ANÁLISIS DE HERRAMIENTAS BIOINFORMÁTICAS PARA EL ESTUDIO DEL CÁNCER EN EL LABORATORIO NACIONAL DE SUPERCÓMPUTO DEL SURESTE DE MÉXICO.

Robles Morales Fernando[a], González López Karina[a]

[a]Laboratorio Nacional de Supercómputo del Sureste de México, Benemérita Universidad Autónoma de Puebla, Aptdo.Postal 1152, 72570, Puebla, Pue. México.

RESUMEN

La bioinformática evoluciona con el análisis de la información genómica y proteómica; actualmente se estudian las interacciones proteína-proteína y proteína-ADN. Se ha visto que las investigaciones más rigurosas y relevantes que apoyan en el desarrollo del campo de las ciencias de la salud, es la simulación de plegamiento de proteínas y las interacciones que sufren una vez alcanzada su forma tridimensional.

Por lo anterior, para el completo entendimiento de estos procesos biológicos que aportan avances en las ciencias de la salud, en el Laboratorio Nacional de Supercómputo del Sureste de México se analizan diferentes herramientas bioinformáticas para el estudio de proteínas. En particular, de forma de inicial, se colabora con el proyecto "Folding@home" (FAH o F@h)[1], que es un proyecto de computación distribuida fundado por Pande Lab, de la Universidad de Stanford. Los principales temas de investigación del proyecto son plegamiento de proteínas, diseño computacional de medicinas y distintos tipos de dinámica molecular.

Unos de los principales intereses dentro de nuestro grupo de investigación es la aplicación de herramientas bioinformáticas para el estudio del cáncer, aprovechando así, los recursos de la supercomputadora del LNS. Por lo anterior, nos hemos enfocado a brindar nuestro capacidad de procesamiento al Proyecto "Cancer and p53" donde se estudian dominios específicos del gen p53, que se encuentra regularmente en la población que padece algún tipo de cáncer, con el fin de predecir las mutaciones relevantes que generan cualquier tipo de cáncer.

El Laboratorio Nacional de Supercómputo del Sureste de México busca desarrollar y optimizar herramientas bioinformáticas de este tipo para disminuir los costos de tratamientos del cáncer y otras enfermedades crónico-degenerativas, así como facilitar la detección y prevención de estas.

1. INTRODUCCION

El proyecto "Folding@home" (FAH ó F@h)[1], realiza el estudio de enfermedades crónico-degenerativas con base en el proceso de plegamiento y agregación de las proteínas. De las enfermedades que se estudian en este proyecto, tales como el Alzheimer, Huntington, Chagas, Malaria, Ostogénesis Imperfecta, Diabetes, Mal de Parkinson y distintos tipos de Cáncer, se ha encontrado que estas se originan por una anomalía en el plegamiento y agregación de la proteína.

En el caso específico del estudio del cáncer, en un principio se encontró que distintas mutaciones del gen p53[2] se encuentran en varios tipos de cáncer. En el espectro de mutaciones de este gen aparece el cáncer de colon, pulmón, esófago, mama, hígado, cerebro, tejidos reticuloendoteliales y hematopoyéticos. Por otro lado, estudios recientes han encontrado que las proteínas-tirosinas quinasas (PTKs por sus siglas en inglés)[3], dependiendo de las señales que emitan o reciban, pueden gobernar el crecimiento extracelular que usualmente genera el cáncer. Por ejemplo la proteína-tirosina quinasa receptora (RPTKs), es una clase de transmembrana susceptible a los estímulos de la actividad de los PTKs, cuando esta es mutada o alterada, la RPTK puede convertirse en una potente oncoproteína, ocasionando una transformación celular, o de manera inversa, actuar inhibiendo los mecanismos de la activación de generación extracelular.

Las PTKs tienen la habilidad de activar (regular de forma ascendente) o desactivar (regular de forma descendente) estados de acuerdo a las señales que emite un mecanismo de interruptores que provienen de una trayectoria de transducción celular, es decir, aquello que regula el crecimiento celular su proliferación y diferenciación. La desregulación de las PTKs puede conducir a un descontrol de la proliferación y transformación de células malignas que se observan en muchos tipos de cáncer. El reto de encontrar la manera de seleccionar la inhibición de las proteínas quinasas cobra importancia cuando este puede aplicarse de forma efectiva en un tratamiento contra el cáncer.

Se han identificado diferencias estructurales entre los estados activos e inactivos de acuerdo a la orientación y conformación de la helix-C en el lóbulo amino-terminal y en el despliegue del bucle de activación (A-loop) en el lóbulo carboxi-terminal (Figura 1)[4]. Estas diferencias estructurales revelan las conformaciones de activación y desactivación, pero hace falta entender el mecanismo de activación, es decir, qué estados intermedios están a lo largo de la trayectoria de activación, y realizar una trayectoria hacia dos estados finales respecto al tiempo para activar/desactivar las proteínas quinasas.

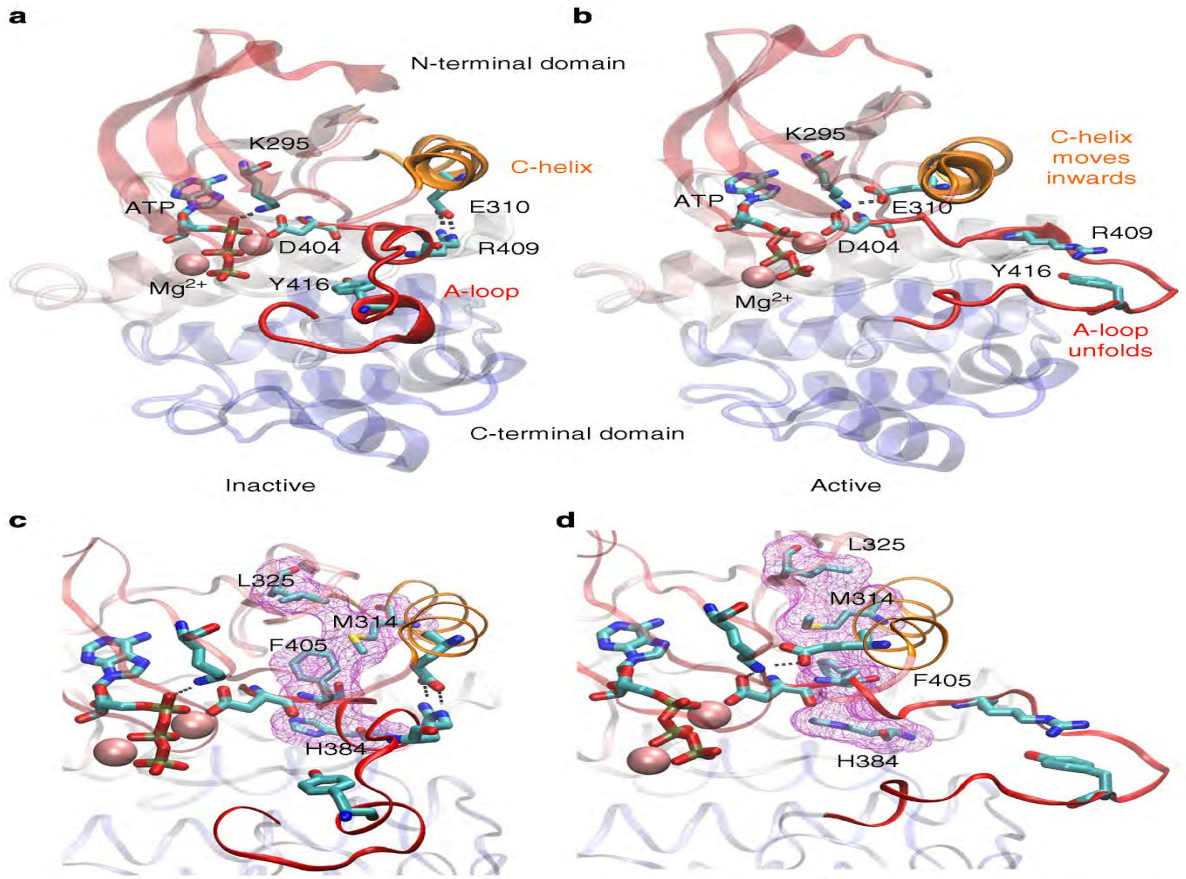

Figura 1. Cambios de conformaciones asociados con la activación de la proteína c-src. Las estructuras cristalinas (a) inactiva y (b) activa muestran los cambios estructurales en el bucle de activación (A-loop, color rojo), y la (C-helix, color naranja), esto intercalando en la red electroestática formada entre Lys295, Glu310, Arg409 y Tyr416, junto con los residuos L325, M314, F405 y H384 formando una columna dorsal reguladora hidrofóbica en el estado activo (d) que se compara con la conformación del estado inactivo (c). Esta columna dorsal hidrofóbica crea una región de conexión entre los dos lóbulos del dominio catalítico, siendo esencial para la actividad que realiza la quinasa.

En el estudio del mecanismo de activación/desactivación se emplea la simulación de dinámica molecular (MD), con el fin de obtener las conformaciones de las transiciones de las proteínas quinasas, el cálculo de la energía liberada de los inhibidores y determinar las trayectorias de activación. Hasta el momento las simulaciones han sido exitosas en el identificar estados metaestables de proteínas, sin embargo aún no se ha logrado estimar la cinética de las conformaciones de transición. Para realizar los cálculos, se emplea un entorno computacional para acoplar los conjuntos de trayectorias en las transiciones de los microestados, por medio de simulaciones de MD distribuidas utilizando el software Folding@home; el método de MD utiliza Modelos de Estados de Markov (MSMs por sus siglas en inglés) así como algoritmos de muestreo para obtener las conformaciones de los estados de transición de la proteína quinasa.

En este trabajo se estudia la herramienta bioinformática Folding@home específicamente en la implementación de los MSMs en el estudio de las proteínas quinasa.

2. TEORIA

En la simulación de desdoblamiento de proteínas, los MSMs representan una herramienta que facilita la interpretación de grandes conjuntos de datos en MD. En la fase de muestreo se parametrizan los estados de transición con base de los estados resultantes, para definir una red dinámica, conformando así un conjunto de metaestados. Los pasos para crear una MSM son:

1. Definir la métrica de distancias para realizar la agrupación en microestados. En este caso se usa la distancia del valor cuadrático medio (RMSD por sus siglas en inglés) entre dos átomos.

2. Para probar la cinética de los microestados, se crean agrupaciones (Clusters) de los átomos considerando la distancia RMSD, calculada en el espacio, a lo largo del intervalo de tiempo de observación en el modelo de la proteína de estudio. Los métodos de agrupación más utilizados son k-Centers (Figura 2)[5] y k-Medoids (Figura 3)[5].

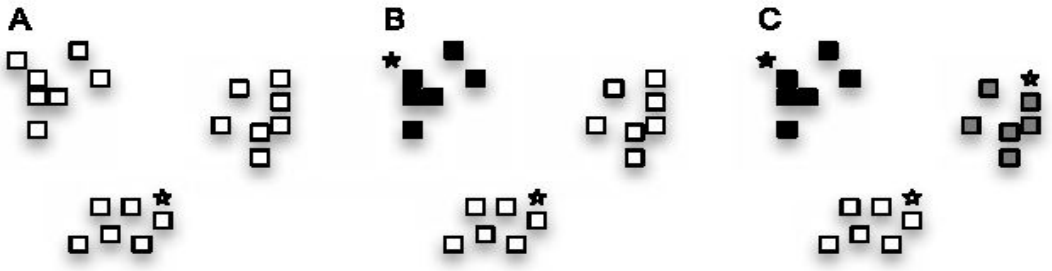

Figura 2. Muestra un ejemplo del método de agrupación k-Centers. (A) Un punto aleatorio (estrella blanca) se escoge como el agrupamiento inicial del centro y todos los puntos se asignan a este (cuadros blancos). (B) El punto más lejano de la agrupación previa se escoge para la siguiente agrupación (estrella negra). Todos los puntos están cerca del nuevo grupo que el existente y son asignados al nuevo centro (cuadros negros). (C) El algoritmo continúa escogiendo puntos más lejanos de su centro (estrella gris) y asigna los puntos más cercanos a este (cuadros grises).

3. Estimar el modelo de transición de la matriz de probabilidades de los microestados. El número de transiciones entre cada par de estados puede ser contado y guardado como una matriz de conteo (C), donde C_{ij} es el número de transiciones observadas del estado i al estado j.

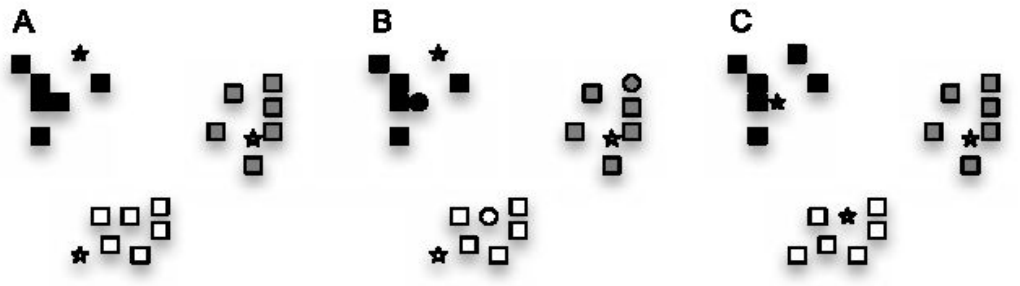

Figura 3. Ejemplifica la agrupación k-medoids. (A) Se decide cuántos grupos se van a construir (En este caso k=3). Entonces k puntos aleatorios se escogen como los puntos iniciales (las tres estrellas). Todos los puntos están asignados al punto central más cercano. (B) Un punto aleatorio en cada grupo se propone como un nuevo centro del grupo (círculos). (C) Si el nuevo centro propuesto es más cercano en promedio a todos los puntos en la agrupación que el grupo previo, entonces se escoge como el nuevo centro para ese grupo (grupos blancos y negros). De otra manera se descarta y se coloca en el otro grupo (grupo gris). Al final todos los puntos son asignados a un centro más cercano.

Cuando es un gran volumen de puntos , se puede estimar la máxima probabilidad entre cada par de estados convirtiendo la matriz de conteo en una matriz de probabilidades de transición, T, (Ecuación 1).

$$T_{ij}(\tau) = \frac{C_{ij}}{\sum_k C_{ik}}$$

Ecuación 1. Matriz de probabilidades de transición, donde τ es el intervalo de tiempo de observación del modelo.

El conteo debe considerar un intervalo de tiempo de observación apropiado, para lograr estimar las matrices de transición (Figura 4)[5].

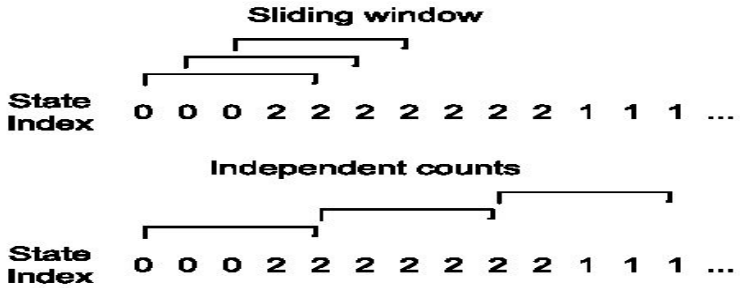

Figura 4. Ejemplo de dos métodos de conteo de transiciones, asumiendo un lapso de tiempo de 4 pasos. En cada panel se muestra una trayectoria como una serie de índices (asumiendo que los estados han sido indexados de 0 a n-1). El panel superior muestra cómo las primeras 3 transiciones estarían contadas usando una ventana de desplazamiento; todas las transiciones son del estado 0 al 2. El panel inferior muestra cómo asegurar conteos independientes; las transiciones van desde el estado 0 al 2, del estado 2 al 2 y del estado 2 al 1.

4. Granular el modelo para crear cualquier cantidad de modelos macroestados o modelos cualitativos. Se puede construir modelos de mesoescalas que solo son predictivos cuantitativamente a partir de los modelos de microestados pero no son compactos. Con un número

suficiente de macroestados se puede comprender y obtener una idea del funcionamiento dell sistema y generar las hipótesis, pero estas no son cuantitativamente predictivas. Debido a esto, se definen métodos para determinar como crear los macroestados, uno de los más utilizados es el Análisis de Agrupamiento de Agrupamientos Perron PCCA . (Figura 5[5]).

El PCAA utiliza un espectro de eigenvalores de la matriz de probabilidades para construir los macroestados, los cuales hacen referencia al eigenvalor más cercano para un intervalo dado. Los eigenvalores de una matriz de probabilidades pueden ser convertidos en escalas de tiempo. El eigenvalor más grande es siempre 1 para un modelo consistente que está conectado.

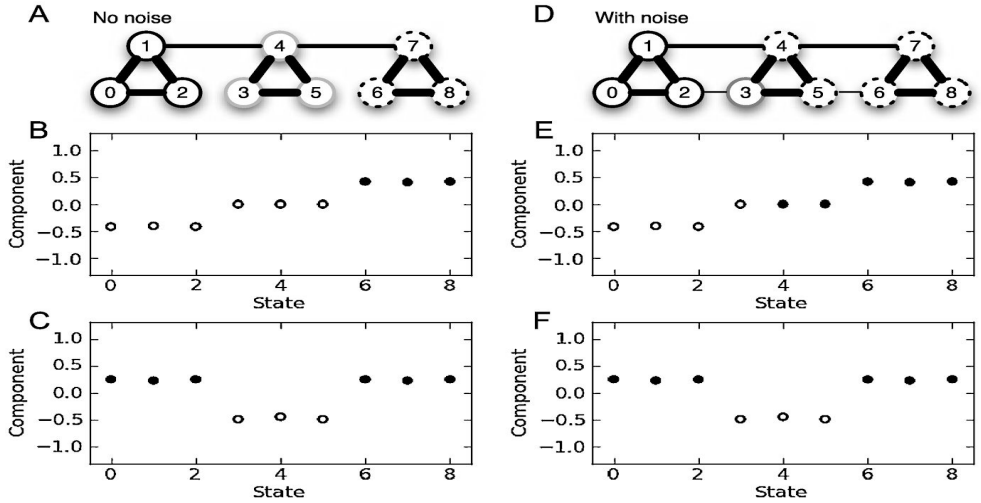

Figura 5. Dos modelos simples demuestran el poder de las limitaciones del PCCA. (A) Un modelo simple con transiciones bien conformadas. Cada uno de los nueve microestados tiene 1000 auto-transiciones. Cada línea gruesa corresponde a 100 transiciones y las líneas medias corresponden a 10 transiciones. Este modelo puede ser granulado en 3 macroestados que consisten de microestados 0-2 (negros), 3-5(grises) y 6-8(punteados). Las gráficas (B) y (C) muestran el segundo y tercer eigenvector de un modelo simple, respectivamente. Los círculos abiertos son usados por los componentes del eigenvector que son menores o iguales a cero y los círculos sombreados son usados para componentes mayores a cero. (D) El mismo modelo con dos transiciones débiles (ruido) entre los estados 2-3 y estados 5-6. Estas transiciones sólo tienen un conteo. Su presencia apenas influye en el modelo de los eigenvectores, pero alteran la estructura del segundo eigenvector (E) y de esta manera PCCA encuentra un error al construir los 3 macroestados, microestados 0-2 (negros), 3(gris) y 4-8(punteados). Las gráficas (B) y (C) muestran el segundo y tercer eigenvector del modelo simple desde la gráfica (D), respectivamente.

La efectividad de los métodos basados en MSMs radica en escoger una variedad de conformaciones iniciales para las simulaciones que después se usarán para construir un MSM. Cuando se escogen los macroestados se debe tener el cuidado de que estos estén conectados. Las simulaciones donde los macroestados distan uno del otro, nunca coincidirán con el resto de las trayectorias, haciendo imposible determinar intervalos de transición entre estos.

3. PARTE EXPERIMENTAL

Se utilizó el programa FAHClient, el cual tiene disponible su descarga gratuita en la página web del proyecto Folding@home[1], este puede ejecutarse en los sistemas Operativos Linux, Windows, Mac OS y Android. Es un sistema de cómputo distribuido que aprovecha la capacidad de cómputo cuando las computadoras, teléfonos se encuentran en estado inactivo; el programa se puede configurar para que en

los momentos de inactividad o con una carga ligera de trabajo de los dispositivos, se comunique con el servidor central de Folding@home y empiece a procesar los datos que le envían. El programa contiene un módulo de control donde se puede observar el progreso del procesamiento asi como sus bitácoras que va generando cuando está en ejecución (Figura 6).

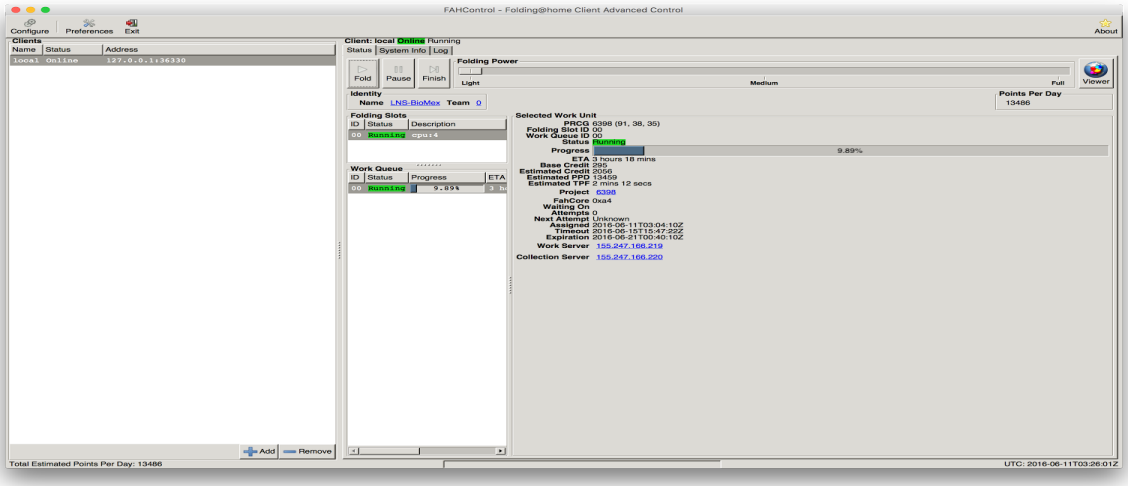

Figura 6. Panel de Control del programa FAHClient

El programa FAHClient utiliza un software llamado MSMBuilder[6], disponible de forma gratuita en el repositorio de software SimTk[7]. Por medio de este software se pueden construir MSMs que el programa gromacs[8] utiliza como parámetros para realizar el procesamiento de MD y realizar el análisis de proteínas. El programa MSMBuilder está desarrollado en lenguaje python para el fácil entendimiento de sus rutinas, con el fin de implementar nuevos métodos que surjan para la construcción de MSMs.

El programa MSMBuilder tiene dos componentes principales: las herramientas en lenguaje python y el clusterer que tiene integrado el software gromacs adaptado para ejecutar una MSM. El clusterer permite conformaciones de un gran número de microestados con base en la similaridad de su estructura. (Figura 7).

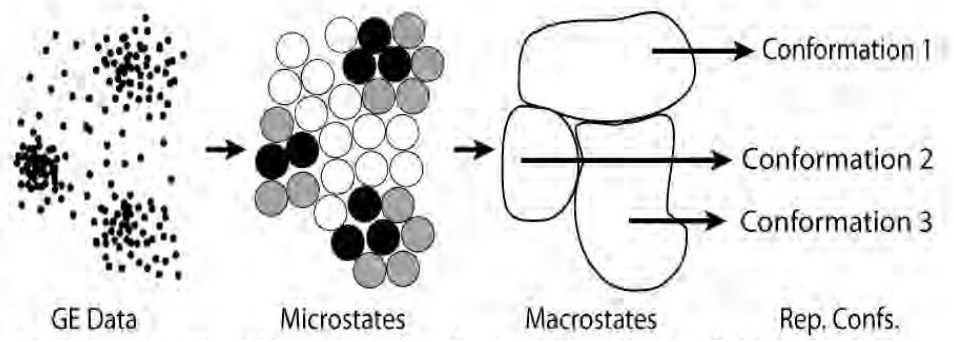

Figura 7. Esquema de los pasos requeridos para construir un MSM y obtener una conformación representativa de cada estado. Primero los datos son representados por puntos que posteriormente son agrupados por microestados representados por círculos, donde los sombreados son los más poblados en puntos. Los microestados relacionados cinéticamente son agrupados en macroestados o estados metaestables representados por las formas amorfas. Al final las conformaciones se obtienen identificando el macroestado con mayor probabilidad.

Se utilizó un conjunto de datos ejemplo de un modelo de potenciales en 2D, al ejecutar la prueba del software MSMBuilder. Para construir el MSM se utiliza el script BuildMSMsAsVaryLagTime.py, que en sus parámetros se definen de la siguiente manera:

-i Es el intervalo entre los tiempos de retraso para construir los modelos.
-m Es el tiempo máximo de retraso
-s Número de microestados
-t Tiempo en ps entre cada entrada
-l Tiempo de retraso en el número de entradas

Se utilizaron los parametros: l= 1, m= 4, n=0, s =25, t =1, obteniendo con esto la Figura 8, donde cada color representa un macroestado. A partir de la teoría sabemos que debido a nuestros datos iniciales debemos obtener 4 macroestados, donde uno de ellos solo corresponderá al punto inicial (0,0). De esto podemos deducir que los parametros empleados no fueron lo suficientemente buenos.

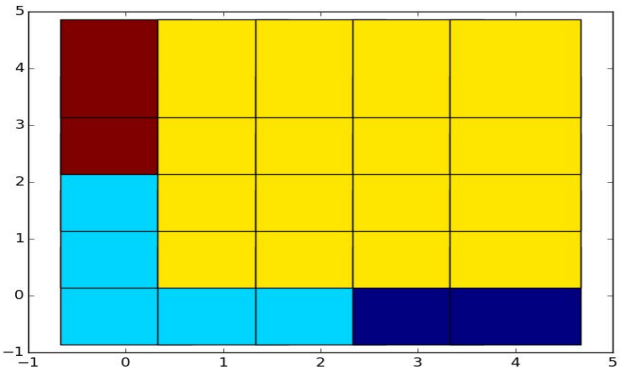

Figura 8. Muestra el mapeo de microestados a macroestados sin indicarle al programa el número de pasos a realizar y sin un refinamiento de los resultados.

Después de la prueba anterior, se restablecen los parámetros de la siguiente forma: l=1,m=4, n=3, i=10000, s=25, t=1; indicándole así al programa que realice 3 simulaciones extras para refinar los resultados (n=3) y que realice 10000 pasos (i=10000) en cada refinamiento. Con esto se obtiene la siguiente gráfica:

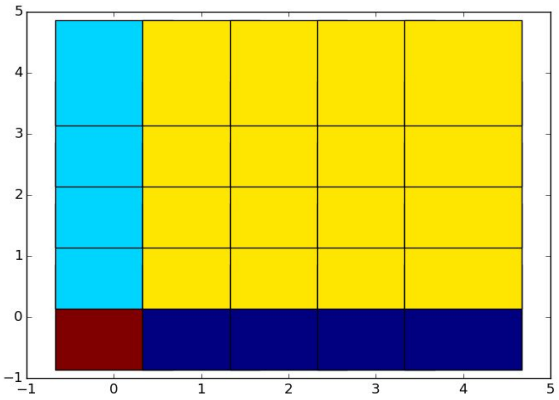

Figura 9. Gráfica reparametrizada. Muestra el mapeo de los microestados en 4 macroestados, donde uno de ellos solo representa solo la coordenada (0,0).

4. CONCLUSIONES

En el presente trabajo se desarrolla un análisis exhaustivo de los métodos utilizados por el programa Folding@Home, en el que actualmente colabora el Laboratorio Nacional de Supercómputo del Sureste de México. A través de este análisis se logran dos objetivos principales: la divulgación de este tipo de software que aporta en gran medida a las investigaciones de frontera realizadas en el área de la salud y la comprensión de su funcionamiento para posteriormente aportar al desarrollo de estos.

Las expectativas a futuro de este proyecto es que seamos capaces de realizar el modelado computacional de una arquitectura funcional celular y parametrizar las tasas de transición hacia los estados resultantes. Para esto se buscará construir un Modelo de Estado de Markov y definir una red dinámica a partir de un grupo de ensambles funcionales. Empleando la teoría de perturbaciones aleatorias, se estudiará la respuesta de las propiedades en el plegamiento de proteínas; en particular se busca presentar un protocolo que permita construir MSMs en equilibrio basados en trayectorias muestreadas a partir de superficies de energía potencial arbitrarias para poderlo aplicar a distintos sistemas de gran interés biológico. Haciendo asi, al LNS partícipe en investigaciones de gran impacto para las áreas de ciencias de la salud y computación.

BIBLIOGRAFIA

1. Folding@home. Página Web http://folding.stanford.edu/home

2. Monica Hollstein, D. Sidransky, B. Vogelstein C. C. Harris (1991), p53 Mutations in Human Cancers, Science Vol. 253, 49-53.

3. Peter Blume-Jensen, T. Hunter (2001), Oncogenic kinase signalling, Nature Vol. 411, 355-365

4. Diwakar Shukla, Y. Meng, B. Roux, V. S. Pande (2014), Activation pathway of Src kinase reveals intermediate states as targets for drug design. Nature Communications No. 4397, 1-11

5. Gregory R. Bowman et al. (2014), An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation, Advances in Experimental Medicine and Biology 797, 7-22.

6. GR Bowman, X Huang, and VS Pande. Methods 2009. “Using generalized ensemble simulations and Markov state models to identify conformational states”

7. SimTk MSMBuilder. Página Web https://simtk.org/projects/msmbuilder

8. Gromacs. Página Web http://www.gromacs.org