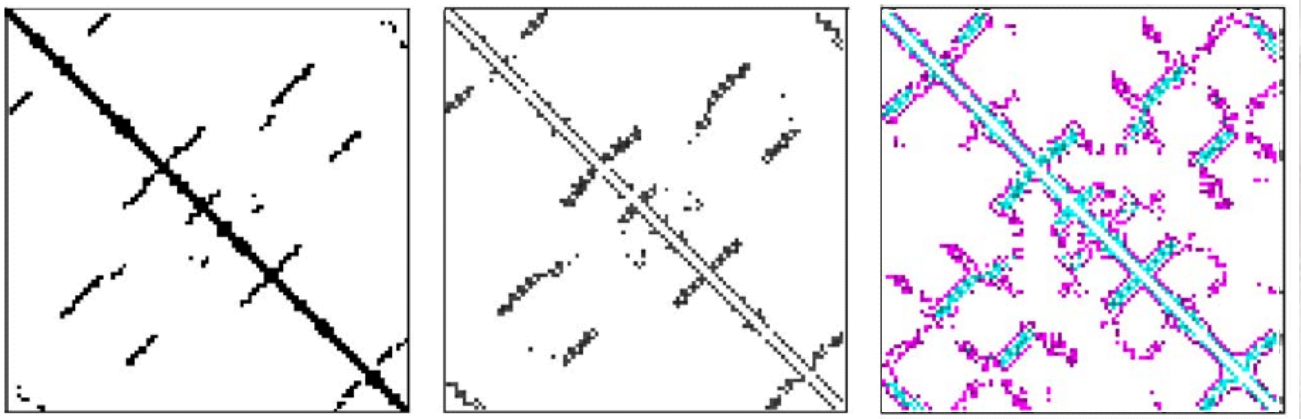
$$C^{r \times r} = (F_{i,j}) \; with \; 0 \leq i,j \leq r$$

Ejemplo: *2 umbrales, 2 funciones. de pertenencia*







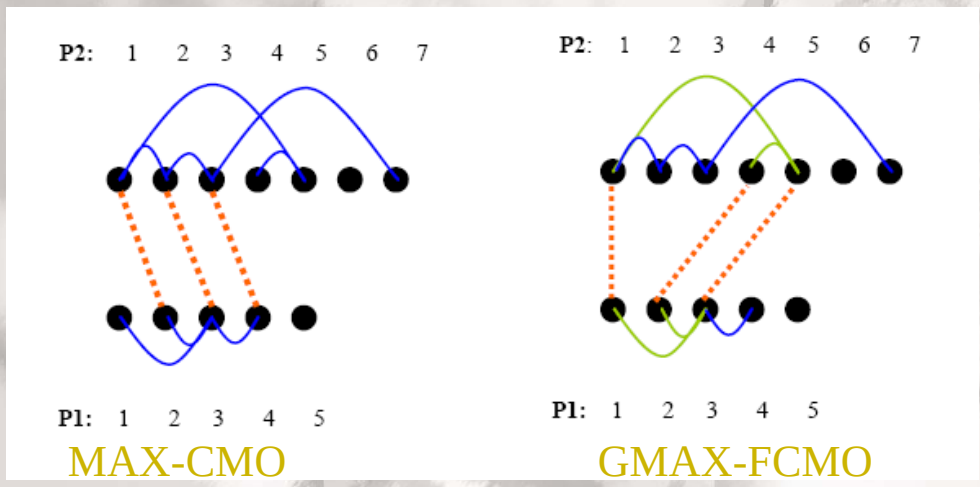


zoom

A medida que generalizamos, agregamos poder expresivo al modelo. En el ultimo, además tenemos información del tipo (y no solo del nivel), del contacto.

Máxima Superposición de Mapas de Contacto Difusos (GMAX-FCMO)

Comparamos proteínas vía la superposición de sus mapas de contacto difusos (con 2 umbrales y 2 funciones de pertenencia)



En GMAX-CMO queremos maximizar el nro de ciclos, preservando el significado semántico de los contactos (alineando "largos" con "largos" y "cortos" con "cortos")

Formalmente:

- dadas 2 proteínas P_1, P_2 con r_1, r_2 residuos cada una $(r_1 \leq r_2)$
- los respectivos mapas de contacto C^1, C^2 (con 2 umbrales y 2 funciones de pertenencia) con elementos $C^1_{i,j}$, $C^2_{i,j}$
- Matrices indicando el "tipo" de contacto (largo o corto) T^1, T^2 con elementos $T^1_{i,j}$, $T^2_{i,j}$

Definimos una superposición como un mapping $\sigma : r_1 \rightarrow r_2$, tal que si $i < j$, entonces $\sigma(i) < \sigma(j)$

Un ciclo s es una cuádrupla $(i, j, \sigma(i), \sigma(j))$ y su contribución al overlap se define como:

$$P(s) = (C^1_{i,j} \times C^2_{\sigma(i),\sigma(j)}) \quad \times \quad (T^1_{i,j} \otimes T^2_{\sigma(i),\sigma(j)})$$

Valores de pertenencia de los contactos

Tipo de los contactos 1 sii son iguales

En GMAX-FCMO buscamos un mapping s que nos maximice la suma de las contribuciones de los ciclos.

Obtenido el valor de superposición $opt(P_1, P_2)$, definimos la similaridad entre ambas proteínas, como:

$$sim(P_1, P_2) = \frac{opt(P_1, P_2)}{MAX\{opt(P_1, P_1), opt(P_2, P_2)\}}$$

El término $opt(P_i, P_i)$ nos da una idea de "autosimilaridad"

MAX-CMO es un caso particular de GMAX-FCMO, por lo tanto, estamos ante un problema NP-Completo.

No sabemos si los métodos exactos aplicables a MAX-CMO se pueden aplicar/adaptar al nuevo modelo.
Por lo tanto $\Rightarrow$ heurísticas

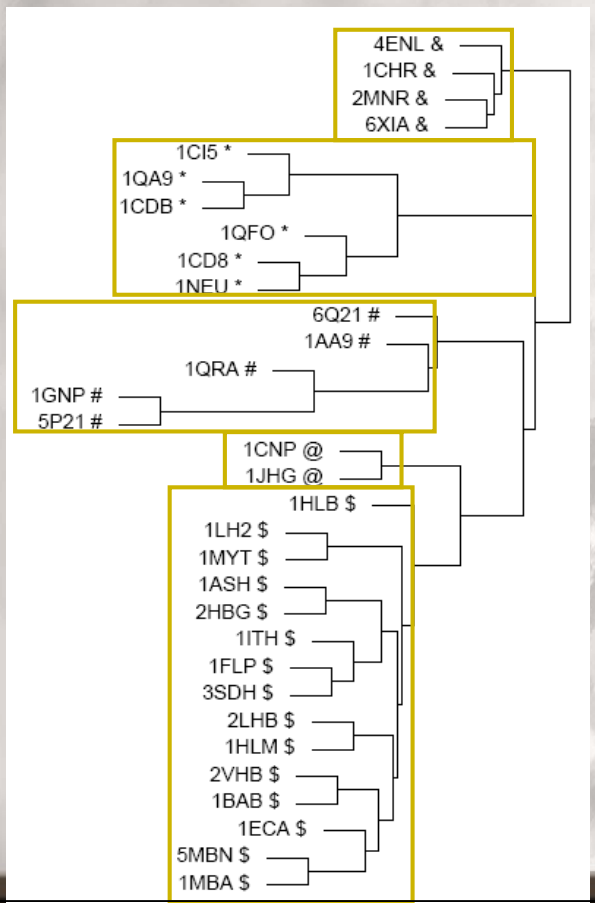
Fuzzy Adaptive Neighborhood Search (FANS)

- Es un método de búsqueda local
- Se basa en considerar el vecindario de una solución como un conjunto difuso de soluciones aceptables
- Se producen trayectorias entre soluciones que satisfagan cierto nivel de aceptabilidad
- Cuando se produce estancamiento, se cambia el operador de movimiento
- Esquema similar a *variable neighborhood descent search*
- Eficacia probada en otros problemas de optimización

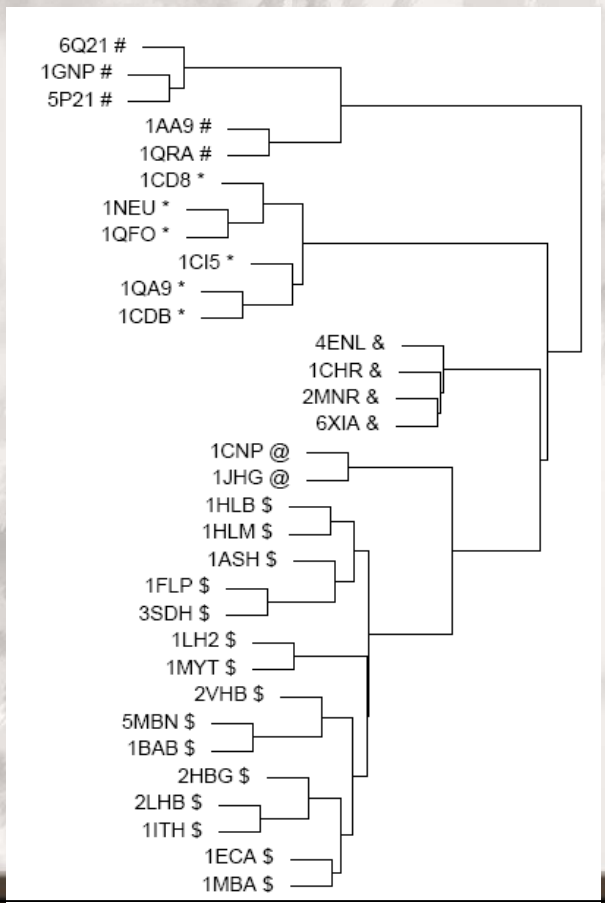
Ejemplo de Aplicación (I)

1. El dataset de Chew-Kedem, formada por 32 proteínas de 5 familias diferentes.
2. Para cada proteína, generar el fuzzy contact map (2 umbrales - 2 funciones de pertenencia)
3. Para cada par de FCM c_1,c_2, calcular el valor de similaridad correspondiente.
4. El valor de overlap $opt(c_1,c_2)$ se calcula usando FANS. 3 runs del algoritmo, dando lugar a 3 valores diferentes de overlap.
5. Usando el máximo / mínimo nivel de overlap, obtuvimos 2 matrices de similaridad entre todos los pares del dataset
6. Aplicamos clustering para ver si los grupos inducidos reproducen el agrupamiento original

Clustering con Mínimo



Clustering con Máximo

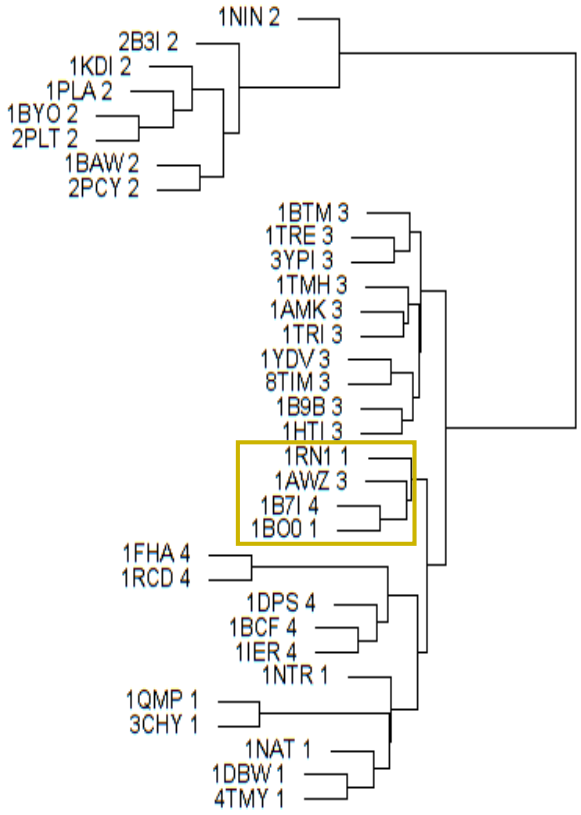


Ejemplo de Aplicación (II)

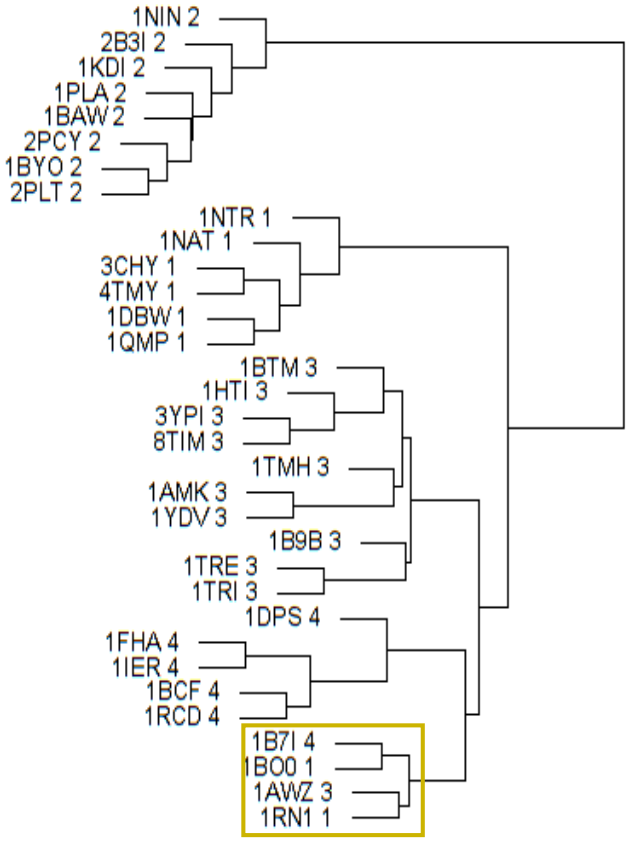
Siguiendo el mismo protocolo, utilizamos un nuevo dataset sugerido por J. Skolnick compuesto por 33 proteínas agrupadas en 4 clases:

- **Flavodoxin-like CheY-related** : *1ntr, 1nat, 1qmp, 1rn1, 3chy, 4tmy, 1bo0, 1dbw,*
- **Plastocyanin**: *1byo, 1baw, 1kdi, 1nin, 1pla, 2b3i, 2pcy, 2plt,*
- **TIM-Barrel**: *3ypi, 8tim, 1tmh, 1tre, 1tri, 1ydv, 1hti, 1amk, 1awz, 1b9b, 1btm*
- **Ferratin like**: *1bcf, 1b7i, 1dps, 1fha, 1rcd, 1ier.*

Clustering con Mínimo



Clustering con Máximo



Por lo tanto...

- Disponemos de una generalización difusa de los mapas de contacto, que da lugar a un modelo mas expresivo y flexible.
- Formulamos un nuevo problema de optimización: el problema generalizado de superposición de mapas de contacto difuso y su utilizacion como herramienta para el calculo de similaridad.
- Aplicando una heuristica simple en la optimización, conseguimos matrices de similaridad en un conjunto de prueba que permitieron recuperar los agrupamientos originales. (tambien verificado con datasets adicionales)

Un SAD para la Comparación de Estructuras de Proteínas

REQUISITOS

- Que incorpore métodos existentes, sin reimplementarlos en un lenguaje específico (*los investigadores proveen sus algoritmos, y no están dispuestos a modificarlos para adaptarlos a un contexto específico*).
- Que permita agregar nuevos métodos de forma dinámica (generación automática de interfaces).
- Que no haya que "recompilar".
- Una verdadera "base de datos" de algoritmos (un gestor de modelos en el contexto de SAD).
- Basado en herramientas del software libre.

ES POSIBLE !!!

En síntesis

Hemos resaltado la importancia del problema de comparar estructuras de proteínas y presentado nuestras líneas de trabajo:

1. *comparación basada en la USM*
2. *comparación con MAX-CMO*
3. *generalización en GMAX-FCMO*
4. *Construcción de un SAD para el proceso de comparación, que incorpore un sistema gestor de modelos avanzado*

Respecto a USM

Dimos evidencia matemática y experimental que USM se puede usar para medir la (di)similaridad estructural entre proteínas

- USM parece capturar otras medidas (mas "heuristicas") de similaridad

- USM necesita complementarse con otros algoritmos que indiquen explicitamente donde se dan las similaridades

- Para eso: MAX-CMO, GMAX-FCMO

Respecto a MAX-CMO

- Un modelo con formulacion exacta
- Disponemos de al menos dos metodos:
    - Relajacion Lagrangiana (LR)
    - Heuristicas (algoritmos memeticos, FANS)
- LR da los mejores resultados para MAX-CMO
- Los MA's (y otros) permiten obtener una familia de soluciones, dando la posibilidad al usuario de elegir aquella con mas sentido o relevancia biologica (y no matematica)
- Dado este rango de posibilidades "puramente matematico - puramente biologico", la búsqueda de una solucion optima no parece ser indispensable.

Respecto a GMAX-FCMO

- Propusimos una generalización difusa de los mapas de contacto, obteniendo un modelo mas expresivo y flexible.
- Formulamos un nuevo problema de optimización: el problema generalizado de superposición de mapas de contacto difuso y su utilizacion como herramienta para el calculo de similaridad.
- Aplicando una heuristica simple en la optimización, conseguimos matrices de similaridad en un conjunto de prueba que permitieron recuperar los agrupamientos originales. (tambien verificado con datasets adicionales)

Bibliografía

- *Measuring the similarity of protein structures by means of the universal similarity metric*, N. Krasnogor, D. Pelta, Bioinformatics (7) (2004) 1015–1021.
- *A fuzzy sets based generalization of contact maps for the overlap of protein structures*, D. Pelta, N. Krasnogor, C. Bousono-Calzon, J. Verdegay, J. Hirst, E Burke, Fuzzy Sets and Systems, Special Issue in Bioinformatics. in press (2005)
- *Multimeme Algorithms using Fuzzy Logic based Memes*, D. Pelta, N. Krasnogor, In Recent Advances in Memetic Algorithms and Related Search Technologies.W. Hart, N. Krasnogor, J. Smith (Eds). (in Press, 2004)
- *Fuzzy adaptive neighborhood search: examples of application*, D. Pelta, A. Blanco, J.L. Verdegay. Fuzzy Sets Based Heuristics for Optimization, Studies in Fuzziness and Soft Computing, 1-20, Physica-Verlag,Wurzburg, 2003,
- *A Fuzzy Valuation-Based Local Search Framework for Combinatorial Problems*, A. Blanco, D. Pelta, J. Verdegay, Journal of Fuzzy Optimization and Decision Making 1 (2), 177-193. 2002

y las referencias citadas en estos artículos....

Métodos Computacionales para la Comparación de Proteínas

David Pelta
http://decsai.ugr.es/~dpelta

Grupo de Trabajo en Modelos de Decisión y Optimización (MODO)

http://decsai.ugr.es/modo

Departamento de CC e IA

E.T.S. Ingeniería Informática

Universidad de Granada