

RESEARCH

Un algoritmo paralelo para la reducción de trayectorias de plegamiento usando una estrategia rápida de agrupamiento

Luis Garreta^{1†}, Mauricio Martínez² and Pedro A Moreno^{1*}

Resumen

Background: La simulación del proceso de plegamiento de proteínas es una de las principales herramientas para estudiar y comprender los mecanismos subyacentes en este proceso. Hoy en día estas simulaciones están llegando a unos tiempos de simulación que hasta hace algunos años eran imposibles de alcanzar y como consecuencia las trayectorias generadas son muy grandes. Analizar este tipo de trayectorias trae complicaciones debido a su tamaño y por lo tanto se necesita crear herramientas que logren reducirlas de tal manera que se logren preservar tanto los eventos principales como el orden temporal en el que ellos ocurren.

Results: Introducimos aquí un algoritmo de reducción para trayectorias grandes de plegamiento de proteínas que se caracteriza por dividir la trayectoria en segmentos y mediante una estrategia rápida de agrupamiento tomar los eventos más disimilares para luego seleccionar entre ellos a los k eventos más representativos. El algoritmo aprovecha el orden temporal implícito en la trayectoria para realizar en cada segmento comparaciones locales, entre eventos vecinos, y así evitar realizar una comparación de todos contra todos que es muy costosa computacionalmente.

Conclusions: El esquema anterior permite que el algoritmo sea muy rápido y que los eventos seleccionados conserven su orden temporal dentro de la trayectoria. Además, el particionamiento en segmentos permite al algoritmo realizar la reducción por cada segmento de forma independiente y por lo tanto realizarse las reducciones de forma paralela lo que lo vuelve aún más rápido cuando se ejecuta en máquinas con procesadores de múltiples cores, como los PCs que se consiguen en el mercado hoy en día. Para mostrar la efectividad del algoritmo propuesto realizamos reducciones sobre tres conjuntos de trayectorias disponibles públicamente: las del supercomputador Anton, las del proyecto folding@home, y las del servidor de despliegamiento de Parasol.

Keywords: Protein folding simulations; Protein structure comparison; Protein structure clustering

Background

En este artículo presentamos un algoritmo para reducir trayectorias de plegamiento de proteínas el cual obtiene rápidamente conformaciones representativas conservando tanto su estructura tridimensional (3D)

como su orden temporal, y que además es altamente paralelizable. Las proteínas desempeñan funciones fundamentales en todos los seres vivos, pero para ser funcionales deben a partir de su cadena de aminoácidos (AA) plegarse hasta alcanzar una forma 3D única o estado nativo, lo que se conoce como el proceso de plegamiento de las proteínas. Entender los mecanismos y reglas de este proceso ha sido uno de los objetivos más perseguidos dentro de la biología y una herramienta teórica importante para estudiarlo son las trayectorias de plegamiento, que describen la evolución del plegamiento de una proteína mediante la secuen-

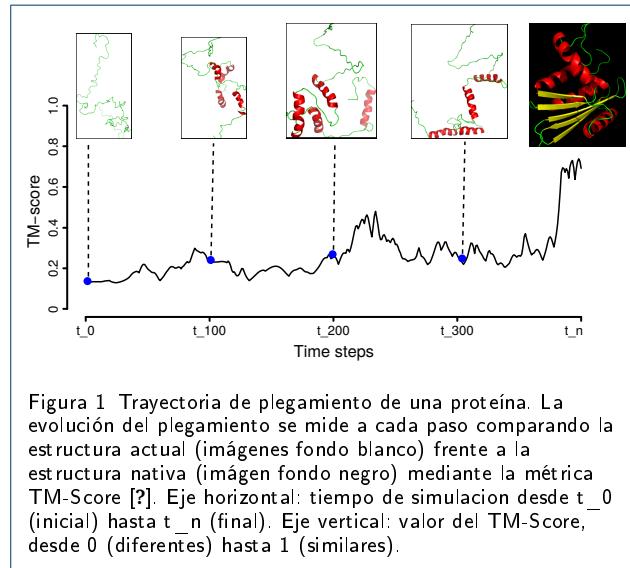
*Correspondence: pedro.moreno@correounivalle.edu.co

¹Escuela de Ingeniería de Sistemas y Computación, Universidad del Valle, Santiago de Cali, Colombia

Full list of author information is available at the end of the article

†Equal contributor

cia de estados que esta atraviesa en función del tiempo durante su proceso de plegamiento (Figura 1).



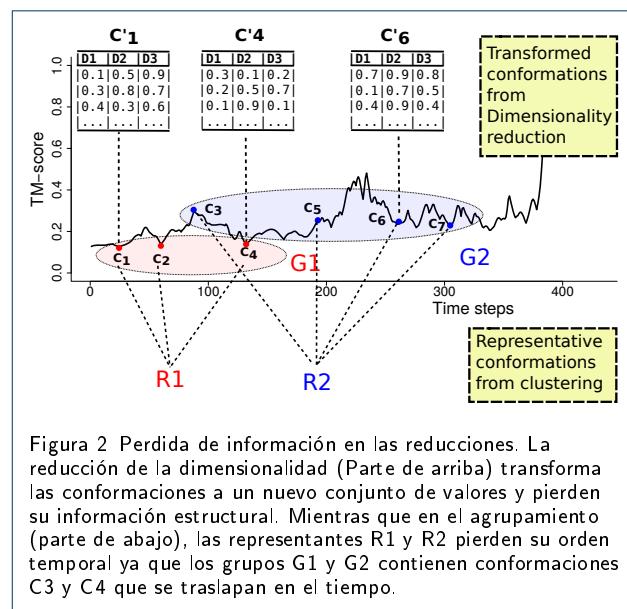
Estas trayectorias son simuladas principalmente por el método de dinámica molecular (DM), el cual por su costo computacional está limitado a proteínas pequeñas (< 100 AA) y a tiempos muy cortos (pico o microsegundos). Sin embargo, nuevos avances tecnológicos evidencian un progreso notable en estas simulaciones. Recientemente en el 2016 se puso en operación la supercomputadora Anton-2 [1], diez veces más rápida que su predecesora Anton-1 [2], diseñada especialmente para el plegamiento de proteínas y de la cual ya se reportó en el 2011 las simulaciones completas de 12 proteínas [3], varias en el orden de los milisegundos. Como una alternativa más económica a estas supercomputadoras, en el 2014 se usó unidades de procesamiento gráfico (GPUs) y se reportó las simulaciones de 17 proteínas en el orden de los microsegundos [4]. Y años antes, en el 2007 utilizando computadoras de escritorio unidas a través de computación distribuida en el proyecto folding@home se realizaron varias simulaciones en el orden de los microsegundos del plegamiento de la proteína villin headpiece [5].

Estos avances muestran un crecimiento notable en estas simulaciones con tiempos en el orden de los micro y milisegundos, y con trayectorias de millones de conformaciones. Muchas de estas trayectorias ya se están colocando a disposición pública, pero debido al gran número de conformaciones, su procesamiento y análisis en computadoras convencionales es muy costoso en tiempo computacional. Por lo tanto se necesitan nuevos algoritmos capaces de reducir estas trayectorias de una forma rápida, aprovechando eficientemente los recursos de este tipo de máquinas, y buscando

conservar la mayor información posible tanto a nivel de representación como a nivel de orden temporal.

Para realizar estas reducciones se han usado dos enfoques: la reducción de la dimensionalidad [6] y el agrupamiento [7]. En el primer enfoque se transforma una conformación a un conjunto reducido de variables que la representan lo mejor posible. Para esto se han usado tanto técnicas lineales como no-lineales (e.g. análisis de componentes principales (PCA), escalamiento multi-dimensional [8], Isomap [9], diffusion maps [10]). Sin embargo, aunque se logra la reducción de las conformaciones, se pierde su representatividad como estructuras 3D (Figura 2.A). Además, estas técnicas consumen mucho tiempo cuando las trayectorias son muy grandes, ya que tienen que transformar todas sus conformaciones.

En el segundo enfoque, agrupamiento, se asignan las conformaciones a grupos que comparten las mismas características (e.g. similaridad con la estructura nativa) y se toma de cada grupo ya sea un representante promedio ó sus características generales. Aquí se han usado tanto agrupamientos particionales como jerárquicos (e.g. k-means [11], linkage [12]). Sin embargo, los grupos pierden su orden temporal ya que pueden abarcar conformaciones que ocurren a tiempos muy distintos (Figura 2.B). Además, se tienen que comparar todos los pares de conformaciones, lo cual es una operación costosa y más aún cuando las trayectorias son muy grandes.



Nuestro algoritmo reduce una trayectoria de plegamiento en tres fases: primero divide la trayectoria en pequeñas subtrayectorias que luego las reduce de manera individual, independiente y paralela (Figura 3).

Segundo toma cada subtrayectoria y extrae de forma muy rápida sus conformaciones características y elimina las redundantes utilizando la estrategia de agrupamiento rápido de Hobohm and Sander (1992). Y tercero toma la conformaciones características y selecciona las más representativas mediante una estrategia tipo k-medoides [13], la cual al trabajar sobre pocas conformaciones, mejora sustancialmente su desempeño. Al final, los resultados de cada reducción se unen para obtener la reducción total de la trayectoria.

Además, a diferencia de otras técnicas de reducción de trayectorias, nuestro algoritmo tiene la ventaja de no cambiar la representación de las conformaciones como lo hacen las técnicas de reducción de dimensionalidad, ni de perder el orden temporal como lo hacen las técnicas de agrupamiento. El resultado de nuestro algoritmo es un conjunto de conformaciones representativas de la trayectoria que siguen conservando tanto su estructura 3D como su orden temporal.

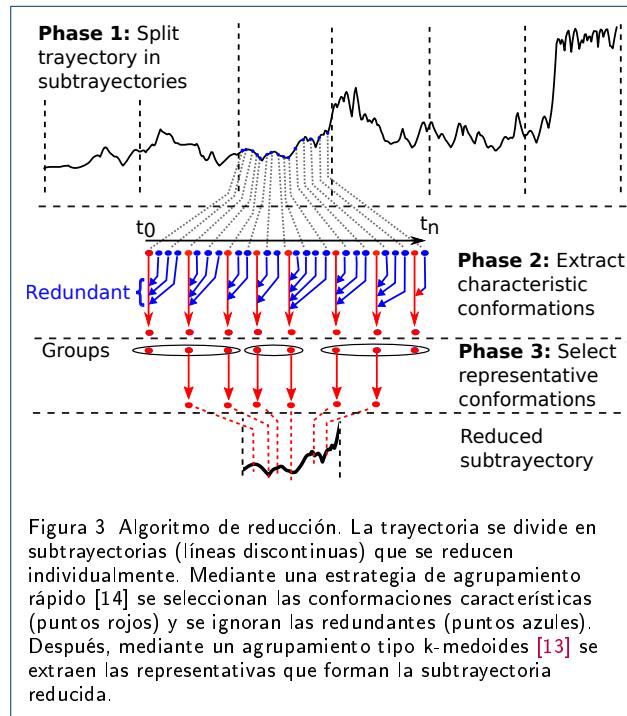


Figura 3 Algoritmo de reducción. La trayectoria se divide en subtrayectorias (líneas discontinuas) que se reducen individualmente. Mediante una estrategia de agrupamiento rápido [14] se seleccionan las conformaciones características (puntos rojos) y se ignoran las redundantes (puntos azules). Despues, mediante un agrupamiento tipo k-medoides [13] se extraen las representativas que forman la subtrayectoria reducida.

La implementación del algoritmo está en lenguaje R excepto la comparación entre conformaciones, que usa la métrica TM-Score para comparar pares de estructuras de proteínas [15], y que por ser la parte que más se repite está implementada en lenguaje Fortran.

Implementación

El algoritmo reduce una trayectoria de plegamiento de proteínas en tres fases: particionamiento, selección, y

extracción (Figura 3). Cada fase conlleva una estrategia para mejorar la eficiencia del algoritmo cuando las trayectorias de plegamiento son muy grandes.

Fase 1: Particionamiento y Paralelización

Dividimos la trayectoria en subtrayectorias con el objetivo de reducirlas de forma independiente y paralela (líneas verticales punteadas, Figura 3).

Esta estrategia de particionamiento tiene un doble objetivo: primero, reducir localmente cada subtrayectoria y así enfocarnos en sus características particulares, lo que al final resulta en la obtención de las características globales de toda la trayectoria. Y segundo, que sus reducciones se puedan realizar en paralelo y así mejorar notoriamente la eficiencia del algoritmo a la hora de ejecutarlo en una máquina con tecnología multi-core (Figura 4).

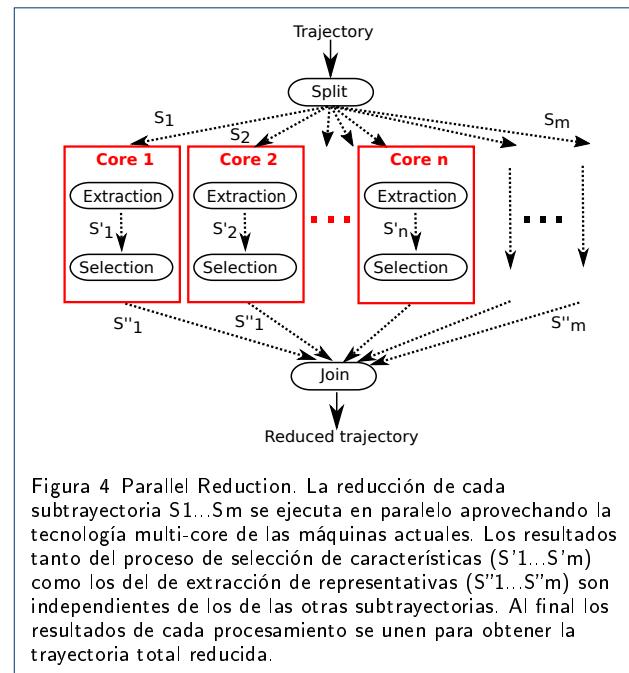


Figura 4 Parallel Reduction. La reducción de cada subtrayectoria $S_1 \dots S_m$ se ejecuta en paralelo aprovechando la tecnología multi-core de las máquinas actuales. Los resultados tanto del proceso de selección de características ($S'^1 \dots S'^m$) como los del de extracción de representativas ($S'^1 \dots S'^m$) son independientes de los de las otras subtrayectorias. Al final los resultados de cada procesamiento se unen para obtener la trayectoria total reducida.

Fase 2: Extracción y Filtración

Esta fase del algoritmo extrae rápidamente de cada subtrayectoria las conformaciones características y filtra las redundantes. Para hacerlo de manera eficiente, modificamos la estrategia de agrupamiento rápido de Hobohm and Sander (1992) para trabajar con estructuras de proteínas y en vez agruparlas busque las más disimilares.

El algoritmo aprovecha el orden temporal implícito en las subtrayectorias para organizar las conformaciones

en orden creciente de tiempo de simulación (flecha horizontal negra, Figura 3). Se asigna la primera conformación como la primera representante característica (punto rojo en t₀, Figura 3), y se toma la siguiente conformación y se comparan. Si son diferentes, entonces se convierte en una nueva representante (puntos rojos, Figura 3), de lo contrario es redundante y se filtra (puntos azules, Figura 3). Después, se toma la siguiente conformación y se continua el mismo proceso hasta terminar con todas.

Fase 3: Búsqueda y Selección

Esta última fase del algoritmo toma como entrada las conformaciones características de la fase anterior y realiza una búsqueda completa de las conformaciones que más las representen. Hablamos de completa porque la búsqueda implica comparar todas las conformaciones entre sí, es decir calcular su matriz de disimilaridades. Por esta razón es que esta búsqueda es factible hacerla ahora y no antes ya que se hace sobre un conjunto mucho menor de conformaciones que el que se tiene al inicio por cada subtrayectoria, .

Para encontrar estas conformaciones representantes calculamos las k conformaciones cuya disimilaridad media a todas las demás integrantes del grupo es mínima, lo que se conoce como *medoides* y el algoritmo que usamos para realizar esto es el particionamiento alrededor de medoides PAM [13].

Esta fase necesita tres datos de entrada: el conjunto de conformaciones características de la fase anterior (C), el umbral mínimo de TM-score para aceptar dos conformaciones como similares (T), y el número deseado de representantes seleccionadas (K).

Comparación entre Estructuras de Proteínas

Para la comparación de las estructuras de las conformaciones utilizamos la métrica TM-score (Template Modeling score) [15]. El TM-score es más preciso que otras métricas usadas en comparación de estructuras, como el Root Mean Square-Deviation (RMSD), ya que es más robusta a variaciones locales.

Nuestro algoritmo requiere como uno de sus parámetro de entrada un valor de puntaje mínimo de TM-score para aceptar como similares a dos conformaciones. Este parámetro se usa después tanto en la fase de extracción de características, al comparar las conformaciones para encontrar disimilares y remover redundantes, como en la fase de selección de representativas, al calcular la matriz de distancias de todas las conformaciones.

Para tener una aproximación del rango de valores de este puntaje mínimo, los puntajes del TM-score varían de 0 a 1, donde 1 indica un emparejamiento perfecto. Además, las estadísticas hechas por sus autores [16] muestran que un puntaje < 0.17 indica dos estructuras aleatorias, sin relación de similaridad, y un puntaje > 0.5 indica que las estructuras tienen un grado de similaridad que no está dado por el azar.

Resultados y Discusión

Conjuntos de datos de trayectorias de plegamiento

Ahora mostramos las reducciones realizadas por nuestro algoritmo a tres trayectorias de tres distintos proyectos de simulación de proteínas (Figura ??): la trayectoria de la proteína Trp-cage (Figura ??A), simulada con dinámica molecular en la supercomputadora Anton por D.E Shaw Research [3]; la trayectoria de la proteína villin-headpiece (Figura ??B), simulada con dinámica molecular utilizando computación distribuida en el proyecto folding@home [17]; y la trayectoria de la proteína Ribonuclease H (Figura ??C), simulada con el método Probabilistic Roadmap Method [18] que realiza el *desplegamiento* de proteínas a partir de su estado nativo.

Evaluación frente a otros métodos de reducción

Competing interests

The authors declare that they have no competing interests.

Author's contributions

Text for this section ...

Acknowledgements

Text for this section ...

Author details

¹ Escuela de Ingeniería de Sistemas y Computación, Universidad del Valle, Santiago de Cali, Colombia. ² The European Bioinformatics Institute (EMBL-EBI), Hinxton, Cambridgeshire, UK.

Referencias

- Shaw, D.E., Grossman, J.P., Bank, J.A., Batson, B., Butts, J.A., Chao, J.C., Deneroff, M.M., Dror, R.O., Even, A., Fenton, C.H., Forte, A., Gagliardo, J., Gill, G., Greskamp, B., Ho, C.R., Ierardi, D.J., Iserovich, L., Kuskin, J.S., Larson, R.H., Layman, T., Lee, L.-S., Lerer, A.K., Li, C., Killebrew, D., Mackenzie, K.M., Mok, S.Y.-H., Moraes, M.A., Mueller, R., Nociolo, L.J., Peticolas, J.L., Quan, T., Ramot, D., Salmon, J.K., Scarpazza, D.P., Schafer, U.B., Siddique, N., Snyder, C.W., Spengler, J., Tang, P.T.P., Theobald, M., Toma, H., Towles, B., Vitale, B., Wang, S.C., Young, C.: Anton 2: Raising the Bar for Performance and Programmability in a

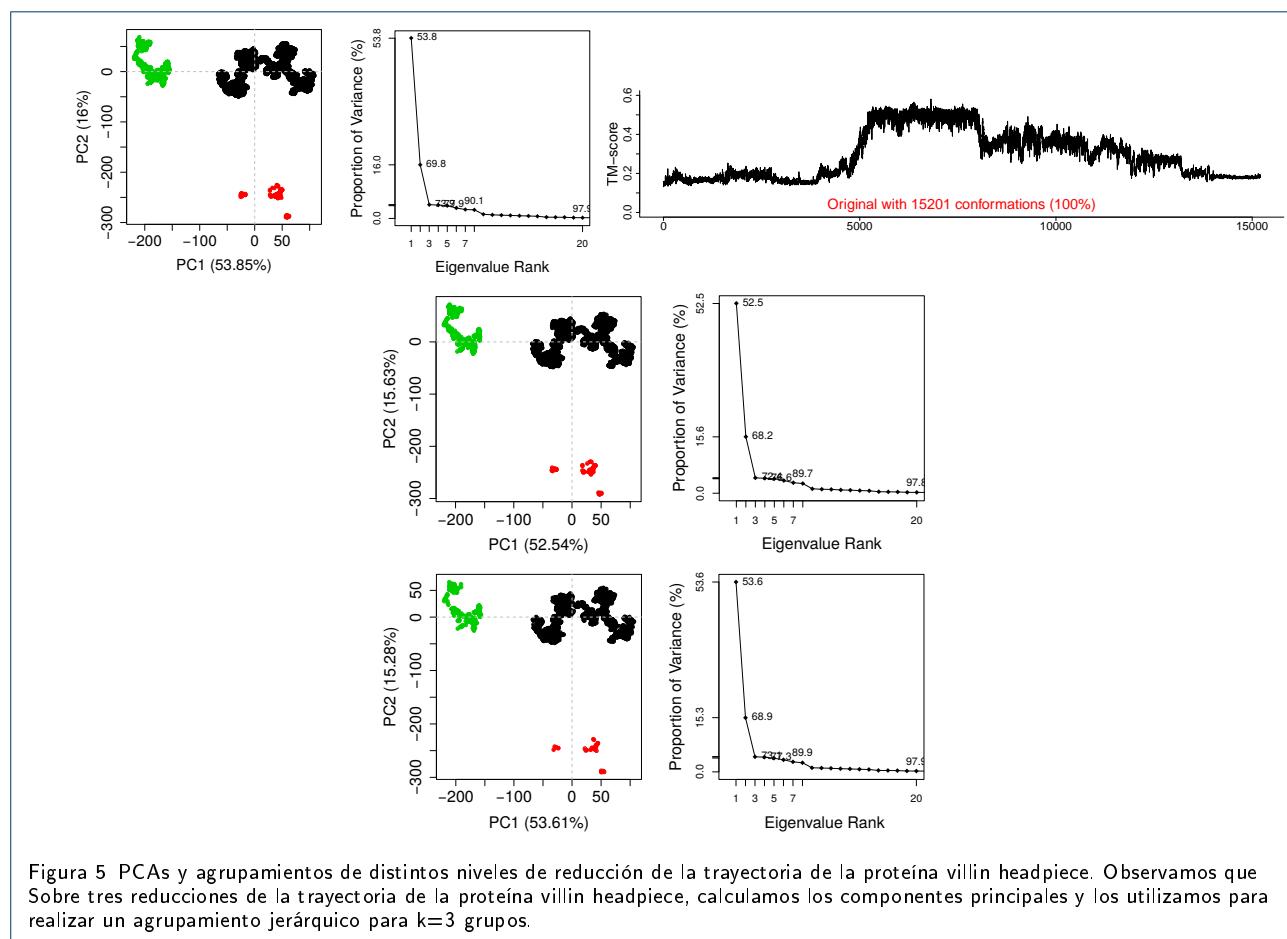


Figura 5 PCAs y agrupamientos de distintos niveles de reducción de la trayectoria de la proteína villin headpiece. Observamos que sobre tres reducciones de la trayectoria de la proteína villin headpiece, calculamos los componentes principales y los utilizamos para realizar un agrupamiento jerárquico para $k=3$ grupos.

- Special-Purpose Molecular Dynamics Supercomputer. In: Shaw2014 (ed.) SC14: International Conference for High Performance Computing, Networking, Storage and Analysis, pp. 41–53. IEEE, Los Alamitos, CA, USA (2014). doi:10.1109/SC.2014.9. <http://ieeexplore.ieee.org/document/7012191/>
2. Shaw, D.E., Chao, J.C., Eastwood, M.P., Gagliardo, J., Grossman, J.P., Ho, C.R., Lerardi, D.J., Kolossváry, I., Klepeis, J.L., Layman, T., McLeavey, C., Deneroff, M.M., Moraes, M.A., Mueller, R., Priest, E.C., Shan, Y., Spengler, J., Theobald, M., Towles, B., Wang, S.C., Dror, R.O., Kuskin, J.S., Larson, R.H., Salmon, J.K., Young, C., Batson, B., Bowers, K.J.: Anton, a special-purpose machine for molecular dynamics simulation. *Communications of the ACM* 51(7), 91 (2008). doi:10.1145/1364782.1364802
 3. Lindorff-Larsen, K., Piana, S., Dror, R.O., Shaw, D.E.: How fast-folding proteins fold. *Science* 334(6055), 517–520 (2011). doi:10.1126/science.1208351. arXiv:1011.1669v3
 4. Nguyen, H., Maier, J., Huang, H., Perrone, V., Simmerling, C.: 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 (2014). doi:10.1021/ja5032776
 5. Larson, S.M., Snow, C.D., Shirts, M., Pande, V.S.: Folding@Home and Genome@Home: Using distributed computing to tackle previously intractable problems in computational biology (2009). 0901.0866
 6. Duan, M., Fan, J., Li, M., Han, L., Huo, S.: Evaluation of Dimensionality-reduction Methods from Peptide Folding-unfolding Simulations. *Journal of chemical theory and computation* 9(5), 2490–2497 (2013). doi:10.1021/ct400052y
 7. Peng, J.-h., Wang, W., Yu, Y.-q., Gu, H.-l., Huang, X.: Clustering algorithms to analyze molecular dynamics simulation trajectories for complex chemical and biological systems. *Chinese Journal of Chemical Physics* 31(4), 404–420 (2018). doi:10.1063/1674-0068/31/cjcp1806147
 8. Rajan, A., Freddolino, P.L., Schulten, K.: Going beyond clustering in MD trajectory analysis: An application to villin headpiece folding. *PLoS ONE* 5(4), 9890 (2010). doi:10.1371/journal.pone.0009890
 9. Das, P., Moll, M., Stamati, H.: Low-dimensional, free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction. *Proceedings of the ...* 103(26) (2006)
 10. Kim, S.B., Dsilva, C.J., Kevrekidis, I.G., Debenedetti, P.G.: Systematic characterization of protein folding pathways using diffusion maps: Application to Trp-cage miniprotein. *The Journal of Chemical Physics* 142(8), 85101 (2015). doi:10.1063/1.4913322
 11. Doerr, S., Ariz-Extreme, I., Harvey, M.J., De Fabritiis, G.: Dimensionality reduction methods for molecular simulations (2017). 1710.10629
 12. Shao, J., Tanner, S.W., Thompson, N., Cheatham, T.E.: Clustering Molecular Dynamics Trajectories: 1. Characterizing the Performance of Different Clustering Algorithms. *Journal of chemical theory and computation* 3(6), 2312–34 (2007). doi:10.1021/ct700119m
 13. Kaufman, L., Rousseeuw, P.: *Finding Groups in Data*. Wiley-Interscience; New York, ??? (1990)
 14. Hobohm, U., Scharf, M., Schneider, R., Sander, C.: Selection of representative protein data sets. *Protein Science* 1(3), 409–417 (1992). doi:10.1002/pro.5560010313
 15. Zhang, Y., Skolnick, J.: Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics* 68(4), 1020 (2007). doi:10.1002/prot.21643
 16. Xu, J., Zhang, Y.: How significant is a protein structure similarity

- with TM-score = 0.5? *Bioinformatics* 26(7), 889–895 (2010). doi:10.1093/bioinformatics/btq066
- 17. Ensign, D.L., Kasson, P.M., Pande, V.S.: 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)
 - 18. Amato, N.M., Tapia, L., Thomas, S.: A Motion Planning Approach to Studying Molecular Motions. *Communications in Information and Systems* 10(1), 53–68 (2010). doi:10.4310/CIS.2010.v10.n1.a4

Figures

Figura 6 Sample figure title. A short description of the figure content should go here.

Figura 7 Sample figure title. Figure legend text.

Tables

Cuadro 1 Sample table title. This is where the description of the table should go.

| | B1 | B2 | B3 |
|----|-----|-----|-----|
| A1 | 0.1 | 0.2 | 0.3 |
| A2 | ... | ... | . |
| A3 | ... | . | . |

Additional Files

Additional file 1 — Sample additional file title

Additional file descriptions text (including details of how to view the file, if it is in a non-standard format or the file extension). This might refer to a multi-page table or a figure.

Additional file 2 --- Sample additional file title

Additional file descriptions text.