Algoritmo Rápido de Reducción de Trajectorias de Plegamiento de Proteínas

Luis Ernesto Garreta U.

3 de octubre de 2019

Índice

1. Introducción **2**

2. Antecedentes **3**
2.1. Simulaciones de Plegamiento 3
2.2. Algoritmos Rápidos de Agrupamiento 3

3. Datos y Métodos **4**
3.1. Trayectorias de Plegamiento de Proteínas 4
3.2. Métrica de Comparación de Estructuras de Proteínas 4

4. Algoritmo Propuesto **4**

5. Resultados y Discusión **6**
5.1. Reducción sobre una trayectoria completa 6
5.2. Reducción sobre un segmento de una trayectoria 6
5.3. Reducción sobre una trayectoria corta 7

6. Procesamiento paralelo **8**

7. Implementación **9**

8. Conclusiones **9**

Resumen

Gracias a los avances en hardware y software, las simulaciones de plegamiento de proteínas están experimentando grandes progresos, alcanzando tiempos de simulación sin precedentes, en el orden de los microsegundos y milisegundos. Como consecuencia, las trayectorias generadas por estas simulaciones son muy extensas, con miles y millones de conformaciones, lo que genera problemas tanto en tiempo como en espacio para procesarlas y analizarlas. Una manera de sobrepasar estos problemas es desarrollar algoritmos rápidos para generar trayectorias reducidas que preserven tanto como sea posible las características de las trayectorias originales, especialmente el orden temporal y la estructura de las conformaciones.

En este artículo presentamos un algoritmo para reducir este tipo de trayectorias que divide la trayectoria en segmentos y por cada segmento extrae los eventos más disimilares, mediante una estrategia rápida de agrupamiento, y luego selecciona a los más representativos, mediante una estrategia de agrupamiento global. El algoritmo aprovecha el orden temporal implícito en la trayectoria para realizar en cada segmento comparaciones locales y evitar la comparación de todos contra todos, que se vuelve impráctica computacionalmente cuando son muchas conformaciones. De esta manera, el algoritmo reduce muy rápidamente la trayectoria y las conformaciones seleccionadas conservan tanto su estructura como su orden temporal. Además, la partición por segmentos permite al algoritmo reducir cada segmento de forma independiente y paralela, lo que lo vuelve aún más rápido cuando se ejecuta en máquinas multi-core, muy comunes hoy en día.

1. Introducción

Las simulaciones del plegamiento de proteínas han demostrado ser de gran utilidad para estudiar las mecanismos subyacentes de este proceso, el cual permite a una cadena de aminoácidos plegarse hasta alcanzar su estructura tridimensional única y convertirse en una proteína activa habilitada para ejecutar una función biológica. Gracias a los avances en hardware y software, estas simulaciones han experimentado grandes progresos, con tiempos de simulación en el orden de los milisegundos, y llevadas a cabo usando diferentes tecnologías, desde costosas supercomputadoras especializadas [7], hasta, económicos arreglos de tarjetas gráficas [9], e incluso PCs distribuidos alrededor del mundo [3]. Muchas de estas simulaciones alcanzan tiempos que antes no se lograban debido a las limitaciones en los recursos computacionales, y las trayectorias generadas por estas simulaciones se caracterizan por abarcar miles o millones de conformaciones, lo cual a pesar de ser una gran ventaja porque se tiene más detalle del proceso, así mismo es un problema debido al tiempo y recursos computacionales necesarios para analizarlas. Por esta razón, se necesitan nuevos algoritmos capaces de reducir estas trayectorias de una forma rápida, y que logren preservar la mayor información posible tanto a nivel de representación como a nivel de orden temporal de las conformaciones de la trayectoria.

Para realizar estas reducciones se han utilizado diferentes técnicas que básicamente caen en dos enfoques: la reducción de dimensionalidad [1] y el agrupamiento [11], las cuales más que reducir las trayectorias realizan un análisis sobre ellas (Figura 1). Mientras que en el primer enfoque se transforma las conformaciones a una forma simplificada para poder interpretar los resultados; en el segundo, se obtienen conformaciones representativas de grupos de conformaciones con propiedades comunes. Sin embargo, aunque en el primer enfoque se conserva el orden temporal de las conformaciones, su estructura se pierda ya que su representación se modifica; por el contrario, aunque en el segundo enfoque se conserva la estructura de las conformaciones, el orden temporal se pierde ya que los grupos pueden contener conformaciones de tiempos muy diferentes.

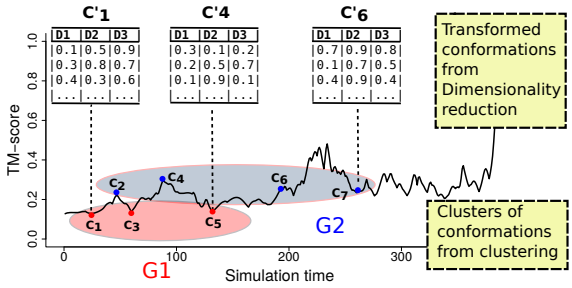


Figura 1: **Perdida de información en las reducciones**. La reducción de la dimensionalidad (Parte de arriba) pierde la información estructural (C_5 se transforma a un nuevo y reducido conjunto de variable C_5'). Mientras que en el agrupamiento (parte de abajo), el orden temporal se pierde (no todas las conformaciones de G1 ocurren antes que las de G2).

En este artículo presentamos un algoritmo rápido para reducción de trayectorias largas de plegamiento de proteínas que toma como base la estrategia de Hobohm&Sander [4] para agrupamientos rápidos y que se basa en tres estrategias: primero una partición de la trayectoria en múltiples segmentos; segundo, una reducción local muy rápida sobre cada una de ellos que aprovecha el orden temporal de las conformaciones; y tercero, una reducción global que busca encontrar las conformaciones más representativas de cada segmento. Estas tres estrategias permiten que este algoritmo sea rápido, fácilmente paralelizable, y produzca trayectorias reducidas que conservan tanto la estructura como el orden temporal de las conformaciones.

2. Antecedentes

2.1. Simulaciones de Plegamiento

Debido a que las simulaciones del plegamiento de proteínas son complejas y demandan gran cantidad de recursos computacionales, hasta hace unos años estas simulaciones se realizaban para proteínas pequeñas y los tiempos simulados eran muy cortos, en el orden de los microsegundos mientras que una proteína se pliega en el orden de los milisegundos [5]. Sin embargo, en los últimos años los avances en el hardware han logrado avances de tal manera que se empiezan a mostrar resultados de simulaciones mucho más largas y de proteínas más grandes. Dos ejemplos de estos avances son los proyectos de folding@home y de la supercomputadora Anton. El proyecto foldin@home logró realizar hace algunos años una de las primeras simulaciones largas utilizando computación distribuida. Una de sus simulaciones alcanzó el orden de los microsegundos para plegar completamente una proteína pequeña, la Villin Headpiece de 36 residuos [8]. Más recientemente, la supercomupatadora Anton ha usado computación paralela y hardware especializado para simular dinámica molecular [12]. Con está máquina se ha logrado plegar completamente varias proteínas medianas (10-80 residuos), alcanzando tiempos de simulación del orden de los milisegundos. En ambos proyectos los resultados de las trayectorias están disponibles para que la comunidad científica los descargue y los analice para avanzar en el entendimiento del plegamiento de las proteínas.

2.2. Algoritmos Rápidos de Agrupamiento

Muchos de los algoritmos para realizar agrupamientos rápidos de secuencias biológicas se basan en el algoritmo de Hobohm&Sander [4] creado inicialmente para agrupar de forma rápida secuencias de proteínas. El algoritmo determina las secuencias más representativas a través de dos actividades: un ordenamiento y una selección rápida. Las secuencias se organizan por longitud en orden descendiente y se selecciona la primera (la más larga) como representativa del primer grupo. Luego, se compara la siguiente secuencia con la representativa y si es similar, de acuerdo a un umbral, se la incorpora al grupo. De lo contrario, pasa a ser la representativa de un nuevo grupo y se hace lo mismo con el resto de secuencias hasta terminar.

Los aspectos determinantes del éxito de este algoritmo son la relación de orden que se establece al inicio y las características que se tomen para comparar las secuencias. En secuencias de ADN y de proteínas, estos aspectos funcionan bien ya que dos secuencias con longitudes cercanas tienen mayor probabilidad de

ser similares que dos secuencias de longitudes muy diferentes. Sin embargo, en el caso de trayectorias de plegamiento, todas las conformaciones tienen la misma secuencia de aminoácidos y por lo tanto son de igual longitud. Esto implica definir la relación de similaridad tomando otras características, como por ejemplo la estructura 3D de las conformaciones, que es diferente para cada conformación, como vamos a ver más adelante cuando describamos el algoritmo de reducción propuesto.

Dos de las implementaciones más usadas de este algoritmo son los programas CD-HIT [6] y UCLUST [2] para agrupamiento rápido de secuencias tanto de nucleóticos como de aminoácidos. CD-HIT realiza un ordenamiento por longitud de la secuencia como lo plantea el algoritmo de Hobhom&Sander, y para la selección utiliza un filtro de palabras cortas–10 aminoácidos en el caso de proteínas–para comparar la similaridad entre dos secuencias–evitando así el alineamiento de las mismas. En cambio UCLUST utiliza para comparar las secuencias una función propia que la llaman como USEARCH que calcula la similitud entre las secuencias a partir de un alineamiento global, es decir un alineamiento que incluye todos los nucleótidos de ambas secuencias.

3. Datos y Métodos

3.1. Trayectorias de Plegamiento de Proteínas

Para mostrar los resultados del algoritmo propuesto se utilizó las trayectorias disponibles de dos proteínas. La primera es la trayectoria de la proteína Trp-cage (PDB: 2JOF), una trayectoria extensa, de más de 1 millón de conformaciones, y simulada con Dinámica Molecular [7]. Mientras que la segunda es la trayectoria de la proteína ferredoxina clostridium acidurici (PDB: 1FCA), una trayectoria corta, de aproximadamente 300 conformaciones, y generada mediante el método Probabilistic Roadmap Method [13]. La reducción se realizó con los parámetros de 40 bins, un umbral de TMscore de 0.5 y un K de 10.

3.2. Métrica de Comparación de Estructuras de Proteínas

Tanto para la reducción local y global el algoritmo utiliza la medida de similitud entre estructuras de proteína llamada TM-score (Template Modeling score) [14]. Esta medida a diferencia de otras ampliamente usadas en comparación de estructuras, como el RMSD (Root Mean Square-Deviation), es más precisa porque evita evaluaciones sesgadas dando poco valor en el puntaje final a las secciones pequeñas de la proteína que alinean incorrectamente, como giros simples o términos flexibles.

4. Algoritmo Propuesto

El algoritmo reduce una trayectoria de plegamiento en cuatro etapas: particionamiento, reducción local, reducción global, y ensamble (Figura 2). En ambas etapas de reducción el algoritmo utiliza la métrica TM-Score para comparar las estructuras de dos conformaciones de proteína (Ver métodos).

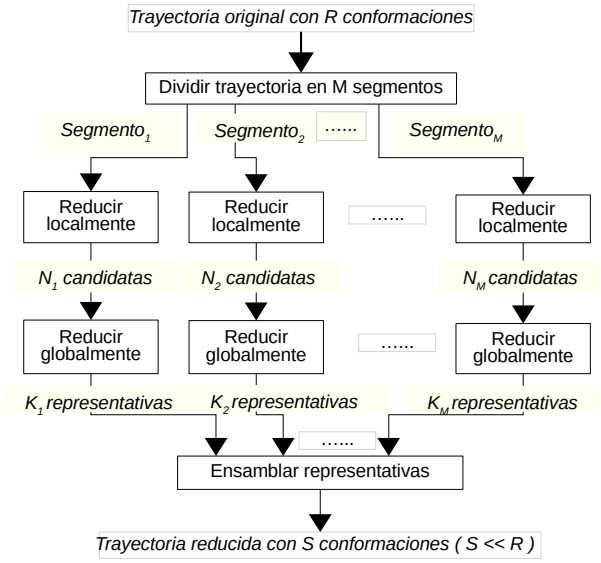


Figura 2: Flujograma del algoritmo propuesto.

El algoritmo primero particiona la trayectoria original en M segmentos de igual número de conformaciones. Después, sobre cada segmento realiza una reducción local rápida (Algoritmo 1, descrito abajo) que extrae N conformaciones candidatas. Para luego, sobre cada conjunto de candidatas realizar un agrupamiento global mediante un algoritmo de particionamiento alrededor de medoides (PAM) [10], que selecciona como representativas a K conformaciones centrales para las cuales la suma de las distancias entre estas y las demás candidatas es mínima. Al final, se ensamblan las conformaciones representativas de cada segmento para producir la trayectoria reducida con S conformaciones (S << R), las cuales conservan tanto su estructura 3D como su orden temporal, ya que el algoritmo organiza las conformaciones seleccionadas por cada segmento de acuerdo a su orden temporal original.

La etapa más importante del algoritmo es la reducción local rápida (Algoritmo 1), donde se extraen rápidamente las conformaciones más disimilares de cada segmento aprovechando su ordenamiento temporal dentro de la simulación. Como consecuencia, las conformaciones que están más cercanas en el tiempo se espera presenten pequeños cambios estructurales y por lo tanto sean más similares que las que están más alejadas. El algoritmo aprovecha está propiedad para realizar un agrupamiento local rápido, pero en vez de extraer un conjunto de conformaciones similares por segmento, extrae las conformaciones más disimilares, que son las que caracterizan los cambios más importantes del proceso de plegamiento en el segmento.

Algoritmo 1 Reducción local rápida para cada segmento.

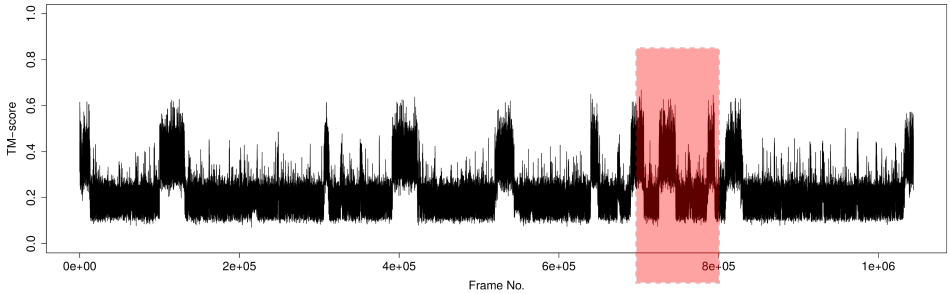
```
1  ALGORITMO ReducciónLocalRápida (entrada: segmento de trayectoria):
2      Tomar del segmento la primera conformación en orden temporal como representativa
3      MIENTRAS existan conformaciones en el segmento:
4          Tomar la siguiente conformación en orden temporal
5          Comparar las estructuras de ambas conformaciones
6          SI son similares, entonces:
7              Remover la segunda conformación
8              Continuar con las siguientes
9          SINO, cuando son similares:
10             Remover del segmento y adicionar la representativa a la trayectoria reducida
11             Asignar la segunda conformación como nueva representativa
12             Continuar con las siguientes
13
14     RETORNAR la trayectoria reducida
```

5. Resultados y Discusión

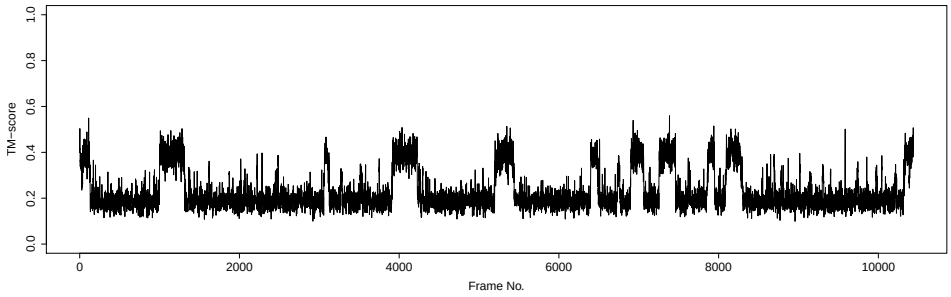
Presentamos los resultados de tres reducciones producidas por nuestro algoritmo: primero, una reducción sobre una trayectoria completa de más de un millón de conformaciones que muestra el grado o porcentaje de reducción que logra el algoritmo; segundo, una reducción sobre un segmento de 100000 (100k) de la trayectoria anterior donde se muestra las reducciones parciales que realiza el algoritmo; y tercero, dos reducciones de trayectorias pequeñas donde se muestra como trabajan las dos fases de reducción local y global del algoritmo.

5.1. Reducción sobre una trayectoria completa

La Figura 3 presenta la reducción realizada sobre la trayectoria completa de la proteína TRP-Cage (PDB 2JOF) con más de 1 millón de conformaciones, donde la trayectoria se reduce en un 99 % (de 1044004 a 10000 conformaciones). Se puede observar que las oscilaciones en el plegamiento presentes en la trayectoria original (figura A), también están en la reducción (figura B), lo que quiere decir que las conformaciones principales de la trayectoria original se preservan en la reducción. La parte sombreada (color rojo) es una subtrayectoria de 100K conformaciones que se utiliza para mostrar en mayor detalle el proceso de reducción en la siguiente sección.



A. Trayectoria original sin reducir (1044004 conformaciones)



B. Trayectoria reducida (10000 conformaciones, 99 % de reducción)

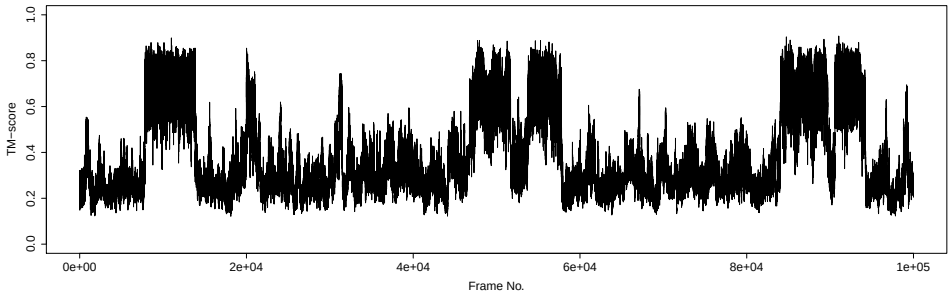
Figura 3: Reducción de una trayectoria larga de plegamiento.

5.2. Reducción sobre un segmento de una trayectoria

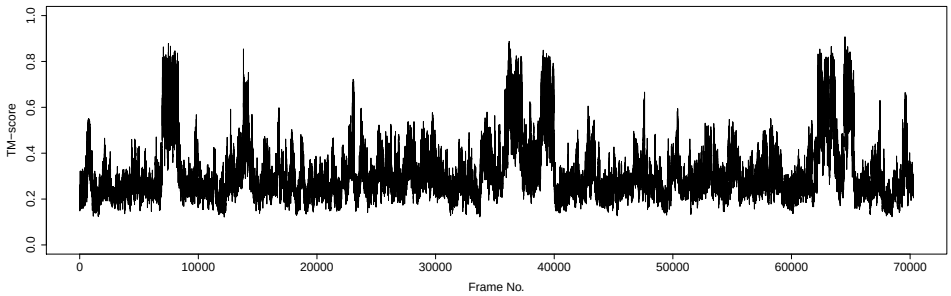
En la figura 4 se presenta las reducciones parciales por etapas de una subtrayectoria de 100k de la trayectoria de las sección anterior (bloque marcado de color rojo, Figura 3). En la figura A se presenta el segmento de trayectoria sin reducir; en la figura B se presenta el resultado de la etapa de reducción local rápida; y en la figura C se presenta el resultado final después de aplicar la reducción global a los resultados de la etapa anterior.

La reducción local es de alrededor del 30 % (de 100000 a 70000 conformaciones) y el agrupamiento rápido que se realiza en esta etapa selecciona tanto eventos principales como otros eventos menos importantes que pueden ya estar representados, por eso se observa que algunas partes de la trayectoria están desproporcionadas con relación a la original. Sin embargo, en la segunda parte del algoritmo, la reducción global es

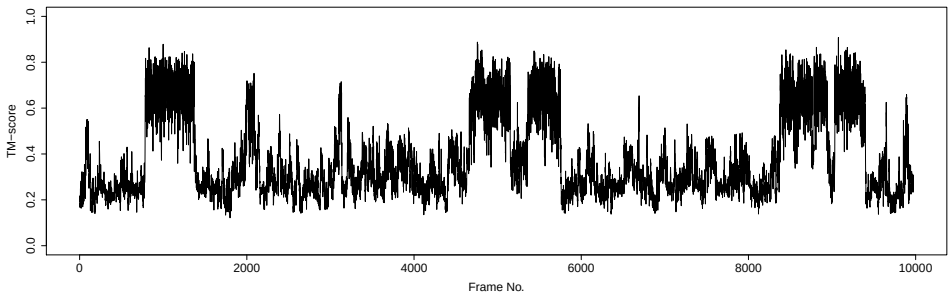
más exhaustiva y alcanza a ser de alrededor del 90 % (de 100000 a 10000 conformaciones), los eventos principales se destacan con mayor claridad, y la amplitud de los mismos se mejora considerablemente respecto al número de conformaciones de la nueva trayectoria.



A. Subtrayectoria original (sin reducir, 100k conformaciones, 100 %)



B. Reducción local rápida (reducción parcial, 70k conformaciones, 70 %)



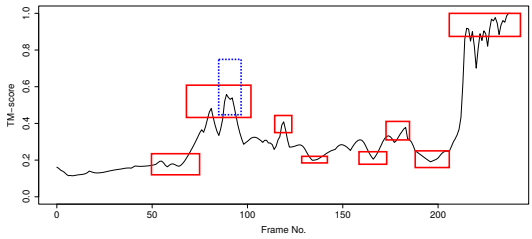
C. Reducción final (reducción global, 10k conformaciones, 30 %)

Figura 4: Reducciones por etapas de una subtrayectoria de plegamiento.

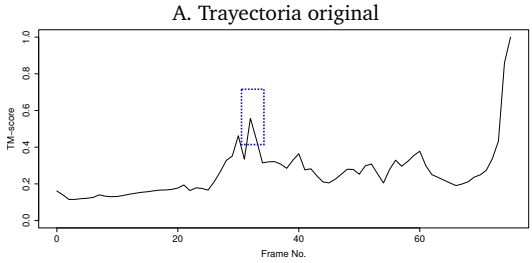
5.3. Reducción sobre una trayectoria corta

En la Figura **??** se presenta la reducción de la trayectoria corta de la proteína 1FCA1 (sección 3.1), donde se señalan los eventos de plegamiento que se presentan en la trayectoria original y como estos se conservan tanto en la reducción local parcial, como en la reducción global final. En la parte superior está la trayectoria original completa; en la parte intermedia la trayectoria después de la reducción local; y en la parte inferior la trayectoria final después de la reducción global.

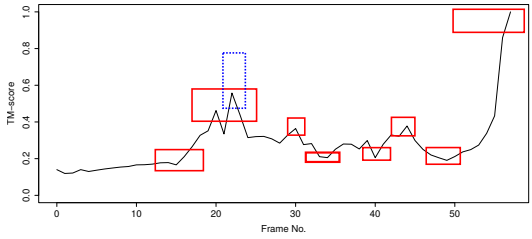
Observamos que los eventos principales se conservan tanto en la reducción local como en la global (recuadros rojos en las trayectorias original y final), reflejando así en las reducciones la dinámica de la trayectoria original. Además, se destaca que en la reducción local (figura intermedia), los eventos principales tienden a desplazarse frente a los originales (recuadros azules). Lo cual se logra corregir en la reducción final ya que la reducción local por ser rápida incluye conformaciones tanto de eventos principales como secundarios, mientras que la global se enfoca en dejar solo los eventos principales y por lo tanto el desplazamiento se reduce. La reducción lograda es del orden de más del 76 %, de 239 a 57 conformaciones.



A. Trayectoria original



B. Reducción local rápida



C. Reducción global

Figura 5: Reducción detallada para la trayectoria de proteína 1FCA1.

6. Procesamiento paralelo

La estructura de nuestro algoritmo rápido de reducción es altamente paralelizable y por lo tanto su desempeño mejora bastante a medida que utiliza más de un procesador. Para ver este desempeño en la figura 6 mostramos los resultados de reducir un segmento de una trayectoria de 100000 conformaciones variando el número de procesadores desde 1 hasta 40 procesadores. En la gráfica puede observarse claramente la disminución de tiempo cuando de un procesador pasa a ajecutarse en paralelo con 2 y 5 procesadores. La ejecución se reduce de más de 16 minutos a la mitad del tiempo con dos procesadores y a casi menos de 1 minuto con 40 procesadores.

Nro. Procesadores	Tiempo (Segundos)
1	977
2	491
5	197
10	166
20	73
30	70
40	59

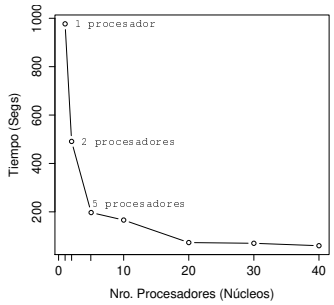


Figura 6: Tiempos de ejecución algoritmo de agrupamiento rápido.

De acuerdo a la estructura del algoritmo, inicialmente la trayectoria se divide en múltiples segmentos, que luego los reduce, primero a través de una reducción local y después una reducción global. La división en segmentos es la que permite paralelizar el algoritmo ya que cada segmento se toma como un trozo de trayectoria independiente de los demás que puede ser manejado por una unidad de proceso distinta.

7. Implementación

Casi todo el algoritmo está implementado en el lenguaje R excepto la comparación entre pares de proteínas, que es la parte que más veces se ejecuta y que está implementada en el lenguaje Fortran tomando como base el programa TM-score de Zhang&Skolnick [14]. Estas comparaciones se realizan tanto en la fase de reducción local como en la global.

8. Conclusiones

En este trabajo presentamos una algoritmo para reducción de trayectorias largas de plegamiento de proteínas que se caracteriza por ser rápido y paralelo. El algoritmo produce reducciones donde la dinámica de la trayectoria original se preserva en cuanto a los eventos principales y la relación temporal de los mismos. El algoritmo tiene tres fases: particionamiento, reducción local, y reducción global. El particionamiento crea segmentos de trayectoria que se reducen de forma independiente y paralela. La reducción local aprovecha el ordenamiento temporal de las conformaciones para extraer los eventos principales sin necesidad de realizar todos los pares de comparaciones entre conformaciones. El algoritmo usa la métrica TM-score, que es más robusta que el RMSD para comparar las estructuras de proteínas. El TM-score produce mejores resultados a la hora de comparar estructuras de conformaciones muy cercanas, que es lo que se tiene en una trayectoria de plegamiento donde las conformaciones están temporalmente muy cercanas. La implementación del algoritmo se realizó en el lenguaje R pero la comparación de estructuras (función TM-score) se implementó en Fortran ya que es la que más se ejecuta tanto en la reudcción local rápido, como en la reducción global.

Referencias

[1] Mojie Duan, Jue Fan, Minghai Li, Li Han, and Shuanghong Huo. Evaluation of Dimensionality-reduction Methods from Peptide Folding-unfolding Simulations. *Journal of chemical theory and computation*, 9(5):2490–2497, may 2013.

[2] Robert C. Edgar. Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, 26(19):2460–2461, 2010.

[3] Daniel L Ensign, Peter M Kasson, and Vijay S Pande. Heterogeneity even at the speed limit of folding : Large-scale molecular dynamics study of a fast-folding variant of the villin headpiece. *Journal of molecular biology*, 374(3):806–816, 2007.

[4] Uwe Hobohm, Michael Scharf, Reinhard Schneider, and Chris Sander. Selection of representative protein data sets. *Protein Science*, 1(3):409–417, mar 1992.

[5] Fabien P E Huard, Charlotte M Deane, and Graham Wood. Modelling sequential protein folding under kinetic control.

[6] W. Li, L. Jaroszewski, and A. Godzik. Tolerating some redundancy significantly speeds up clustering of large protein databases. *Bioinformatics*, 18(1):77–82, 2002.

[7] Kresten Lindorff-Larsen, Stefano Piana, Ron O. Dror, and David E Shaw. How fast-folding proteins fold. *Science*, 334(6055):517–520, oct 2011.

[8] A. Marsden. M. Lougher, M. Lücken, T Machon, M. Malcomson. Computational Modelling of Protein Folding. Technical report.

[9] Hai Nguyen, James Maier, He Huang, Victoria Perrone, and Carlos Simmerling. Folding simulations for proteins with diverse topologies are accessible in days with a physics-based force field and implicit solvent. *Journal of the American Chemical Society*, 136(40):13959–13962, oct 2014.

[10] Hae-Sang Park and Chi-Hyuck Jun. A simple and fast algorithm for K-medoids clustering. *Expert Systems with Applications*, 36(2):3336–3341, mar 2009.

[11] Jun-hui Peng, Wei Wang, Ye-qing Yu, Han-lin Gu, and Xuhui Huang. Clustering algorithms to analyze molecular dynamics simulation trajectories for complex chemical and biological systems. *Chinese Journal of Chemical Physics*, 31(4):404–420, aug 2018.

[12] David E Shaw, Jack C. Chao, Michael P. Eastwood, Joseph Gagliardo, J. P. Grossman, C. Richard Ho, Douglas J. Lerardi, István Kolossváry, John L. Klepeis, Timothy Layman, Christine McLeavey, Martin M. Deneroff, Mark A. Moraes, Rolf Mueller, Edward C. Priest, Yibing Shan, Jochen Spengler, Michael Theobald, Brian Towles, Stanley C. Wang, Ron O. Dror, Jeffrey S. Kuskin, Richard H. Larson, John K. Salmon, Cliff Young, Brannon Batson, and Kevin J. Bowers. Anton, a special-purpose machine for molecular dynamics simulation. *Communications of the ACM*, 51(7):91, 2008.

[13] Guang Song and Nancy M Amato. Using Motion Planning to Study Protein Folding Pathways. *Journal of Computational Biology*, pages 287–296, 2001.

[14] Yang Zhang and Jeffrey Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 68(4):1020, 2007.