Algoritmos Rápido de Clustering para Trajectorias de Plegamiento de Proteínas

August 1, 2018

Introducción

Contexto: Actualmente se están liberando con más frecuencia datos de simulaciones de plegamiento de proteínas para que la comunidad científica los analice y avance en el entendimiento de este proceso. Estas simulaciones alcanzan tiempos de simulación que antes no se lograban debido a las limitaciones en los recursos computacionales. Hace algunos años el proyecto folding@home [1] liberó varias trayectorias de la simulación de la proteína Villin Headpiece la cual alcanzó el orden de los microsegundos utilizando computación distribuida. Más recientemente, el grupo de David Shaw liberó simulaciones de varias proteínas en el orden de los milisegundos utilizando la supercomputadora Anton diseñada especialmente para simular dinámica molecular [2]. Todas estas simulaciones se caracterizan por generar trayectorias que abarcan miles o millones de conformaciones, lo cual por un lado es una gran ventaja porque se tiene más detalle del proceso, pero así mismo es un problema debido al tiempo y recursos computacionales necesarios para analizarlas.

Necesidad Para tratar este problema se buscan conjuntos de conformaciones representativas, que generalmente utilizan métodos de agrupamiento que construyen una matriz con las distancias entre cada una de las conformaciones, usualmente se usa la distancia RMSD. Estas agrupamientos se vuelven muy costosos en tiempo y recursos computacionales cuando se trata de muchas conformaciones y por esta razón los algoritmos buscan simplificar estos costos. Una forma es reducir el número de átomos que comparar en las conformaciones (solo carbonos alfa), otra forma es crear agrupamientos rápidos que no tengan que comparar todas las conformaciones. Este mismo problema pero no con estructuras de proteínas sino con secuencias de ADN y proteínas se ha trabajado a través de agrupamientos rápidos basados en el algoritmo de Hobohm&Sander. Este algoritmo realiza un primer agrupamiento muy rápido sin comparar todas las conformaciones para generar un conjunto de secuencias representativas, luego

realiza un segundo agrupamiento–mas detallado–con las conformaciones representativas resultantes del agrupamiento rápido.

Tarea: Nosotros hemos tomado como base este algoritmo para crear un algoritmo rápido de agrupamiento para el caso de las conformaciones 3D resultantes de las trayectorias de simulaciones de plegamiento de proteínas que estamos utilizando para reducir las trayectorias de plegamientos simuladas en la supercomputadora Anton [2]. Las reducciones buscan realizar un cambio de escala del paso de simulación mediante el agrupamiento de las conformaciones pertenecientes a un rango de tiempo que determina una escala, por ejemplo llevar un paso de simulación de picosegundos a nanosegundos, es decir reducir un orden de 1000.

0.1 Contexto

- El plegamiento de proteínas es un proceso que puede llevarle a la proteína algunos milisegundos pero que al simularlo computacionalmente podría abarcar tiempos supremamente mayores de días o meses. Hasta hace algunos años las simulaciones reportadas eran de la escala de algunos nanosegundos a microsegundos, sin embargo esto ha ido cambiando con el surgimiento de nuevas técnicas y supercomputadoras diseñadas especialmente para este tipo de simulaciones. Por ejemplo, el proyecto folding@home [1], empleando una nueva técnica de computación distribuida, a simulado el plegamiento de proteínas pequeñas (36 residuos) en el orden de microsegundos (500 us), lo cual hasta hace algunos años eran escalas de tiempo imposible de alcanzar. Así mismo, la supercomputador Anton [2], diseñada especialmente para este tipo de simulaciones, ha logrado simular el plegamiento de varias proteínas (de 10 a 92 residuos) en el orden de los milisegundos.

- Todas estas simulaciones producen miles o millones de conformaciones o *snapshots* que para su análisis computacional se necesita gran cantidad de tiempo de máquina y recursos computacionales que muchas veces o son muy extensos (horas o días) o no están disponibles fácilmente (supercomputadoras o clusters). Par

- que pueden no estar disponibles y por lo tanto se necesita reducir esas trayectorias

- Dependiendo del nivel de detalle con el que se quiera estudiar el plegamiento,

- Con la

-

- El supercomputador Anton es una máquina especialmente diseñada para simular el plegamiento de la proteína, es decir simular los cambios en la estructura tridimensional de una proteína en un periodo de un milisegundo, la cual es una escala bastante superior comparado con simulaciones realizadas utilizando otras técnicas y recursos.

- Los resultados obtenidos se representan con la trayectoria de la proteína, es decir, cientos o miles de secuencias de proteínas que indican la posición de cada uno de sus átomos en un instante de tiempo.

- En el caso de las proteínas el caso no es la excepción. Cada vez más se tiene acceso a servidores con gran cantidad de secuencias de proteínas listas para ser procesadas y analizadas. Algunos centros de investigación, que cuentan con gran capacidad de cómputo, colocan a disposición de la comunidad científica datos que han sido procesados en sus máquinas, los cuales pueden ser accesados libremente a través de una descarga o previa solicitud de los centros.

1 Marco Teórico

En esta sección describiremos los elementos básicos que se manejan en este trabajo, principalmente hablaremos de plegamiento de proteína, simulaciones de plegamiento de proteínas, trayectorias de plegamiento y métodos de reducción de datos biológicos.

1.1 Métodos de Comparación de Estructuras de Proteínas

Revisar [Kufareva2015]

1.1.1 RMSD

1.1.2 TM-score

1.2 Plegamiento de Proteínas

Revisar sección 2.2 Tesis LG[1].

1.3 Simulaciones de Plegamiento

Revisar secciones 2.7, especialmente la 2.7.1. de Tesis LG [1]

1.4 Simulaciones Largas de Plegamiento

Las simulaciones del plegamiento de proteínas son complejas y demandan gran cantidad de tiempo y recursos computacionales. Debido a estas limitaciones tecnológicas, las simulaciones del plegamiento de proteínas hasta hace unos años 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 [**?**]. Sin embargo, en los últimos años los avances en el hardware han logrado algunos avances de tal manera que se empiezan a mostrar resultados de simulaciones 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 [2]. La supercomupatadora Anton es un proyecto más reciente (2010) que usa computación paralela y hardware especializado para simular dinámica molecular. Con está máquina se ha logrado plegar completamente varias proteínas medianas (10-80 residuos), alcanzando tiempos de simulación del orden de los milisegundos [3]. 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 Antecedentes

Describimos a continuación el algoritmo rápidos de agrupamiento propuesto por Hobhon & Sander, mostrando sus principales características, después describimos como algunas herramientas han implementado este algoritmo para agrupar secuencias de ADN o de Proteínas, y finalmente planteamos los elementos que tomamos de este algoritmo y de sus implementaciones para crear nuestro algoritmo para agrupar conformaciones de proteínas de trayectorias de simulación [3][3].

2.1 Algoritmo de Hobhon y Sander

Este algoritmo se creó inicialmente para agrupar de forma rápida secuencias de proteínas [?], y busca las secuencias más representativos a través de dos actividades: un ordenamiento y selección rápida. En la primera actividad, el algoritmo ordena las secuencias por longitud en orden descendiente, luego toma la primera secuencia (la más larga) y la toma como representativa del primer grupo. En la segunda actividad, la selección viene dada comparando el resto de secuencias con la representativa e incorporándola al grupo si son cercanas (e.g. si son similares a nivel de secuencias). Si la secuencia que se está comparando no es muy cercana, entonces está 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 del algoritmo son la relación de orden que se establezca al inicio y las propiedades que se tomen para comparar las secuencias. En secuencias de ADN y de proteínas estos aspectos funcionan bien ya que dos secuencias de más o menos de igual longitud tienen mayor probabilidad de ser similares que dos secuencias de longitudes completamente diferentes. Sin embargo en estructuras tridimensionales de proteínas que pertenecen a una misma trayectoria, la longitud y la similaridad de la secuencia va a ser la misma para todas la conformaciones, lo que implica redefinir estos aspectos en términos de las características de las estructuras 3D de proteínas de una misma trayectoria, como vamos a describir más adelante cuando mostremos nuestra implementación del algoritmo.

2.2 Implementaciones del Algoritmo de Hobhon y Sander

Dos de las implementaciones más usadas para agrupamiento rápido de secuencias son los programas CD-HIT y UCLUST. El programa CD-HIT [?] realiza un ordenamiento por longitud de la secuencia como lo plantea el algoritmo de Hobhon, y para la selección utiliza un filtro de palabras cortas para comparar si dos secuencias son similares–evitando el alineamiento de las mismas–y así asignarlas a un mismo grupo o crear uno nuevo. En el caso de secuencias de proteínas el programa usa por defecto una palabra de 10 aminoácidos o *decapeptido*. En cambio el programa UCLUST [?]utiliza para comparar las secuencias una función creada por los mismos autores que la llaman como USEARCH y que calcula la similitud entre las secuencias a partir de un alineamiento global.

En el caso de estructuras de proteínas provenientes de trayectorias, la selección se puede realizar usando la distancia media cuadrática mínima o RMSD con umbrales muy altos, ya que dos conformaciones que ocurran muy cerca en el tiempo (eg. tn y tn+1) van a presentar cambios mínimos en su configuración (un leve desplazamiento de los átomos), y por lo tanto deberían estar en un mismo grupo. Sin embargo, si el umbral de RMSD se sobrepasa, esto indica que las conformaciones tiene diferencias apreciables y por lo tanto se debería iniciar otro grupo.

3 Datos y Métodos

Aquí describimos las características de la trayectoria que utilizamos para realizar la reducción.

3.1 Trayectorias de Plegamiento de Proteínas generada por Anton

Estas simulaciones fueron realizadas por en una supercomputador especialmente diseñado para resolver problemas de Dinámica Molecular, llamada Anton [?], que es el método de más utilizado para simular el plegamiento de proteínas. Las trayectorias puestas a disposición corresponden a las simulaciones del plegamiento completo en solvente explicito de 12 proteínas de 10 a 02 residuos [?]. Por cada

4 Algoritmo de Reducción de Trayectorias de Plegamiento

El algoritmo contiene tres fases: primero inicia particionando la trayectoría en secciones o *bins* que van a contener un número N de estructuras de proteínas contiguas en el tiempo; después sobre cada bin se hace un agrupamiento local rápido que realiza una primera reducción; y luego con las estructuras resultantes se realiza un agrupamiento global detallado que produce las estructuras finales.

Para la partición de la trayectoría se toma para cada *bin* un número M de estructuras contiguas en la trayectoría y se las asigna al *bin*, de esta manera las primeras M estructuras se ubicarán en el bin1, las segunda M se ubicarán en el bin2, y así sucesivamente, creándose un número de *bins* igual al número total de estructuras sobre el tamaño M de estructuras por bin. Por ejemplo si la trayectoría tiene N=1000 estructuras y se escoje M=200, entonces se crearán K=5 *bins,* cada uno con 200 estructuras (1000/2000.

El agrupamiento local rápido es la clave de este algoritmo de reducción. Aquí se aprovecha el ordenamiento temporal de las estructuras implícito en la trayectoría para formar grupos donde se toma inicialmente la primera estructura del *bin* (estado inicial) como representativa del primer grupo. Después, la siguiente estructura dentro del *bin,* en orden de tiempo, se compara con la última representativa y si son semejantes de acuerdo a una métrica y a un umbral

predeterminado, entonces la estructura se asigna a este grupo, de lo contrario se forma uno nuevo que toma a esta ultima estructura como la nueva representativa. La métrica para las comparaciones en esta fase es el *TM-score* que tiene en cuenta las propiedades globales de plegamiento y por lo tanto es adecuada para agrupar estructuras que están más alejadas o separadas tiempos más largos dentro de la trayectoria. Este proceso se sigue con las siguientes estructuras pero teniendo en cuenta que las comparaciones se realizan solo con las estructuras representativas de cada grupo y no con las estructuras que conforman el grupo, lo cual reduce el número de comparaciones en gran medida frente a un algoritmo convencional de agrupamiento. Además, para evitar más comparaciones, estas se realizan de atrás hacia adelante, es decir, las estructuras se comparan con la última representativa y si son semejantes entonces se agrega al grupo y no se siguen las comparaciones. Esto debido a que a medida que avanza la simulación de plegamiento, una nueva estructura no es más que una modificación de la anterior y por lo tanto se espera que sea más semejante a esta última.

En la tercera fase se toman las estructuras representativas de lo grupos formados previamente y sobre esas estructuras se seleccionrealiza un nuevo agrupamiento utilizando un algoritmo *k-medoides* usando como métrica el *TM-score*

grupos que también tiene en cuenta la posición de los átomos entre las estructuras pero es menos sensible a las variaciones estructurales locales. La métrica para las comparaciones en esta fase es el TM-score que tiene en cuenta las propiedades globales de plegamiento y por lo tanto es adecuada para agrupar estructuras que están más alejadas o separadas tiempos más largos dentro de la trayectoria.

En la segunda fase, el algoritmo toma las estructuras representativas de cada grupo y realiza con ellas un nuevo agrupamiento utilizando la métrica TM-score que tiene en cuenta propiedades globales del plegamiento y que no es tan sensible a variaciones locales como sucede con el RMSD. Este nuevo agrupamiento forma K grupos de los cuales se toman las estructuras centrales o medoides como representativas y que finalmente serán las que después del proceso de reducción representan a todo el *bin.*

El algoritmo es fácilmente paralelizable ya que una vez particionada la trayectoría el proceso de reducción es el mismo para cada sección o *bin,* lo que permite que el procesamiento se reparta sobre cada *bin*, es decir, tanto la reducción local como la reducción global se ejecutan al mismo tiempo sobre cada *bin* y por lo tanto si existen N *bins*, cada uno de ellos se podría asignar a un proceso, hilo, o procesador.

5 Detalles de Implementación

El algoritmo está implementado como a través de tres scripts:

- pr00_main.py: Script principal en lenguaje Python que toma los parámetros iniciales y llama a los otros scripts envíandole los parámetros necesarios.

- pr01_createBins.py: Script en lenguaje Python que realiza la partición
- pr02_localReduction.R : Script en lenguaje R que realiza la reducción local.
- pr03_globalReduction.R: Script en lenguaje R que realiza la redución global..

La reducción local utiliza la función *rmsd* de la librería *bio3d* para el sistema R, y la reducción global utiliza el programa TMscore [http://cssb.biology.gatech.edu/skolnick/webservice/TM-score/index.shtml]

6 Resultados y Discusión

Utilizando los datos de la trayectoria descrita en la sección 2, aquí mostramos como estas escala escala al realizar agrupamientos de bloques de 1000, es decir de picosegundos a nanosegundo y después a milisegundos. A continuación mostramos los agrupamientos utilizando diferentes métodos para de tres bloques de 1000 conformaciones de la trayectoria: el primer bloque corresponde a las primeras 1000 conformaciones; el segundo bloque corresponde a las 1000 conformaciones de la mitad; y el tercer bloque corresponde a las 1000 ultimas conformaciones.

6.1 Escalamiento de la trayectoría

6.2 Agrupamiento k-means

6.3 Agrupamiento jerárquico

6.4 Agrupamiento rápido

7 Conclusiones

References

[1] Luis Garreta. *Conformational Folding Status and Folding Levels Based on Global Protein Properties : a Computational Approach*. PhD thesis, Universidad del Valle, 2015.

[2] A. Marsden. M. Lougher, M. Lücken, T Machon, M. Malcomson. COMPUTATIONAL MODELLING OF PROTEIN FOLDING. Technical report.

[3] David E Shaw, Martin M Deneroff, Ron O Dror, Jeffrey S Kuskin, Richard H Larson, John K Salmon, Cliff Young, Brannon Batson, Kevin J Bowers, Jack C Chao, and Others. Anton, a Special-Purpose Machine for Molecular Dynamics Simulation. *COMMUNICATIONS OF THE ACM*, 51(7), 2008.