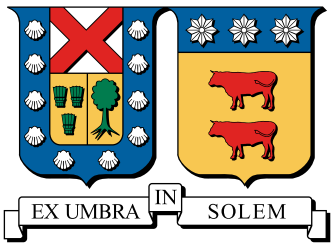
UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA

DEPARTAMENTO DE INFORMÁTICA

SANTIAGO - CHILE



EXPLORANDO ESTRATEGIAS DE BÚSQUEDA PARA EL PROBLEMA DE ACOPLAMIENTO DE PROTEÍNAS

RAIMUNDO GROSS LABBE

MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN INFORMÁTICA

PROFESOR GUÍA: NICOLAS GALVEZ RAMIREZ
PROFESOR CORREFERENTE: CARLOS CASTRO VALDEBENITO

NOVIEMBRE - 2023

AGRADECIMIENTOS

Quisiera agradecer en primera instancia a mi familia, por el gran apoyo que han significado a lo largo de este proceso, dándome ánimos, escuchando mis reflexiones, y siempre atentos a los avances que iba logrando. Agradecer a mi profesor guía, por siempre motivarme a dar lo mejor de mí, a explorar y salir de mi zona de confort, y por ayudarme a darle forma a este gran trabajo. A mis compañeros de memoria por esas tardes de conversaciones, de ideas y reflexiones sobre los trabajos de cada uno, ciertamente se hizo más llevadero vivir esto junto a ustedes. Y por último agradecer a aquellas personas que me entregaron una palabra de aliento, me acompañaron durante los tiempos de desarrollo y redacción, y que siempre confiaron en mí.

A mi hermano, para que nunca se rinda.

Resumen

Las proteínas son moléculas muy importantes para la vida debido a la gran cantidad de funciones que desarrollan dentro de las células, a través de interacciones con otras proteínas. El predecir estas interacciones permite un mejor entendimiento del funcionamiento celular, mejor desarrollo de medicinas, control de epidemias, entre otros. Debido a que aún hay muchas interacciones desconocidas, la predicción de acoplamiento de proteínas ha cobrado relevancia. Los principales trabajos desarrollados están basados en complementariedad geométrica y minimización de la energía libre del sistema. En esta memoria, se construyen dos algoritmos de búsqueda local basados en Hill Climbing para resolver el problema de acoplamiento con cuerpo rígido. Los resultados obtenidos indican que el algoritmo de búsqueda por ejes logra mejor detección de zonas de mayor contacto entre superficies y que una aproximación puramente geométrica no es suficiente para poder encontrar conformaciones cercanas a las nativas.

Abstract

Proteins are essential molecules due to the many functions in the cell through interactions with other proteins. Predicting these interactions leads to a better understanding of cell functioning, better drug design, epidemic control, and others. Because there are still many unknown interactions, protein docking prediction has gained relevance in recent years. The main works developed are based on geometric complementarity and system free-energy minimization. In this work, two local search Hill Climbing-based algorithms are presented to solve the docking problem with rigid body. Results suggest axle search algorithm is better than the sphere search algorithm in finding great contact zones between surfaces, and a purely geometric approximation is not enough to find near-native conformations.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

Las proteínas son moléculas fundamentales para nuestro día a día, ya que participan en funciones críticas, tales como procesamiento de información, manejo del sistema inmune o la regulación del metabolismo. Al realizar estas tareas, dentro y fuera de la célula, las proteínas se relacionan con muchas otras moléculas para poder desempeñar sus funciones, incluyendo a otras proteínas. A estas interacciones se les llama reconocimiento molecular, y corresponden al campo de estudio de *molecular docking*, cuyo objetivo es entender y predecir estas interacciones, tanto estructuralmente, para encontrar formas de acople, como energéticamente para predecir la afinidad entre ellas.[1]. En un principio las interacciones estudiadas eran entre proteína-ligando, siendo el ligando una molécula pequeña, pero durante las últimas décadas el estudio de las interacciones entre proteínas se ha ido extendiendo.

El estudiar cuáles son sus interacciones y cómo es que ocurren en nuestro cuerpo y en la naturaleza, junto con la estructura misma de las proteínas, permite alcanzar un mejor entendimiento de los procesos intrínsecos de nuestro cuerpo y del resto de los seres vivos, lo que tiene como consecuencia avances en medicina, farmacología, epidemiología, preservación de especies, entre otros.

1.1. El estudio de las proteínas

Las áreas de la ciencia encargadas de estudiar las proteínas corresponden a la biología molecular y la bioinformática que logrado un gran crecimiento en las últimas décadas. La bioinformática surge a partir de la necesidad de manejar la gran cantidad de datos de secuenciación generados por la biología molecular [2], y aporta con la integración de procesos informáticos para poder procesar de manera más eficiente los datos, ya que los procesos tradicionales son costosos en términos de recursos y tiempo.

Una definición formal de bioinformática es la siguiente: *"La bioinformática, en relación con la genética y la genómica, es una subdisciplina científica que implica el uso de ciencias informáticas para recopilar, almacenar y analizar y diseminar datos e información biológicos, como secuencias de ADN y aminoácidos o anotaciones sobre esas secuencias"* [3].

Dentro de la bioinformática, una de las áreas que estudia las proteínas es el modelado de estructuras proteicas, que comprende la predicción de la estructura de una proteína y la predicción del acoplamiento entre proteínas. La predicción de la estructura de una proteína consiste en poder predecir como se organizará espacialmente una proteína, dada su composición. Este desafío se enfrenta como un problema de optimización con el objetivo de encontrar una configuración estructural que minimice la energía libre del sistema. En cambio, la predicción de interacción de proteínas, busca determinar si una proteína interactúa con alguna otra molécula para formar un compuesto nuevo, basándose en complementariedad geométrica entre las superficies de ambas, contacto entre cadenas laterales de las proteínas, energía del solvente, etc. Esta predicción resulta un problema complejo debido la cantidad de variables involucradas y a que es necesario comparar diferentes conformaciones posibles para encontrar aquella que minimice la energía libre del sistema convirtiéndolo en un problema NP-Hard.

1.2. Objetivos y contribuciones

El objetivo de esta investigación es *aplicar estrategias de búsqueda de local para problemas complejos aplicadas al problema de interacción de proteínas, acotado a la interacción proteína - proteína, con el propósito de comparar algoritmos basados en complementariedad geométrica*. El objetivo general puede, además, desglozarse en los objetivos específicos:

- Modelar y resolver el problema de interacción de proteínas con técnicas de programación con restricciones, aplicado a distintas instancias.
- Aplicar una metaheurística de búsqueda local y las técnicas seleccionadas.
- Analizar y comparar los resultados obtenidos a partir de las técnicas.

Conjuntamente, las contribuciones de este trabajo radican en determinar y comparar la efectividad de diferentes estrategias de búsqueda aplicadas a la metaheurística Hill Climbing. Estas estrategias están basadas en complementariedad geométrica, para comprobar si se obtienen soluciones cercanas a la nativas, o en su defecto, si constituye un buen acercamiento para que otras técnicas de refinamiento puedan alcanzar dichas soluciones.

1.3. Organización del documento

La presente memoria está organizada de la siguiente forma:

- En el Capítulo 2, se define el concepto de proteína y *molecular docking*, caracterizando sus interacciones, la relevancia del campo y la dificultad del problema abordado.
- El Capítulo 3 presenta un estado del arte del problema de interacción proteína proteína y similares.
- El Capítulo 4 abarca el diseño de la solución, cómo se representan las proteínas, las formas de evaluación y los algoritmos utilizados.
- En el Capítulo 5 se presentan la configuración de los algoritmos utilizados y se discuten los resultados obtenidos.
- Finalmente, en el Capítulo 6 se concluye sobre el presente trabajo, incluyendo reflexiones respecto a los resultados obtenidos y opciones de trabajos futuros.

Capítulo 2

La proteína y sus interacciones

Una proteína es una biomolécula y macromolécula grande formada a partir de largas cadenas de aminoácidos conectados entre sí con enlaces polipéptidos covalentes. Dicha secuencia de aminoácidos está definida por una secuencia genética y es la que determina la estructura y su funcionalidad. De acuerdo a su funcionalidad se pueden agrupar en: anticuerpos, enzimas, mensajeras, estructurales y de transporte y almacenamiento. Para cumplir con todas estas funciones las proteínas interactúan con otras moléculas tales como ácidos nucleicos, lípidos u otras proteínas. Cuando una proteína interactúa con otra molécula cualquiera se le conoce como una interacción proteína-ligando o proteína-ligante. Cuando una proteína interactúa con otra se llama interacción proteína-proteína.

2.1. Molecular Docking

En nuestro cuerpo, constantemente están ocurriendo diversas interacciones entre moléculas: proteína-proteína, enzima-sustrato, proteína-ácido nucleico, etc. Estas interacciones son el objeto de estudio del acoplamiento molecular o molecular docking. Esta disciplina trata de predecir la formación de complejos receptor-ligando. Los receptores generalmente son proteínas o ácidos nucleicos y los ligandos moléculas pequeñas u otras proteínas (la Figura 2.1 incluye la representación de una proteína) . El proceso de acoplamiento trata de imitar, por métodos computacio-

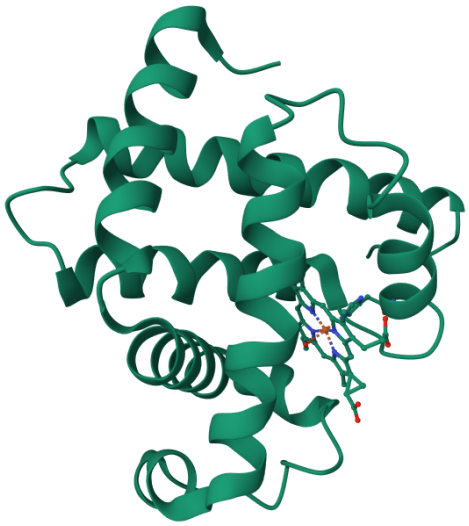


Figura 2.1: Representación 3D de la estructura de la mioglobina. Esta fue la primera proteína a la cual se le determinó la estructura mediante cristalografía de rayos X [4].

nales, cómo es que ocurre la unión entre las moléculas en la naturaleza, y predecir así la forma en la que enlazan y la afinidad misma del enlace, lo cual es de gran utilidad para el diseño de nuevas medicinas.

La predicción de interacciones receptores y ligandos puede ser categorizada en dos grupos de acuerdo a la flexibilidad con la que se consideren las moléculas, lo que influye en la dificultad [5]:

- **Acoplamiento rígido:** Considera al receptor y al ligando como cuerpos rígidos y que mantienen su forma durante el acople. Este acoplamiento es más rápido y con un espacio de búsqueda más pequeño.
- **Acoplamiento flexible**: El caso más común es considerar la flexibilidad del ligando y su espacio conformacional, y al receptor como una proteína rígida. También están los casos en que se trabaja con la flexibilidad de la proteína y que debido a su tamaño, su flexibilidad involucra mayor cantidad de grados de libertad, siendo el problema más complejo del acoplamiento molecular.

Todo método de acoplamiento está compuesto por un algoritmo de búsqueda y una

función de puntuación o *scoring*. El algoritmo de búsqueda se encarga de navegar el espacio y encontrar múltiples configuraciones de acople entre las moléculas. La función de puntuación es la que se encarga de predecir la fuerza de la interacción, también llamada afinidad de la interacción, a través de métodos matemáticos [6]. El resumen de los métodos utilizado se muestra en la Figura 2.2.

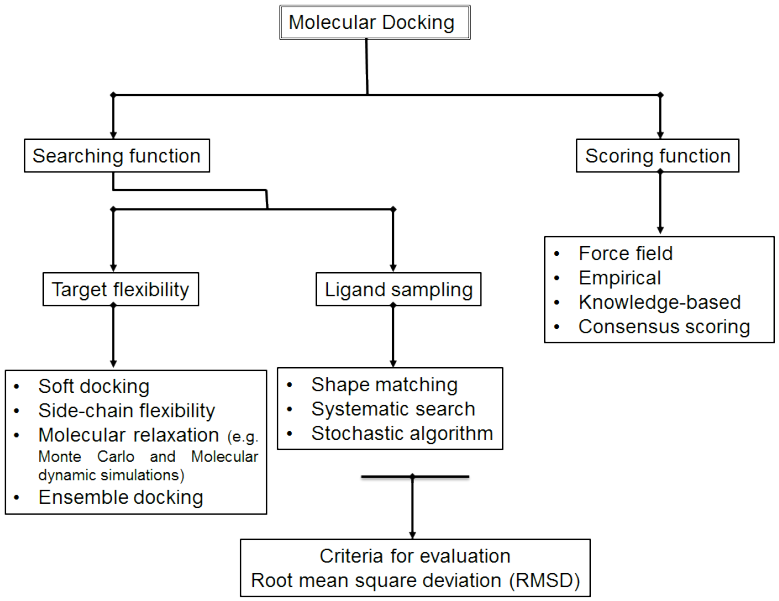


Figura 2.2: Esquema con los métodos usados para molecular docking [6].

2.2. Interacciones proteína-proteína

Una interacción proteína-proteína (o PPI, del ingles *Protein-protein interaction*) corresponde a un evento de asociación física, específica e intencional que ocurre bajo fuerzas biomoleculares y está condicionado por el tipo de célula, el estado de desarrollo de la misma, estímulos medioambientales u otros factores. Estas PPIs tienen un rol clave en funciones como control del ciclo celular, traducción, transcripción y replicación de ADN, transducción de señales, sentir el ambiente o convertir energía en movimiento [7; 8]. También, una proteína puede participar en más de una interacción dando lugar a múltiples procesos celulares interrelacionados, altamente organizados. Ese conjunto de interacciones es conocido como red de interacciones de proteínas y el entenderlas permite, por ejemplo, anticipar

posibles efectos adversos de ciertos tratamientos que tengan como objetivo una proteína en particular.

Las PPI pueden ser clasificadas de acuerdo a tres características principales:

- **Similitud de los complejos:** Clasifica las PPI dependiendo si sus cadenas son idénticas o no, en homoligoméricas o heteroligoméricas respectivamente.
- **Duración:** Separa en interacciones estables (o permanentes), en las que la interacción es fuerte y por lo general las proteínas involucradas sólo son vistas en forma de complejo; y las interacciones transitorias, que poseen un tiempo de vida más corto y que pueden ser fuertes o débiles dependiendo si requieren algún gatillo molecular para que ocurran. Además las interacciones débiles ocurren constantemente y pueden ser apreciadas *in vivo*.
- **Obligación:** La obligación de una interacción se relaciona a si los protómeros [1] del complejo formado son encontrados por su cuenta como estructuras estables *in vivo*.

2.3. Relevancia y aplicaciones de las PPI

Se estima que dentro de una célula humana ocurren más de 300.000 PPI para poder llevar a cabo sus funciones, y muchas de éstas están relacionadas entre sí. Los virus, bacterias, u otros agentes patógenos también tienen su propio conjunto de proteínas e interacciones, y al ingresar a nuestro organismo, dichas proteínas pueden interactuar con las nuestras. El poder detectar estas interacciones y entender cómo es que ocurren entrega herramientas para poder generar medidas para contrarrestarlas, siendo uno de los ejemplos más reveladores de lo importante que es conocer estas interacciones. Si bien se han hecho grandes esfuerzos para recopilar información en bases de datos [9] obtenidas a partir de los experimentos y aportes computacionales, siguen existiendo interacciones desconocidas, tanto entre proteínas del cuerpo humano, como entre proteínas humanas y de agentes externos. A continuación, se describen dos casos relevantes de aplicaciones de las PPI para el tratamiento del cáncer y del COVID-19.

[1] Protómero: Subunidad estructural de una proteína oligomérica

2.3.1. COVID-19

Dentro de las aplicaciones que tiene el estudio de las PPIs destaca la inmunología, con la finalidad de desarrollar componentes que logren inhibir, frenar o combatir el desarrollo de ciertas enfermedades. Entre las más recientes y destacables está el estudio de las PPI para desarrollar medicina para combatir el COVID-19. En [10] se señala que las enfermedades de tipo Coronavirus tienen en común una gran cantidad de proteínas que son importantes para el ciclo de vida viral, donde muchas de estas interactúan con las proteínas del huésped, convirtiendo dichas PPI en un blanco para nuevas drogas.

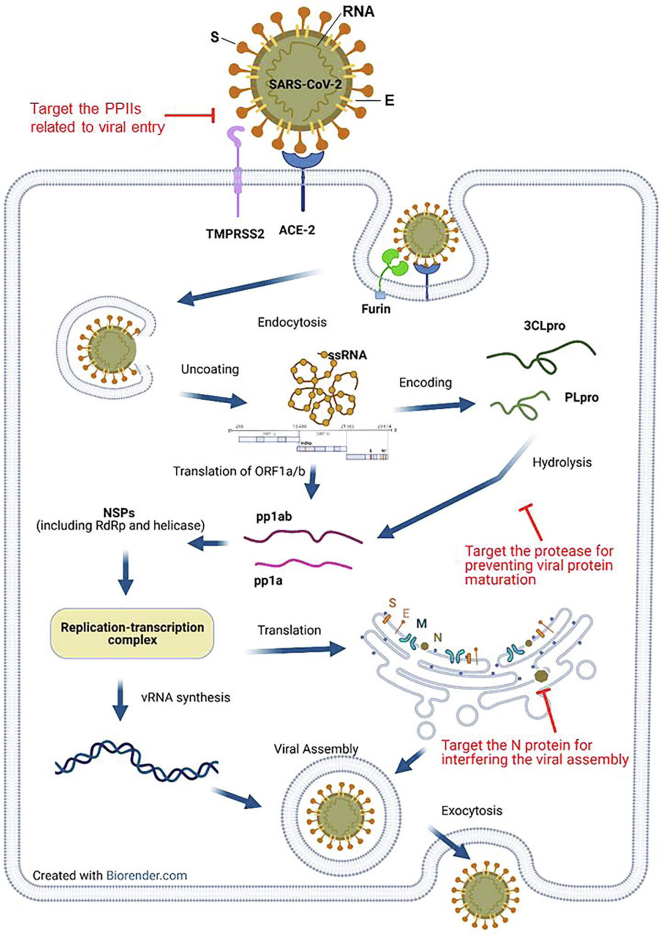


Figura 2.3: PPIs y proteínas involucradas en el ciclo de vida viral. Las interacciones apropiadas para inhibidores están destacadas en rojo [10].

La Figura 2.3 muestra una representación del ciclo de vida viral y las proteínas involucradas, en donde se pueden apreciar tres momentos en los cuales se puede intervenir: durante el proceso de entrada a través de la membrana, la maduración de proteínas y replicación, y el ensamblaje viral.

Se ha descubierto que la entrada viral a través de la membrana se logra gracias a una PPI entre un componente presente en el virus, la glicoporteína-S, con una enzima presente en la membrana celular, la enzima convertidora de angiotensina-2. Por ende, desarrollar inhibidores de focalicen dicha interacción evitaría la infección de la célula, y por consecuencia la replicación del virus, tal como se replica en [11; 12].

Por otro lado, existe otra interacción proteína-proteína presente en el proceso de replicación viral, que es propia del coronavirus. La proteína nucleocápside del coronavirus (abreviada como N) se une a RNA viral desempeñando un rol clave en proteger el genoma viral para asegurar una replicación oportuna y transmisión confiable, así como también en el ensamblado de las partículas virales [13].

Estas menciones son sólo algunas de las interacciones con las que se está trabajando actualmente y que se encuentran dentro al menos otras 332 interacciones proteína-proteína entre el SARS-CoV-2 y los humanos [14], cuya caracterización es insuficiente para diseñar nuevas drogas.

2.3.2. Tratamiento de cáncer

En el combate contra el cáncer, también se han identificado ciertas PPIs que están asociadas a la supresión de tumores y al desarrollo y propagación de células cancerígenas. Con respecto al primer caso se encuentra la interacción entre la proteína p53, una de las proteínas más estudiadas en el campo del cáncer debido a su función de apoptosis[2] (una de las muchas otras que desempeña), y la proteína MDM2, que se encarga de regular la acción de p53 y degradarla. El lograr interferir en la interacción MDM2/p53 contribuiría a una mejor acción antitumoral de parte de la proteína p53 y un tratamiento más eficiente [16].

Un ejemplo de esta interferencia se puede ver en la Figura 2.4 en donde un inhibidor nutlin se acopla a la proteína MDM2 en el punto de anclaje para la proteína

[2]Apoptosis: Tipo de muerte celular en la que una serie de procesos moleculares en la célula conducen a su muerte. Este es un método que el cuerpo usa para deshacerse de células innecesarias o anormales. El proceso de apoptosis puede estar bloqueado en las células cancerosas. También se llama muerte celular programada.[15]

p53, evitando así que ésta pueda interactuar con MDM2 y por ende, que se degrade, manteniendo su función antitumoral.

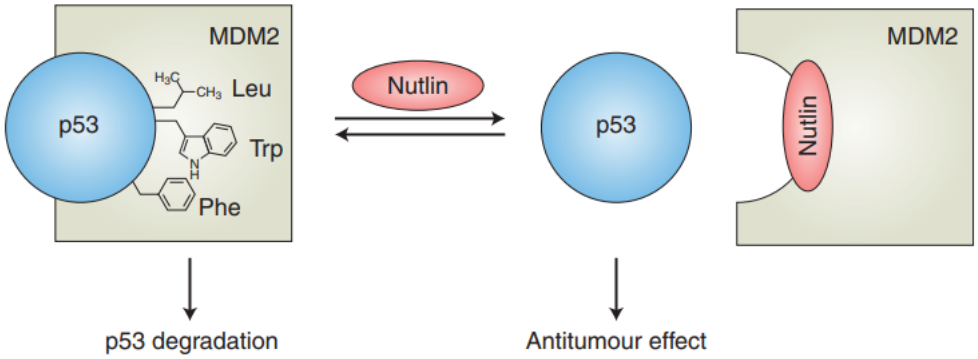


Figura 2.4: Diagrama conceptual de un inhibidor de PPI actuando sobre un punto de anclaje de la interacción p53-MDM2 [17].

Otro foco exitoso en el tratamiento contra el cáncer es la tubulina, presente en los microtúbulos celulares, éstos participan en varios procesos biológicos como en el desarrollo y mantención de la forma de la célula, comunicación y movimiento celular. No obstante, su participación en el proceso de reproducción y división celular es de vital importancia para terapias. Los objetivos de las drogas actuales que afectan las interacciones proteína-proteína de la tubulina se pueden separar en dos grupos: aquellas que estabilizan las PPIs y fortalecen los microtúbulos, evitando así la división celular, como ejemplo tenemos a paclitaxel; y aquellas que las desestabilizan, como la colchicina [18]. Ambas acciones conducen a la muerte de la célula.

Por último una proteína de gran importancia, es la proteína E6, esta proteína interactúa con varias otras dentro de la célula, dentro de ellas la p53 mencionada anteriormente, comprometiendo la capacidad de la célula de enviar señales para la apoptosis, y por ende facilitando su supervivencia [19]. Esta proteína está asociada al cáncer cervical, causado por el virus del papiloma humano (VPH). Se tiene registro de aproximadamente 600.000 nuevos casos cada año, de los cuales más de la mitad terminan en muerte [20]. Es por esto que encontrar formas de combatir este cáncer se vuelve imperativo, y el entender las interacciones asociadas a la E6 juega un rol crucial.

2.4. Dificultad de predecir una interacción

El predecir correctamente una interacción entre proteínas, implica encontrar el mejor punto de acoplamiento entre ambas que está dado por la configuración que minimice la energía libre del sistema, ya que así es como se encuentran de manera nativa en la naturaleza. Que un punto corresponda al mínimo global se debe a varios factores: a la interacción electro estática entre los átomos [21], fuerzas de van der Waals, complementariedad geométrica entre las superficies de las proteínas en el punto, contacto de cadenas laterales [22], entre otros. El encontrar esta posición es una tarea compleja ya que no se tiene conocimiento de la función de energía para toda la superficie de contacto entre las proteínas, la que además, generalmente cuenta con mínimos locales (que aumentan junto con el tamaño de la proteína), por lo que es necesario analizar punto a punto mediante un proceso de búsqueda. Para lograrlo, generalmente de deja fija un proteína y se mueve la otra alrededor para probar las diferentes locaciones y orientaciones entre ambas, si además se agregan más grados de libertad, como permitir el movimiento de cadenas laterales o de la columna de la proteína (acoplamiento flexible), el tamaño del espacio de búsqueda aumenta de manera proporcional, haciendo más difícil encontrar al mejor candidato. Por otra parte, el decidir cuál es el mejor candidato de entre todas las conformaciones distintas implica comparar entre todos ellos aquel que tenga en mínimo global, y si la cantidad de mínimos locales crece exponencialmente con el tamaño de las moléculas, este problema es de una complejidad NP-Hard [23].

2.5. Formalización del Problema

Para tener un mejor entendimiento del problema y de las variables involucradas, a continuación se presenta una formalización que describa de manera general el problema:

Sea P el conjunto de proteínas, con $p_1, p_2 \in P$, Ω el conjunto de configuraciones rotacionales de las proteínas, con ω_1, $\omega_2 \in \Omega$, la función $pos(p_1, p_2, \omega_1, \omega_2) \rightarrow \mathbb{R}^3 \times \mathbb{R}^3$ que genera el conjunto de todas las posiciones posibles para ambas proteínas dadas sus orientaciones, y una función de evaluación $F(p_1, p_2, \omega_1, \omega_2, u, v)$, donde u, v son los vectores de posición de cada proteína, que calcula la energía del sistema de acuerdo a la disposición de ambas proteínas. Se busca encontrar la configuración de posición y orientación para cada proteína, es decir, (u, ω_1) y (v, ω_2) para p_1 y p_2 respectivamente, de tal forma que el valor de F sea mínimo.

Este modelo corresponde un acoplamiento rígido ya que es el que considera una menor cantidad de variables. Un acoplamiento flexible involucraría considerar variables para los movimientos la columna de la proteína, de sus cadenas laterales, u otros ejes. En el próximo capítulo se presentan múltiples trabajos que abarcan ambos tipos de acoplamientos, y, si bien, el manejo de una mayor cantidad de variables corresponde a una representación más fiel, por ende que llega a mejores soluciones, surgen las siguientes interrogantes: ***¿qué tan buenos resultados se pueden lograr con acoplamiento rígido en comparación a acoplamiento flexible? ¿y cuál es el "piso mínimo" para lograr una predicción cercana a los complejos naturales?***.

Capítulo 3

Estado del Arte

Durante gran parte del siglo XX, una de las principales herramientas para determinar la estructura tridimensional de complejos proteicos y proteínas oligoméricas fue la cristalografía de rayos X. Este proceso consiste en generar cristales a partir de una solución purificada con una alta concentración de la proteína que luego son expuestos a rayos X para analizar su patrón de difracción. Este patrón permite identificar la distribución de electrones en la proteína y se genera un mapa de densidad electrónica, el cual es utilizado para determinar la posición de los átomos, tal como se muestra en la Figura 3.1. En [24] se hace una revisión de cómo se realiza este proceso, desde cómo se hacen crecer los cristales hasta como se procesa el mapa de densidad electrónica. No obstante, la cristalización es compleja y no funciona para proteínas flexibles, ya que requiere estructuras cristalizadas en las que muchas moléculas tengan una misma alineación, volviendo prácticamente invisibles a las porciones flexibles de una proteína en el mapa de densidad electrónica [25].

Por otro lado, uno de los primeros intentos de detectar interacciones fue a través de cromatografía de afinidad para detectar interacciones. Este método consiste de tres pasos principales: *(i)* se agrega un mezcla de proteínas a una columna con resinas a la cual puede unirse la proteína objetivo.*(ii)* se realiza un lavado de la columna que se encargará de eliminar el resto de proteínas que no se vincularon a la resina y (iii) se agrega un solvente que separe la proteína de la resina. La Figura 3.2 ejemplifica este proceso. Inicialmente este procedimiento era utilizado para

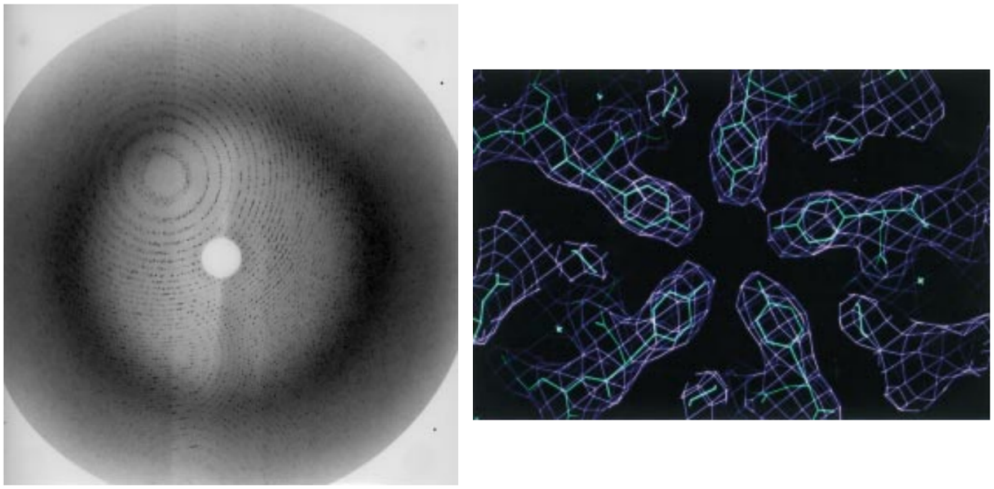


Figura 3.1: En la imagen de la izquierda se muestra una difracción de rayos X de enterovirus bovino. En la derecha se encuentra una porción de un mapa de densidad de electrones del corte de un virus [24].

purificar enzimas y fue popularizado por [26]. Posteriormente se aplicó en variados trabajos tales como [27], [28] y [29], para estudiar la interacción entre proteínas y *Escherichia coli*, las interacciones en "máquinas proteicas" y las interacciones entre proteínas y el citoesqueleto, respectivamente.

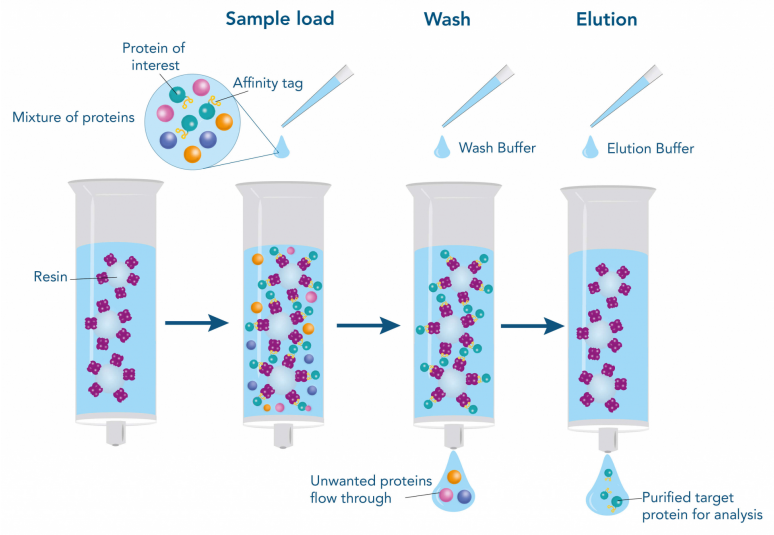


Figura 3.2: Esquema del proceso de cromatografía de afinidad [30].

Sin embargo, tanto la cristalografía de rayos X para determinar las estructuras como la cromatografía de afinidad para predecir las interacciones, al ser métodos físicos experimentales no lograban abarcar toda la gama de interacciones. Más adelante surge la espectroscopia de resonancia magnética nuclear como alternativa a la cristalografía para obtener las estructuras moleculares. Es una técnica espectroscópica que, basándose en que los núcleos atómicos están cargados eléctricamente, utiliza campos magnéticos para generar transferencias de energía que es absorbida por los átomos. Éstas transferencias tiene frecuencias que son características de cada átomo y de su entorno químico, y los cambios en resonancia de los núcleos atómicos son captados por radio receptores para determinar las posiciones atómicas [31]. Esta técnica está limitada a proteínas pequeñas o medianas debido a problemas de superposición de cimas de ondas en el espectro de resonancia. Aún con nuevos y sofisticados métodos, existía una gran brecha entre la cantidad de cantidad de componentes (proteínas) con estructura conocidas y los complejos que integran.

Es por esto que nace la necesidad de apoyarse en métodos computacionales para solventar esa gran diferencia. Uno de los trabajos pioneros [32] se centró en la interacción de la tripsina con su inhibidor, presentando un algoritmo basado en la minimización de la energía (ME) para calcular la mejor posición de acople. En él, las moléculas de las proteínas son representadas como esferas, y se definen puntos sobre su superficie dispuestos en forma de grilla, ordenados por latitud y longitud, y son los lugares de interacción entre ambas y en donde se medirá la energía. Dado que ocupan una representación rígida de las proteínas, la función de energía depende de dos contribuciones: la interacción entre pares de aminoácidos no vinculados (V_{nb}) y la interacción entre proteína-solvente (V_S). Además, se describe que al optimizar la función, el componente V_S se mantiene débil con respecto al otro, pero actúa como una restricción para mantener a las moléculas cercanas entre sí. En [21], al igual que el trabajo anterior, se consideran niveles energéticos a lo largo de la superficie, pero con ubicación de puntos y funciones de energía distintas. Se discretiza en forma de grilla una serie de planos XY a los largo del eje Z, intersectando a la proteína, y en cada punto se evalúa la función de energía. SU función de energía está formada por tres componentes: la función de energía de Lennard-Jones, una función que mide las fuerzas electroestáticas, y una función basada en los enlaces de hidrógeno.

Una de las primeras propuestas de predicción basada en complementariedad geométrica [33] utiliza este método para formar el dímero $\alpha\beta$ de la hemoglobina. En ese trabajo, se plantean tres criterios que son esenciales para una buena medición

de complementariedad geométrica y buena función de forma: (1) debiese ser local y no depender de partes alejadas en la proteína, ya que las asociaciones ocurren exclusivamente en dicho nivel; (2) debe ser independiente del sistema coordenado, en el caso que ambas estructuras a evaluar no lo compartan, porque si así fuera, existirían dos funciones con su propio sistema coordenado y sería muy difícil calcular complementariedad entre ambas; y (3) tiene que usarse un método fácil y rápido para determinar si existe complementariedad de forma entre dos regiones. Con esto en mente, se propone un sistema de caracterización geométrica entre pomos y agujeros, que se distinguen en base a la cantidad de volumen de proteína que queda dentro de un esfera puesta en la superficie, La Figura 3.3(a) ejemplifica esto. Con estas categorías, el algoritmo busca asociaciones entre pomos y agujeros (véase Figura 3.3(b)) en puntos críticos entre las superficies de ambas, junto con un campo de vectores basado en los centroides de los pomos y agujeros.

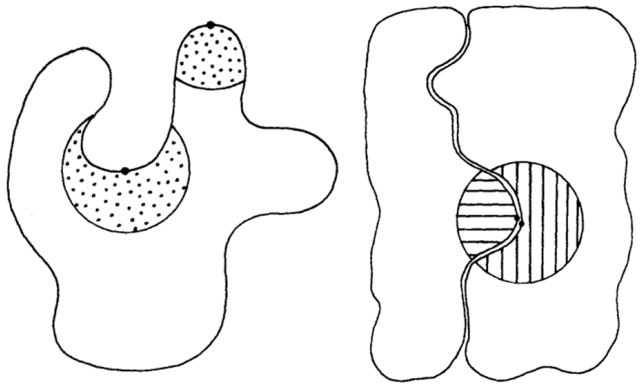


Figura 3.3: a) Sección transversal de un proteína, a partir de los puntos en la superficie se define una esfera que abarca volumen de la proteína (zona punteada) y caracteriza la superficie entre pomos (arriba) y agujeros (abajo). b) Complementariedad de forma. Para dos puntos en la superficie, habrá complementariedad si la suma de ambos volúmenes sea cercana a la esfera completa [33].

Más adelante, en [34] se presenta un algoritmo de búsqueda geométrica completa para interacciones de molécula sobre proteína, siendo uno de los primeros en proponer la representación de la proteína en un volumen 3D, cada punto de la grilla tendrá un valor de 1 si corresponde a la superficie de las proteínas, valores ρ y δ para la zonas de núcleo de cada uno y 0 para los puntos exteriores (Figura

3.4). Para calcular el acople se hace mediante funciones de correlación (Transformada discreta de Fourier y Transformada Inversa de Fourier) entre las grillas de ambas proteínas, obteniendo un valor de 0 para cuando no existe contacto y un valor positivo para el contacto entre ambas superficies. Para evitar que exista penetración de una proteína en la otra, los valores de ρ y δ se fijan en número negativo muy grande y un valor positivo pequeño (ej. 1), respectivamente, para que la multiplicación entre ambos actúe como una penalización para el valor de la correlación.

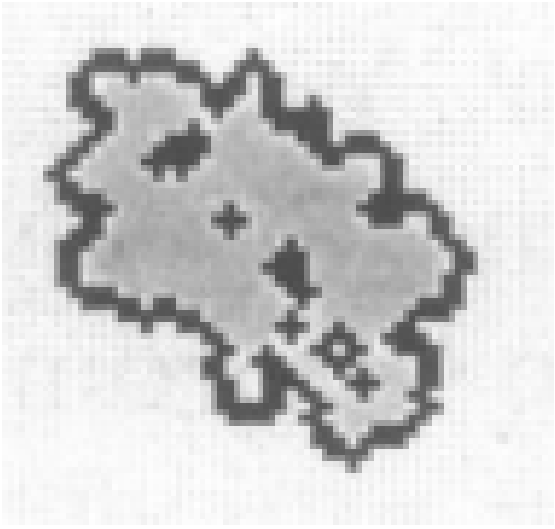


Figura 3.4: Representación de un proteína en una grilla. En negro están los puntos que pertenecen a la superficie, en gris los puntos interiores de la proteína y en blanco el exterior [34].

En [22] se propone el algoritmo BiGGER, que complementa complementariedad geométrica como un primer paso de selección de candidatos, con minimización de la energía para encontrar el óptimo. Además, otorga libertad a las cadenas laterales y no considera sus colisiones para considerar los cambios estructurales de las proteínas al formar complejos. A este enfoque se le conoce como *soft docking* (acoplamiento suave). Más específicamente, la "suavidad" comprende imprecisiones en las estructuras y en sus cambios conformacionales. En [35] se usa simulated Annealing junto con ME en su algoritmo HADDOCK, que consta de tres etapas. La primera etapa consiste en colocar inicialmente ambas proteínas a distancia de 150 [Å] (Ångström) y son rotadas de manera aleatoria alrededor de su centro de masa para luego realizar ME de cuerpo rígido, primero con ciclos de optimización rotacional y después optimización rotacional y traslacional. En la segunda etapa

se realiza simulated annealing semirígido en espacio de torsión angular (TAD-SA), que consiste en 3 ejecuciones de simulated annealing: la primera para optimizar las orientaciones de las proteínas considerándolas como cuerpos rígidos, la segunda permite el movimiento de las cadenas laterales en la zona de interfaz y la tercera permite movimientos en las cadenas laterales y en la columna en la zona de la interfaz para arreglos conformacionales. La tercera etapa es un refinamiento de la cascara de moléculas de agua que se encuentren a una distancia máxima de 8 [Å] mediante simulated annealing y dinámica molecular. Este trabajo sería mejorado en [36], en donde se estudia el impacto de cambiar la escala de propensión de contactos mediados por agua, que está relacionada a la probabilidad de que un par de aminoácidos se enlacen mediante puntes de agua. Allí se propone el uso de la escala de hidrofobicidad de Kyte-Doolitle[37], que resulta en incremento de la tasa de éxito en un 10 % en puntuación de estructuras de manera singular y clusters. En [38] se propone usar ME para la predicción de interacciones proteína-proteína con el factor diferenciador de tener el cuenta la forma de la función de energía libre. En este trabajo, se plantea que esta función podría tener una forma de embudo multidimensional, por lo tanto, el diseñar métodos de optimización teniendo eso en consideración trae consigo dos potenciales ventajas: los algoritmos específicos para minimizar funciones con dicha forma pueden ser más efectivos que los algoritmos genéricos, y, segundo, servirá como una prueba rigurosa de qué tan bien representa un embudo la forma de la función de energía. El método utilizado es de minimización por subestimación, un método específico para funciones con forma de embudo y que sugiere que la función de energía puede ser localmente subestimada por una función convexa. El algoritmo desarrollado utiliza mínimos locales encontrados en la función de energía para subestimar una función cuadrática general. El estimador es obtenido tras resolver un problema de programación semi-definida, dando el nombre al algoritmo SDU (del inglés *Semi-definite programming-based underestimation*). El algoritmo progresa de manera iterativa, en cada iteración se construye una lista de mínimos locales que contiene los puntos mínimos del subestimador, además se agregan puntos generados aleatoriamente con probabilidad proporcional a su cercanía al mínimo del subestimador. A partir de estos puntos se generan nuevas conformaciones para alcanzar el mínimo global. En la Figura 3.5 se representa la función de energía y el subestimador del algoritmo.

Durante la de década del 2010 aparecen también algoritmos genéticos basados en PSO[1] como los son SwarmDock [39] y EigenHex [40] que han sido usados para optimizar sobre el espacio conformacional considerando flexibilidad de columna

[1]PSO: Particle Swarm Optimization

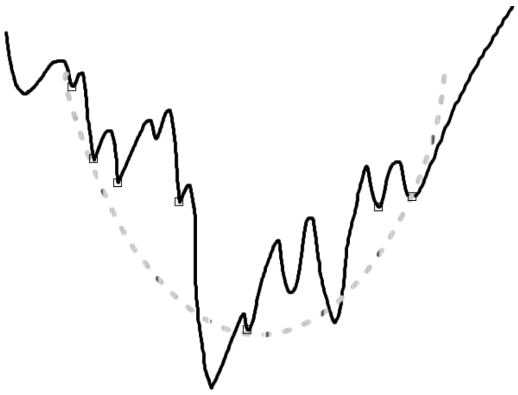


Figura 3.5: Representación de la función de energía con forma de embudo. Los mínimos locales se muestran con cuadrados y la función subestimadora en línea punteada [38].

en las proteínas. PSO es un algoritmo genético que evoluciona un enjambre de partículas que corresponden a un conjunto de a distintas conformaciones del complejo en el espacio conformacional. Cada partícula navega dicho espacio concierta velocidad de acuerdo a la que distancia que tenga con el punto de menor energía encontrado anteriormente por dicha partícula o sus vecinos. El algoritmo de FiberDock [41] se construye bajo el modelo de ajuste inducido (*induced-fit* en inglés) de acoplamiento de proteínas, el que postula que las estructuras entre las proteínas son parcialmente compatibles, y que durante el acoplamiento, la fuerzas químicas propias de la interacción inducen sus cambios conformacionales. Para aplicarlo, se utiliza refinamiento de acoplamiento, que refina los candidatos a solución y los clasifica en base al análisis de modos normales (o NMA, del ingles *Normal Mode Analysis*) considerando flexibilidad tanto en la columna como en las cadenas laterales. Estos modos normales definen una red de interacciones armónicas en la proteína que caracterizan cómo se propagan los cambios estructurales producto del movimiento molecular de una proteína al ligarse a otra. Estos métodos de mayor resolución; que consideran proteínas a un nivel atómico, y con un mayor grado de libertad en la columna y sus cadenas laterales; proveen mejores resultados debido a que existe mayor información adicional más allá de la estructura [42].

Posteriormente, gracias a los avances de internet y las tecnologías de desarrollo web, es que actualmente existen *web servers* tales como FRODOCK [43], ClusPro [44] o HDOCK [45] que permiten realizar experimentos de acoplamiento utilizando

archivos estructurales (generalmente en formato PDB[2]) subidos a la plataforma por usuarios en tiempo real. Tradicionalmente, las funciones de puntuación o scoring estiman la calidad de un modelo obtenido por acoplamiento y se pueden clasificar en tres tipos: (i) aquellas basados en energía física, que suelen ser una combinación lineal ponderada de diferentes términos energéticos; (ii) aquellas basadas en potenciales estadísticos, que convierte las distribuciones de contactos de pares residuo-residuo que depende de la distancia en potenciales a través de inversión de Boltzmann; y (iii) aquellas basadas en machine learning, cuyo acercamiento permite descubrir combinaciones no lineales complejas entre los atributos para poder clasificar los modelos en cercanos a nativos o no. No obstante, han surgido opciones de funciones de puntuación basada en grafos, como iScore [46]. En iScore se utilizan grafos para representar el área de interfaz entre dos proteínas, los nodos corresponden a residuos y las aristas a los contactos entre ellos. Para determinar la calidad de un modelo, se calcula la similitud de su grafo con grafos con interfaces positiva (nativas) y negativas (no nativas) para predecir la pertenencia a cada clase. Por otro lado, en [47] se reconocen que todas estas aproximaciones al problema se sustentan en una selección de características basadas en conocimiento del dominio, pero que no necesariamente han sido optimizadas para el propósito de puntuar. Para corregir esto, proponen una red convolucional de grafos basada en energía, que no sólo prediga la calidad del modelo sino también logre aprender cuales son los atributos más importantes para ese proceso. En los avances más recientes de este año, [48] propone un método de optimización de caja negra basado en tren de tensores, que puede ser implementado tanto en CPU's como GPU's, e incluso, dada su estructura podría ser extendido unidades de procesamiento cuántico, cuando éstas alcancen una mayor escala y confiabilidad.

[2]Formato del Protein Data Bank

Capítulo 4

Diseño de solución

Debido a que se realizará un proceso de búsqueda para predecir el acople entre las proteínas, es importante tener claro el espacio de búsqueda. El espacio de búsqueda corresponde al espacio (multi)dimensional en el cual se encuentran las soluciones a un problema de optimización. Como se menciona en la Sección 2.5, en términos generales ese espacio de búsqueda está dado por el espacio rotacional y traslacional de las proteínas, es decir, cómo están orientadas y ubicadas entre sí. Los algoritmos diseñados son de búsqueda basada en complementariedad geométrica, esto motivado en que la mayoría de los complejos proteicos presentan coincidencia geométrica en las zonas de interfaz [49]. Además, el espacio estará definido también por la representación de las proteínas, ya que eso dictara que clase de movimientos pueden realizarse. La representación clásica [22; 34] es representar las proteínas en una grilla 3D.

4.1. De proteína a grilla 3D

La búsqueda a realizar está basada en complementariedad geométrica, esto quiere decir que necesitamos representar la superficie de la proteína de manera que podamos calcular el área de contacto entre ambas proteínas. Para este efecto, la grilla será un arreglo tridimensional del tamaño de la proteína, donde cada celda de la grilla es de 1 [$Å^3$]. De acuerdo a la ubicación de los átomos de la proteína la

grilla se irá llenando, y una vez generada, se podrán generar la grilla de superficie y núcleo. En la Figura 4.1 se representa el paso de proteína a grilla 3D.

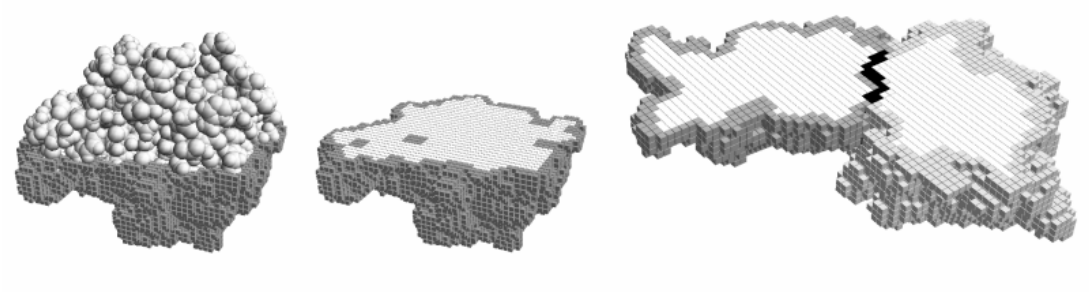


Figura 4.1: La imagen a la izquierda muestra a la parte superior de una proteína, con los átomos en gris, con la mitad inferior en forma de grilla. En la imagen del centro se muestra un corte de una proteína, en gris oscuro la parte de superficie, y en gris claro la de núcleo. En la imagen de la derecha se muestra dos proteínas en contacto y en negro se muestra la zona de interfaz. Fuente: [50]

4.1.1. Grilla de proteína

El proceso de generación de la grilla de una proteína consiste en marcar en el arreglo tridimensional toda casilla cuyo centro se encuentre dentro del rango de [1Å] de la esfera de *van der Waals* de cualquier átomo perteneciente a la proteína. La esfera de *van der Waals* es una esfera imaginaria cuyo radio representa la mínima distancia a la que puede encontrarse la nube electrónica de otro átomo sin que exista repulsión entre ellos. En la Figura 4.2 se muestra un esquema de cómo se marcan las celdas de la grilla. En este paso, tanto las celdas punteadas como achuradas son marcadas de la misma manera. El algoritmo 1 describe este proceso, donde r^a_{vdw} corresponde al radio de van der Waals del átomo a y r^a_{cov} su radio de cobertura, este radio define la esfera alrededor del átomo que se considerará parte de la proteína.

4.1.2. Grilla de superficie y núcleo

Una vez definida la grilla de la proteína, se pueden construir las grillas de superficie y núcleo. La grilla de superficie es la capa exterior de la proteína y que está vacía.

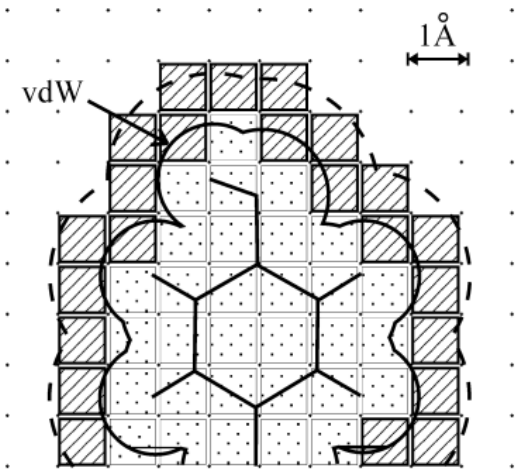


Figura 4.2: Representación 2D de la grilla de una proteína. La cadena proteica está representada por las lineas negras en el centro. Las casillas punteadas son aquellas que se encuentran dentro del radio de van der Waals (línea de contorno continua) de cada átomo. Las casillas achuradas son aquellas dentro del rango de 1 [Å] extra (línea punteada). Fuente: [22]

Algoritmo 1 Algoritmo de generación de grilla 3D

Require: $p \in P$
  return matriz de proteína G_p
  $D \leftarrow dimensions(p)$
  inicializar matriz G_p en base a D
  for cada átomo a en p **do**
    marcar celda G_p de acuerdo a la posición de a
    $r^a_{cov} \leftarrow r^a_{vdw} + 1$
    for all celda c en $vecindad(a, r^a_{cov})$ **do**
      marcar celda c en G_p que tenga su centro dentro de r^a_{cov}
    end for
  end for

Ésta está compuesta por aquellas celdas que tengan al menos una celda vacía[1] (no marcada), en la Figura 4.3 se ejemplifica el proceso. Para generarla, una copia de la matriz es desplazada una posición a la vez (de las 26 totales del espacio 3D[2]) (Figura 4.3 B y G), luego se realiza un *Or Exclusivo (XOR)* con la matriz original (Figura 4.3 C y H). Esta operación toma un valor de Verdadero solo si una celda está marcada y la otra no, dando como resultado dos porciones de grilla: una que pertenece a la proteína original en la parte en que la copia ya no cubre, y otra de parte de la copia que ya no está sobre la original. A continuación se realiza la operación *AND* entre el resultado de la operación anterior y la grilla original (Figura 4.3 D e I). Esta operación se vuelve Verdadera sólo cuando ambas celdas estén marcadas, por lo que nos quedamos con la parte que dejó de estar cubierta de la copia en la proteína original. Esta secuencia debe ser repetida por cada una de las 26 direcciones y cada porción que se obtenga acumularla en la matriz de superficie, el algoritmo describe el proceso.

Algoritmo 2 Generación de grilla de superficie

Require: Grilla de proteína G_p
  inicializar matriz S_p vacía
  for all dirección d **do**
    $C \leftarrow G_p$
    $C \leftarrow shift(C, d)$
    $C \leftarrow XOR(C, G_p)$
    $C \leftarrow AND(C, G_p)$
    $S_p \leftarrow OR(S_p, C)$
  end for
  return Grilla de superficie S_p

Teniendo las grillas de superficie y proteína, la grilla de núcleo se puede obtener como resultado de un *XOR* entre ambas.

[1] Por como se construye a veces hay celdas que pertenecen a esquinas en la grilla que quedan formando parte de la grilla de superficie aún no cuando no tienen como vecina a alguna celda vacía

[2] Las 26 direcciones están compuestas por las 8 direcciones de alrededor, junto con la combinación de esas direcciones con arriba y abajo

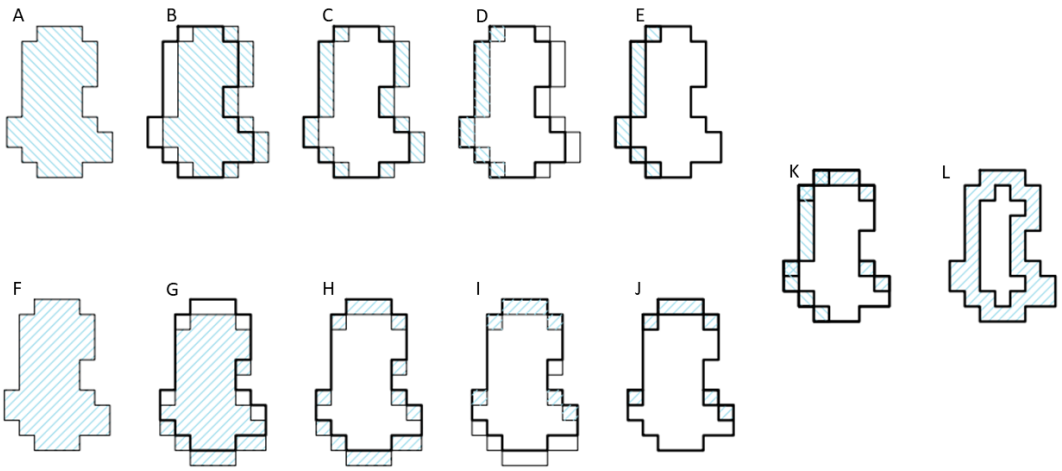


Figura 4.3: La imagen muestra el proceso de generación de la grilla de superficie vista en un plano 2D. La secuencia A-E muestra la porción de superficie obtenida de un shift a la derecha. La secuencia F-J muestra la porción obtenida por un shift hacia abajo. En K están ambas porciones se superficie juntas en la grilla. En L la superficie completa para la proteína. El contorno de la proteína original en negrita.

4.2. Forma y función de evaluación

Para poder decidir que un acople entre dos proteínas es mejor que otro, es necesario tener un criterio para discernir entre ambas, en este caso, el área de superficie de contacto. Es importante notar que no todas las posiciones forman acoples válidos, ya que la superposición de casillas de núcleo de ambas proteínas no está permitida de acuerdo a las fuerzas atómicas presentes. La superposición de celdas de superficie con celdas nucleares no supone acople incorrecto mas no se consideran dentro de la superficie de contacto. Por lo tanto, la calidad del acople estará dado por la cantidad de celdas de superficie superpuestas entre sí. Más formalmente, la función de evaluación del algoritmo puede plantearse como:

Sean D^i las dimensiones de la proteína i, y D^i_x, D^i_y, D^i_z las dimensiones en x, y y z. Sea S_i la función de superficie de una proteína i:

$$S_i(x,y,z) = \begin{cases} 1, & \text{si la celda en } (x,\ y,\ z) \text{ en la proteína } i \text{ es de superficie} \\ 0, & \text{en otro caso.} \end{cases} \tag{4.1}$$

y el área de contacto entre superficies de ambas proteínas se calcula como:

$$AC(p_1,p_2) = \sum_{i,j,k}^{D^1_\alpha} \sum_{l,m,n}^{D^2_\beta} S_1(i,j,k) \cdot S_2(l,m,n) \tag{4.2}$$

, donde D^1_α tomará los valores de D^1_x, D^1_y, D^1_z para i,j,k respectivamente, al igual que D^2_β tomará los valores de D^2_x, D^2_y, D^2_z para l,m,n.

Sea N_i la función de núcleo de una proteína i:

$$N_i(x,y,z) = \begin{cases} 1, & \text{si la celda en } (x,\ y,\ z) \text{ en la proteína } i \text{ es de núcleo} \\ 0, & \text{en otro caso.} \end{cases} \tag{4.3}$$

, el contacto nuclear entre ambas está dado por:

$$CN(p_1,p_2) = \prod_{i,j,k}^{D^1_\alpha} \prod_{l,m,n}^{D^2_\beta} N_1(i,j,k) \cdot N_2(l,m,n) \tag{4.4}$$

Finalmente, la función de evaluación F tendrá la forma de:

$$F(p_1,p_2) = \begin{cases} AC(p_1,p_2), & \text{si } CN(p_1,p_2) = 0 \\ -1, & \text{si } CN(p_1,p_2) \neq 0 \end{cases} \tag{4.5}$$

4.3. Algoritmos de búsqueda local

Un algoritmo de búsqueda local, a diferencia de uno de búsqueda completa, es un algoritmo que solo recorre el espacio de búsqueda de manera parcial. El espacio que visita está determinado por su vecindario, que corresponde a un conjunto de

soluciones que se encuentran cercanas a la solución actual. Lo que define que una solución sea cercana a la actual es el movimiento que pueda realizar el algoritmo para pasar de una solución a otra. Además, las decisiones que toma están basadas en el espacio local en que se encuentra en cierto momento. Por ejemplo, si se encuentra en una grilla 2D y el algoritmo puede moverse a una posición que esté a 1 casilla alrededor de la actual (considerando las diagonales) eso quiere decir que el vecindario actual está compuesto por las 8 casillas circundantes.

Generalmente los vecindarios y movimientos de los algoritmos están definidos por una metaheurística. Una metaheurística es un proceso iterativo que guía la búsqueda, combinando de manera conveniente la exploración y explotación del espacio de búsqueda.Entre las metaheurísticas más conocidas se encuentran: Hill Climbing, Tabu Search, Simulated Annealing, etc [51].

En el caso de esta memoria, la metaheurística a utilizar será de Hill Climbing, la que consiste en "ascender una colina", es decir, en cada movimiento elegir un estado que mejore la solución actual. Existen dos criterios de mejora estricta: alguna mejora (AM) y mejor mejora (MM). AM acepta el primer estado encontrado dentro de la vecindad que mejore la solución. En cambio, MM revisa el vecindario completo y en base a eso toma la decisión de cuál será el próximo movimiento. Existen otros criterios de movimiento que permiten tomar soluciones equivalentes (tienen la misma calidad) o peores con el objetivo de salir de óptimos locales. No obstante, para esta investigación se utilizaron solo criterios de mejora estricta.

Además de una metaheurística, los algoritmos pueden contar con una estrategia de reinicio que permite empezar la búsqueda en puntos diferentes del espacio para evitar quedarse con un único mínimo local. Dada la estructura de grilla escogida, el reinicio de los algoritmos está asociado a la aplicación de una rotación a las coordenadas iniciales, para permitir que se explore todo el espacio rotacional mediante diferentes orientaciones entre ambas.

A continuación se detallan los dos algoritmos a comparar.

4.3.1. Búsqueda por esfera

La búsqueda por esfera consiste en mover una proteína hacia las áreas circundante en el espacio hasta encontrar la posición de mayor contacto, con un criterio de MM. El primer paso es dejar las proteínas una al lado de la otra pero sin que exista contacto entre ellas, está será la posición inicial. A partir de allí, una de las proteína se deja fija,digamos p_1 y la otra proteína p_2 será la que se mueva. Las

posiciones a las que p_2 podrá moverse están determinadas por dos parámetros que describen la esfera vecindario alrededor de la proteína: el radio r y el ángulo θ. El radio r define el tamaño de la esfera, es decir, qué tan lejos estará la próxima solución de la solución actual, siendo 1 su valor mínimo. θ describe qué posiciones podrá ocupar p_2 en la superficie de la esfera, ya que la divide de manera angular las circunferencias de los planos XY e XZ. Por ejemplo, si los parámetros elegidos fuera $r = 3$ y $\theta = \pi/2$ eso quiere decir que las soluciones vecinas se encuentran a 3 celdas de distancia y en rotaciones de $\pi/2$ radianes (90°) de la posición actual. El algoritmo 3 muestra una implementación general de búsqueda por esfera, donde $V_S(r, \theta)$ es la función que entrega las posiciones de la vecindad de esfera de parámetros r y θ.

Algoritmo 3 Búsqueda por esfera

Require: $S_p^1, N_p^1, S_p^2, N_p^2$ grillas de superficie y núcleo.
  $best\_score \leftarrow 0$
  $curr\_best \leftarrow 0$
  inicializar proteínas en posición inicial (u_0, v_0).
  loop
    for all posición vecina v en $V_S(r, \theta)$ **do**
      $x \leftarrow F(u_0, v, S_p^1, N_p^1, S_p^2, N_p^2)$
      if $x > curr\_best$ **then**
        $curr\_best \leftarrow x$
      end if
    end for
    if $curr\_best > best\_score$ **then**
      $best\_score \leftarrow curr\_best$
    else
      return $best\_score$
    end if
  end loop

4.3.2. Búsqueda por eje

La búsqueda por eje se caracteriza por buscar a lo largo de los ejes cartesianos con criterio de AM, una vez que se encuentra una solución mejor se cambia el eje a recorrer. Para inicializar la posición de las proteínas, se selecciona un eje a (entre x, y, o z) al azar con igual probabilidad. Luego se considera un espacio

del tamaño $L = 2D_a^2 + D_a^1$, donde D_a^i es la dimensión de la proteína i en el eje a seleccionado. La proteína p_2 se coloca al comienzo del eje a y se centra en los otros dos. La proteína p_1 se centra en los tres ejes, quedando en el centro del eje seleccionado, alineada con p_2. Para la búsqueda, p_2 se va moviendo una cantidad *step* de casillas por vez hasta encontrar una solución mejor. Una vez encontrada se debe seleccionar el próximo a eje a visita de manera al azar, teniendo el eje anterior al actual un probabilidad q de ser escogido y el tercero una probabilidad $1 - q$. En el caso de la primera iteración se realiza de manera equiprobable. Si en el siguiente eje escogido se encuentra una mejora, se repite el proceso de selección, si no, la búsqueda pasa al eje no seleccionado. Si en ninguno de los dos ejes se encuentra una mejora, el algoritmo termina. El algoritmo 4 describe de manera general este proceso, con $V_E(q, a)$ como función que escoge a los ejes distintos de a, y un parámetro de probabilidad q.

4.4. Medición del ECM con Cadenas de carbones alfa

El error cuadrático medio (ECM) o error de raíz cuadrada media corresponde a la desviación estándar de la diferencia entre un valor de predicción y el original. Este error mide la dispersión de los datos, es decir, cuánto se aleja el modelo predicho de la realidad. En el caso del acoplamiento de proteínas, este error resulta útil para medir la calidad de los algoritmos y su capacidad para encontrar complejos que sean cercanos a las estructuras nativas. En esta investigación, calcularemos este error usando las cadenas de carbones de cada una las proteínas. Un carbón alfa (C^α) es el átomo de carbono central en un aminoácido y el que está conectado al radical de éste. De esta manera, la cadena de carbones corresponde al conjunto de C^α de cada aminoácido presente en una proteína. Esta cadena representa la columna de una proteína.

La medición a realizar, una vez que se realizó la predicción del complejo, consiste en extraer las coordenadas de los C^α de cada una de las cadenas del complejo predicho, y calcular el error con las cadenas de cómo se encuentran esas proteínas unidas en la naturaleza. La suma de las diferencias entre las coordenadas se calcula para cada cadena, que luego son sumadas para calcular el error total. A continuación se detalla el calculo realizado.

Sea Z_i^α la cantidad de C^α de la proteína i. Sean $\hat{C}_j^\alpha$ y C_j^α los j-ésimos C^α de una cadena de proteínas, el primero perteneciendo a la cadena predicha y el segundo a la natural. Sea E_i la suma de las diferencias entre las cadenas predichas y nativas

Algoritmo 4 Búsqueda por eje

Require: $S_p^1, N_p^1, S_p^2, N_p^2$ grillas de superficie y núcleo.
  $best\_score \leftarrow 0$
  $curr\_best \leftarrow 0$
  inicializar proteínas en posición inicial (u_0, v_0).
  $a \leftarrow rand\_eje()$
  loop
    $a_1, a_2 \leftarrow V_E(q, a)$
    for all posición v en a_1 **do**
      $x \leftarrow F(u_0, v, S_p^1, N_p^1, S_p^2, N_p^2)$
      if $x > best\_score$ **then**
        $best\_score \leftarrow x$
        $a \leftarrow a_1$
        break
      end if
    end for
    if $x \neq best\_score$ **then**
      for all posición w en a_2 **do**
        $x \leftarrow F(u_0, w, S_p^1, N_p^1, S_p^2, N_p^2)$
        if $x > best\_score$ **then**
          $best\_score \leftarrow x$
          $a \leftarrow a_2$
          break
        end if
      end for
      if $x \neq best\_score$ **then**
        return $best\_score$
      end if
    end if
  end loop

de la proteína i, que se calcula:

$$E_i = \sum_{j}^{Z_i^\alpha} \|\hat{C}_j^\alpha - C_j^\alpha\|^2 \tag{4.6}$$

Finalmente, ECM estará dado por:

$$ECM = \sqrt{\frac{E_1 + E_2}{Z_1^\alpha + Z_2^\alpha}} \tag{4.7}$$

4.5. Tipos de instancias

Se entiende por instancia, al conjunto de proteínas que se intentará acoplar entre sí. Éstas instancias se encuentran separadas en tres grupos diferentes de acuerdo a la dificultad de realizar el acoplamiento.

- **Bound**: En este grupo los complejos son reconstruidos a partir de estructuras ya cristalizadas [3] separadas de un complejo.
- **Pseudo-unbound**: En este grupo una de las estructuras se encuentra en su forma libre o nativa (corresponde a la forma en que se puede encontrar la estructura sin estar acoplada a otra en la naturaleza) y la otra en su forma cristalizada. En este grupo también se incluyen instancias en las que un monómero se acopla con una copia de sí mismo en vez de con su par. Este grupo corresponde a una dificultad media ya sólo una de sus cadenas está cristalizada.
- **Unbound**: En este grupo ambas estructuras se encuentran en su forma libre o nativa y corresponde al grupo de mayor dificultad. En este caso ambas cadenas presentan diferencias con respecto a su estructura a cuando están acopladas.

[3] Proteína cristalizada: Proteína estabilizada en una estructura de cristal. Al ser separadas del complejo estando cristalizadas, mantienen la forma en la que se acoplaron.

Capítulo 5

Resultados

En este capítulo se revisa la configuración de los experimentos y algoritmos, junto con el conjunto de instancias a utilizar. Se presentan las características del entorno de ejecución en el cual se realizan los experimentos y se analizan los resultados obtenidos. Finalmente se hace un análisis estadístico para determinar si existe diferencia en el desempeño de los algoritmos.

5.1. Configuración de los algoritmos y la experimentación

Los algoritmos utilizados en la experimentación son uno de búsqueda por esfera (Sección 4.3.1) y uno de búsqueda por eje (Sección 4.3.2). El algoritmo de búsqueda por esfera tiene como parámetro $r = 1$ y $\theta = \pi/4$ (45°) y algoritmo de búsqueda por eje sus parámetros son $step = 1$ y $q = 0{,}5$. Además ambos cuentan con reinicio para cambiar la orientación de las proteínas. El de búsqueda por esfera aplicaba rotaciones de $\pi/4$ (90°) alrededor de cada eje para ambas proteínas, es decir, 4 posiciones por cada eje, 64 orientaciones por proteína, totalizando 4096 de reinicios. Para el algoritmo de eje, como la proteína p_2 atraviesa a la proteína p_2, la proteína p_1 considera solo 8 orientaciones, mientras que la segunda mantiene los 8, dando un total de 512 reinicios.

Las instancias utilizadas para comparar el desempeño de los algoritmos, están separadas en los grupos mencionados en la Sección 4.5. En la tabla 5.1 se encuentran listadas las instancias de tipo *Bound*, en la tabla 5.2 están las instancias de tipo *Pseudo-unbound* y finalmente en la tabla 5.3 están las instancias de tipo *unbound*. Cada tabla incluye la sigla de identificación de la instancia, el nombre, y el(los) archivo(s) asociado(s). Todos ellos fueron obtenidos del Protein Data Bank [4] en formato PDBx/mmCIF y el listado de instancias seleccionado, fue extraído desde [22].

Para cada ejecución, se consideró un tiempo máximo de 14 horas, y para evitar que hubiera un sesgo con respecto a que algunas orientaciones nunca se visitaran por límite de tiempo, se aplicó una rotación aleatoria a las coordenadas de ambas proteínas, que luego recibirían las rotaciones de cada reinicio.

ID	Nombre	Archivo(s) PDB
2SICXX	Subtilisin-inhibitor (wt)	2sic
1SBNXX	Subtilisin-eglin c	1sbn
1TECXX	Thermitase-eglin c	1tec
1ACBXX	a-chymotrypsin-eglin c	1acb
3SDHXX	Clam hemoglobin dimer 7	3sdh
2PCCXX	CcP-cytochrome c	2pcc

Tabla 5.1: Lista de instancias *Bound*

ID	Nombre	Archivo(s) PDB
3SDHXF	Clam hemoglobin dimer	3sdh
1DXGXF	Desulforedoxin dimer	1dxg
6EBXXF	Erabutoxin dimer	6ebx
2MIPXF	HIV-2 protease dimer	2mip
1YQVXF	HyHel5 Fab-lysozyme	1lza, 1yqv
3HFMXF	HyHel10 Fab-lysozyme	1lza, 3hfm
1CTAXF	Troponin c dimer	1cta

Tabla 5.2: Lista de instancias *Pseudo-unbound*

El entorno de ejecución en donde se llevaron a cabo los experimentos cuenta con un procesador Ryzen 3 3100 de 3.6GHz y 16GB de RAM , el código fue escrito en lenguaje C++ y compilado con g++ versión 9.3.0

ID	Nombre	Archivo(s) PDB
2PTCFF	Trypsin-inhibitor	2ptn, 4pti, 2ptc
2SICFF	Subtilisin-inhibitor (wt)	2st1, 3ssi, 2sic
2PCCFF	CcP-cytochrome c (yeast)	1ccp, 1ycc, 2pcc
2PCBFF	CcP-cytochrome c (horse)	1ccp, 1hrc, 2pcb
2SNIFF	Subtilisin-chymotrypsin inhibitor	1sup, 2ci2, 2sni
1FSSFF	Acetylcholinesterase-fasciculin II	2ace, 1fsc, 1fss

Tabla 5.3: Lista de instancias *Unbound*

5.2. Resultados

En las Tablas 5.4,5.5 y 5.6 se listan los resultados obtenidos por los algoritmos para los tres grupos de instancias con respecto al puntaje de la función de evaluación y el error cuadrático medio (ECM), que están en [Å^3] y [Å] respectivamente. De el total de 19 instancias evaluadas, el algoritmo de búsqueda por eje logró puntajes de complementariedad superiores en 13 de ellos, y empató en 2 (1DXGXF, 6EBXXF). En términos de ECM el desempeño de ambos algoritmos fue similar llevándose 8 el de esfera y 9 el de eje además de dos empates. En particular, el método de búsqueda de ejes produce buenos resultados para las instancias de tipo Bound, teniendo la mayoría de mejores puntajes y ECM, y en uno de ellos la diferencia es < 1[Å]. En las figuras 5.1 y 5.2 están graficados los puntajes y ECM en donde se aprecia con mayor claridad estas tendencias.

Es importante señalar que para que una configuración se considere cercana a una nativa, el ECM debiese ser < 4[Å], lo que no se cumple para ninguna instancia y método, y si bien se hallaron valores "cercanos", los errores su encuentran un orden de magnitud más arriba de lo deseado, por lo que las estructuras predichas no corresponden a los complejos nativos.

En las Tablas 5.7, 5.8 y 5.9 se hace la comparación entre los tiempos de ejecución para cada instancia de parte de ambos algoritmos, en donde se incluye el tiempo promedio de una interacción y el tiempo total de ejecución, medidos en segundos [s] y en horas y minutos [hh:mm] respectivamente. En las instancias Bound y Pseudo-unbound no se aprecia una gran diferencia entre ambos, teniendo el eje un mejor tiempo en cada una. Además hubo 1 y 2 instancias respectivamente que no lograron completar todas sus iteraciones y alcanzaron su tiempo límite. Sin embargo con respecto a las instancias Unbound, el método de esfera logra de

	Esfera		Eje	
ID	Ptje	ECM	Ptje	ECM
2SICXX	**243**	47.31	203	**23.00**
1SBNXX	204	51.29	**234**	**43.29**
1TECXX	236	**14.24**	**282**	15.31
1ACBXX	223	26.68	**258**	**11.34**
3SDHXX	223	**18.23**	**254**	47.45
2PCCXX	257	61.08	**282**	**40.20**

Tabla 5.4: Resultados de puntaje y ECM para instancias *Bound*

	Esfera		Eje	
ID	Ptje	ECM	Ptje	ECM
3SDHXF	200	85.42	**396**	**73.22**
1DXGXF	214	12.90	214	12.90
6EBXXF	306	32.60	306	32.60
2MIPXF	172	73.87	**244**	**20.70**
1YQVXF	273	**17.13**	**311**	37.03
3HFMXF	205	**27.88**	**243**	45.76
1CTAXF	**226**	**11.77**	213	12.37

Tabla 5.5: Resultados de puntaje y ECM para instancias *Pseudo-unbound*

	Esfera		Eje	
ID	Ptje	ECM	Ptje	ECM
2PTCFF	**239**	25.06	216	**11.80**
2SICFF	255	**37.66**	**280**	38.62
2PCCFF	205	**36.76**	**222**	41.09
2PCBFF	177	45.24	**305**	**14.91**
2SNIFF	191	**24.18**	**292**	27.82
1FSSFF	**300**	26.18	257	**15.97**

Tabla 5.6: Resultados de puntaje y ECM para instancias *Unbound*

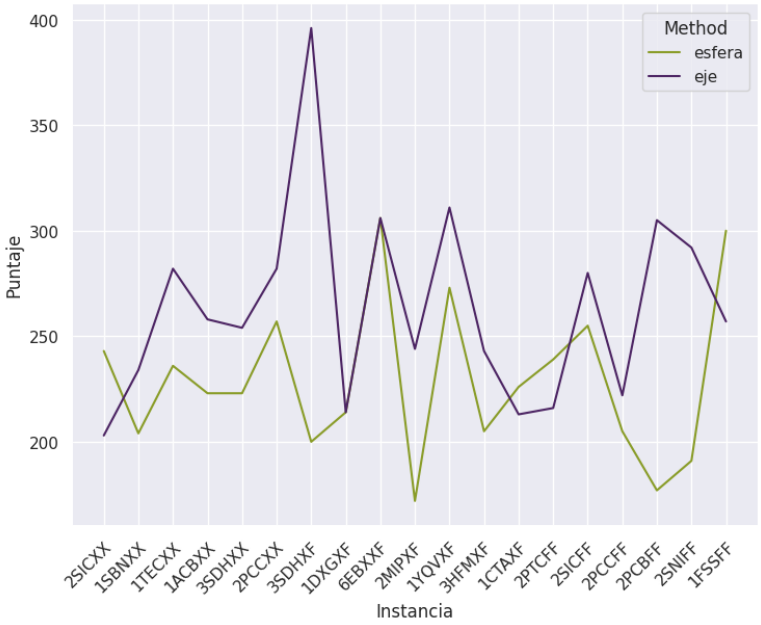


Figura 5.1: Gráfico de los puntajes de complementariedad geométrica obtenidos para cada instancia, separado por método.

manera consistente mejores tiempos que el método de eje, que en dos instancias llegó al tiempo límite. En la Figura 5.3 se grafican los tiempos totales.

En las Tablas 5.10, 5.11 y 5.12 están los datos correspondientes a la cantidad de veces que se llamó la función de evaluación, es decir, la cantidad de configuraciones distintas revisadas por cada uno, tanto promedio por iteración como totales. A lo largo de los tres grupos de instancias, el método de eje mantiene la cantidad de evaluaciones totales más bajas. Estos resultados están influenciados por dos factores principales: el primero a tomar en cuenta es la cantidad de iteraciones, ya que el método de eje realiza 8 veces menos iteraciones que el método de esfera; el segundo factor corresponde al criterio de mejoría, ya que el método de esfera utiliza el criterio MM y el de eje AM. Esto se ve amortiguado por el hecho de que la vecindad del método de eje es mucho mayor que las de esfera. Más claramente eso se nota en los valores de evaluaciones promedio por iteración, en los que el método de eje es mucho mayor que el de esfera, llegando a ser incluso más de 4 veces mayor.

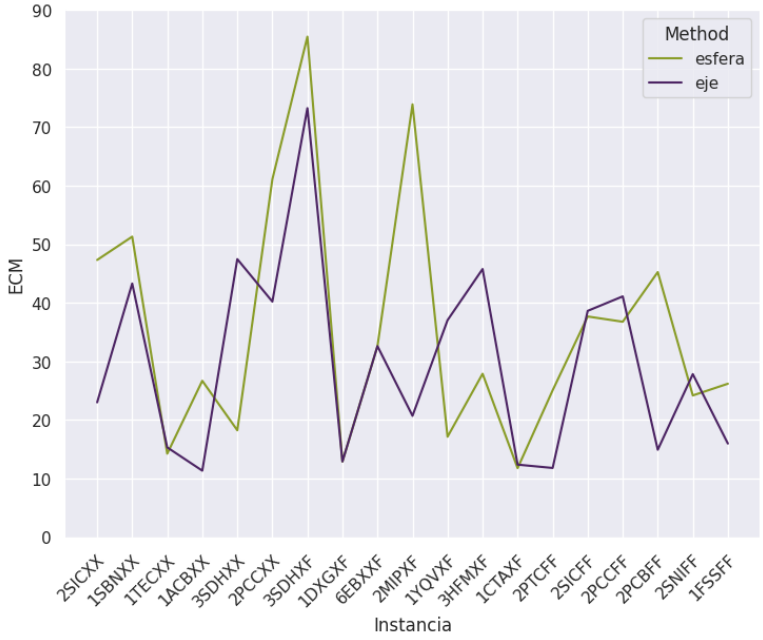


Figura 5.2: Gráfico de los errores cuadrático medios obtenidos para cada instancia, separado por método.

	Esfera		Eje	
ID	Prom	Total	Prom	Total
2SICXX	9.11	11:20	77.49	**11:09**
1SBNXX	6.75	08:29	52.07	**07:31**
1TECXX	6.25	**07:53**	55.15	07:57
1ACBXX	6.65	08:23	53.30	**07:41**
3SDHXX	7.96	**09:58**	75.78	10:54
2PCCXX	28.32	14:00	320.61	14:05

Tabla 5.7: Resultados de tiempo para instancias *Bound*

	Esfera		Eje	
ID	Prom	Total	Prom	Total
3SDHXF	7.89	**09:57**	81.87	11:47
1DXGXF	2.38	02:57	13.26	**01:57**
6EBXXF	3.76	04:37	22.66	**03:16**
2MIPXF	6.33	**07:54**	55.58	08:00
1YQVXF	15.68	14:00	150.06	14:00
3HFMXF	16.28	14:00	161.74	14:00
1CTAXF	3.43	04:16	21.66	**03:08**

Tabla 5.8: Resultados de tiempo para instancias *Pseudo-unbound*

	Esfera		Eje	
ID	Prom	Total	Prom	Total
2PTCFF	7.21	**09:03**	63.36	09:08
2SICFF	8.57	**10:40**	75.32	10:51
2PCCFF	9.47	**11:56**	103.62	14:00
2PCBFF	9.40	**11:55**	97.62	14:00
2SNIFF	7.22	**09:05**	63.56	09:09
1FSSFF	13.36	14:00	116.41	14:02

Tabla 5.9: Resultados de tiempo para instancias *Unbound*

	Esfera		Eje	
ID	Prom	Total	Prom	Total
2SICXX	143	586872	632	**324528**
1SBNXX	138	566592	583	**299290**
1TECXX	123	504920	586	**300701**
1ACBXX	134	549328	595	**305580**
3SDHXX	128	525720	634	**325471**
2PCCXX	134	220402	848	**133139**

Tabla 5.10: Resultados de cantidad de evaluaciones para instancias *Bound*

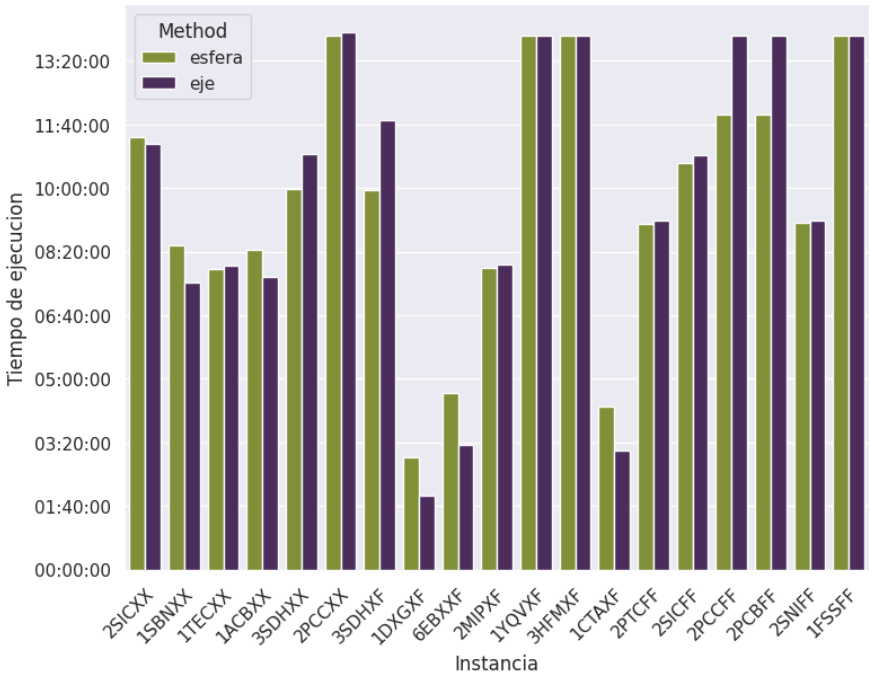


Figura 5.3: Gráfico de los tiempos de ejecución de cada instancia, separado por método.

	Esfera		Eje	
ID	Prom	Total	Prom	Total
3SDHXF	122	502840	633	**324745**
1DXGXF	138	565968	394	**202150**
6EBXXF	140	574600	441	**226366**
2MIPXF	125	512200	567	**291308**
1YQVXF	141	416416	732	**244057**
3HFMXF	138	389610	718	**221929**
1CTAXF	122	499928	416	**213788**

Tabla 5.11: Resultados de cantidad de evaluaciones para instancias *Pseudo-unbound*

	Esfera		Eje	
ID	Prom	Total	Prom	Total
2PTCFF	129	531856	590	**302982**
2SICFF	140	575120	609	**312648**
2PCCFF	130	534976	704	**339638**
2PCBFF	124	508248	675	**345298**
2SNIFF	130	534560	606	**310967**
1FSSFF	149	509834	683	**293433**

Tabla 5.12: Resultados de cantidad de evaluaciones para instancias *Unbound*

5.3. Análisis estadístico

En esta Sección se pretende concluir si un algoritmo es mejor que otro mediante un análisis estadístico. La prueba estadística a utilizar el *Wilcoxon Signed-Rank test* de dos colas con un nivel de significancia de 0.05. La *hipótesis nula* señala que no existe diferencia entre las medias de ambos algoritmos, es decir, no hay diferencia en el desempeño de ambos algoritmos. La *hipótesis alternativa* indica que sí existe diferencia entre ambos algoritmos.

Esta prueba fue aplicada para el valor de puntaje obtenido por ambos métodos ya que fue el valor utilizado para optimizar, en la tabla 5.13 se adjuntan los resultados del test.

Sum of Neg. Ranks	Sum of Pos. Ranks	Z-value	p**-value**
27	126	-2.343	**0.019**
Media (W)	Desv. estandar (W)	W-value	Valor crítico
76.5	21.12	27	34

Tabla 5.13: Resultados del test de hipótesis *Wilcoxon Signed-Rank* de dos colas, con significancia de 0.05 entre método de esfera y eje

En base a este test de hipótesis y con un p-valor de 0.019 se acepta la hipótesis alternativa y se concluye que existe diferencia entre las medias obtenidas por ambos algoritmos.

Capítulo 6

Conclusiones y Trabajo Futuros

En el presente trabajo, se analizaron exitosamente dos estrategias de búsqueda sobre la metaheurística Hill climbing para el problema de acoplamiento entre proteína, siendo comparados estadísticamente, arrojado que el método de búsqueda por eje es significativamente mejor que el método método de esfera, cumpliendo con la directriz principal de la investigación.

A nivel cualitativo, el algoritmo de búsqueda por eje es superior en términos de encontrar lugares de mejor complementariedad geométrica, a un coste de tiempo mayor. No obstante, ninguno de los dos algoritmos logró encontrar conformaciones cercanas a las nativas. Dicho comportamiento se explica por la fijación en el criterio de complementariedad geométrica, dejando de lado otros criterios que son importantes, y deben ser agregados en estrategias futuras.

La elección de la metaheurística de Hill Climbing presenta un buen primer acercamiento para resolver el problema, sin agregar ruido al análisis de las estrategias de búsqueda definidas. A nivel algorítmico, su procedimiento es insuficiente respecto a la complejidad del problema, al no presentar estrategias de diversificación de la búsqueda o salir de máximos locales, a pesar de ser complementado por reinicios. Por lo tanto, el uso de otras metaheurísticas como Tabu Search, Simulated Annealing o Algoritmos Genéticos puede complementar el desempeño de las estrategias de búsqueda analizadas.

6.1. Trabajos Futuros

Las opciones de trabajo futuro para mejorar el desempeño actual, se listan a continuación:

- Integrar otros parámetros de exploración de acoplamiento asociados a la complementariedad geométrica en los algoritmos para que los vecindarios incluyan más configuraciones existentes espacio de búsqueda, evitando caer en óptimos locales.
- Incorporar otros criterio de optimización que suplementen a la complementariedad geométrica. como por ejemplo las fuerzas electrostáticas.
- Trabajar con un conjunto de mejores candidatos encontrados en vez del mejor, de tal forma de someterlos a otro proceso de reparación metaheurística fuera del proceso de optimización geométrica.

Bibliografía

[1] G. M. Morris and M. Lim-Wilby, *Molecular Docking*, pp. 365–382. Totowa, NJ: Humana Press, 2008.

[2] P. Barahona, L. Krippahl, and O. Perriquet, *Bioinformatics: A Challenge to Constraint Programming*, pp. 463–487. New York, NY: Springer New York, 2011.

[3] NIH, "Definición bioinformática." `https://www.genome.gov/es/genetics-glossary/Bioinformatica`, 2023. Accessed 2023-10-07.

[4] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne, "The Protein Data Bank," *Nucleic Acids Research*, vol. 28, pp. 235–242, 01 2000.

[5] I. D. Kuntz, J. M. Blaney, S. J. Oatley, R. Langridge, and T. E. Ferrin, "A geometric approach to macromolecule-ligand interactions," *Journal of molecular biology*, vol. 161, no. 2, pp. 269–288, 1982.

[6] A. Hernández-Santoyo, A. Y. Tenorio-Barajas, V. Altuzar, H. Vivanco-Cid, and C. Mendoza-Barrera, "Protein-protein and protein-ligand docking," in *Protein Engineering* (T. Ogawa, ed.), ch. 3, Rijeka: IntechOpen, 2013.

[7] MedlinePlus, "What are proteins and what do they do?." `https://medlineplus.gov/genetics/understanding/howgeneswork/makingprotein/`, 2021. Accessed 2023-10-06.

[8] A. Sharma, G. Kumar, S. Sharma, K. Walia, P. Chouhan, B. Mandal, and A. Tuli, "Chapter 11 - methods for binding analysis of small gtp-binding

proteins with their effectors," in *Biomolecular Interactions Part A* (A. K. Shukla, ed.), vol. 166 of *Methods in Cell Biology*, pp. 235–250, Academic Press, 2021.

[9] M. Kotlyar, C. Pastrello, A. E. Rossos, and I. Jurisica, "Protein-protein interaction databases," in *Encyclopedia of Bioinformatics and Computational Biology* (S. Ranganathan, M. Gribskov, K. Nakai, and C. Schönbach, eds.), pp. 988–996, Oxford: Academic Press, 2019.

[10] C. ke Chang, S.-M. Lin, R. Satange, S.-C. Lin, S.-C. Sun, H.-Y. Wu, K. Kehn-Hall, and M.-H. Hou, "Targeting protein-protein interaction interfaces in covid-19 drug discovery," *Computational and Structural Biotechnology Journal*, vol. 19, pp. 2246–2255, 2021.

[11] J. Yang, S. J. L. Petitjean, M. Koehler, Q. Zhang, A. C. Dumitru, W. Chen, S. Derclaye, S. P. Vincent, P. Soumillion, and D. Alsteens, "Molecular interaction and inhibition of sars-cov-2 binding to the ace2 receptor," *Nature Communications*, vol. 11, p. 4541, Sep 2020.

[12] S. Xiu, A. Dick, H. Ju, S. Mirzaie, F. Abdi, S. Cocklin, P. Zhan, and X. Liu, "Inhibitors of sars-cov-2 entry: current and future opportunities," *Journal of medicinal chemistry*, vol. 63, no. 21, pp. 12256–12274, 2020.

[13] R. McBride, M. Van Zyl, and B. C. Fielding, "The coronavirus nucleocapsid is a multifunctional protein," *Viruses*, vol. 6, no. 8, pp. 2991–3018, 2014.

[14] D. E. Gordon, G. M. Jang, M. Bouhaddou, J. Xu, K. Obernier, K. M. White, M. J. O'Meara, V. V. Rezelj, J. Z. Guo, D. L. Swaney, *et al.*, "A sars-cov-2 protein interaction map reveals targets for drug repurposing," *Nature*, vol. 583, no. 7816, pp. 459–468, 2020.

[15] Instituto Nacional del Cancer, "`https://www.cancer.gov/espanol/publicaciones/diccionarios/diccionario-cancer/def/apoptosis`, 10 de Diciembre de 2022," 2022.

[16] I. Petta, S. Lievens, C. Libert, J. Tavernier, and K. De Bosscher, "Modulation of protein-protein interactions for the development of novel therapeutics," *Molecular Therapy*, vol. 24, no. 4, pp. 707–718, 2016.

[17] A. W. White, A. D. Westwell, and G. Brahemi, "Protein–protein interactions as targets for small-molecule therapeutics in cancer," *Expert reviews in molecular medicine*, vol. 10, 2008.

[18] M. Jordan, "Mechanism of action of antitumor drugs that interact with microtubules and tubulin," *Current medicinal chemistry. Anti-cancer agents*, vol. 2, pp. 1–17, 02 2002.

[19] L. Chitsike and P. Duerksen-Hughes, "Ppi modulators of e6 as potential targeted therapeutics for cervical cancer: Progress and challenges in targeting e6," *Molecules*, vol. 26, p. 3004, 05 2021.

[20] M. Arbyn, E. Weiderpass, L. Bruni, S. de Sanjose, M. Saraiya, J. Ferlay, and F. Bray, "Estimates of incidence and mortality of cervical cancer in 2018: a worldwide analysis," *The Lancet Global Health*, vol. 8, 12 2019.

[21] P. J. Goodford, "A computational procedure for determining energetically favorable binding sites on biologically important macromolecules," *Journal of Medicinal Chemistry*, vol. 28, no. 7, pp. 849–857, 1985. PMID: 3892003.

[22] P. N. Palma, L. Krippahl, J. E. Wampler, and J. J. Moura, "Bigger: a new (soft) docking algorithm for predicting protein interactions," *Proteins: Structure, Function, and Bioinformatics*, vol. 39, no. 4, pp. 372–384, 2000.

[23] D.-S. Kim, *Protein Docking Problem as Combinatorial Optimization Using Beta-Complex*, pp. 2685–2740. New York, NY: Springer New York, 2013.

[24] M. S. Smyth and J. H. J. Martin, "x ray crystallography," *Molecular Pathology*, vol. 53, no. 1, pp. 8–14, 2000.

[25] PDB, "Methods for determining atomic structures." `https://pdb101.rcsb.org/learn/guide-to-understanding-pdb-data/methods-for-determining-structure`. Accessed 2023-10-09.

[26] P. Cuatrecasas, *Affinity Chromatography of Macromolecules*, pp. 29–89. John Wiley & Sons, Ltd, 1972.

[27] D. Ratner, "The interaction of bacterial and phage proteins with immobilized escherichia coli rna polymerase," *Journal of Molecular Biology*, vol. 88, no. 2, pp. 373–383, 1974.

[28] T. Formosa, J. Barry, B. M. Alberts, and J. Greenblatt, "[3] using protein affinity chromatography to probe structure of protein machines," in *Protein 3- DNA Interactions*, vol. 208 of *Methods in Enzymology*, pp. 24–45, Academic Press, 1991.

[29] K. G. Miller, C. M. Field, B. M. Alberts, and D. R. Kellogg, "[26] use of actin filament and microtubule affinity chromatography to identify proteins that bind to the cytoskeleton," in *Molecular Motors and the Cytoskeleton*, vol. 196 of *Methods in Enzymology*, pp. 303–319, Academic Press, 1991.

[30] I. Lifescience, "Protein affinity chromatography." `https://www.iba-lifesciences.com/applications/protein-affinity-chromatography/`. Accessed: 2023-10-18.

[31] M. K. Singh and A. Singh, "Chapter 14 - nuclear magnetic resonance spectroscopy," in *Characterization of Polymers and Fibres* (M. K. Singh and A. Singh, eds.), The Textile Institute Book Series, pp. 321–339, Woodhead Publishing, 2022.

[32] S. J. Wodak and J. Janin, "Computer analysis of protein-protein interaction," *Journal of Molecular Biology*, vol. 124, no. 2, pp. 323–342, 1978.

[33] M. L. Connolly, "Shape complementarity at the hemoglobin $\alpha 1\beta 1$ subunit interface," *Biopolymers*, vol. 25, no. 7, pp. 1229–1247, 1986.

[34] E. Katchalski-Katzir, I. Shariv, M. Eisenstein, A. A. Friesem, C. Aflalo, and I. A. Vakser, "Molecular surface recognition: determination of geometric fit between proteins and their ligands by correlation techniques.," *Proceedings of the National Academy of Sciences*, vol. 89, no. 6, pp. 2195–2199, 1992.

[35] C. Dominguez, R. Boelens, and A. M. Bonvin, "Haddock: a protein- protein docking approach based on biochemical or biophysical information," *Journal of the American Chemical Society*, vol. 125, no. 7, pp. 1731–1737, 2003.

[36] P. L. Kastritis, K. M. Visscher, A. D. van Dijk, and A. M. Bonvin, "Solvated protein–protein docking using kyte-doolittle-based water preferences," *Proteins: Structure, Function, and Bioinformatics*, vol. 81, no. 3, pp. 510–518, 2013.

[37] J. Kyte and R. F. Doolittle, "A simple method for displaying the hydropathic character of a protein," *Journal of Molecular Biology*, vol. 157, no. 1, pp. 105–132, 1982.

[38] Y. Shen, I. C. Paschalidis, P. Vakili, and S. Vajda, "Protein docking by the underestimation of free energy funnels in the space of encounter complexes," *PLoS computational biology*, vol. 4, no. 10, p. e1000191, 2008.

[39] I. H. Moal and P. A. Bates, "Swarmdock and the use of normal modes in protein-protein docking," *International journal of molecular sciences*, vol. 11, no. 10, pp. 3623–3648, 2010.

[40] V. Venkatraman and D. W. Ritchie, "Flexible protein docking refinement using pose-dependent normal mode analysis," *Proteins: Structure, Function, and Bioinformatics*, vol. 80, no. 9, pp. 2262–2274, 2012.

[41] E. Mashiach, R. Nussinov, and H. J. Wolfson, "Fiberdock: flexible induced-fit backbone refinement in molecular docking," *Proteins: Structure, Function, and Bioinformatics*, vol. 78, no. 6, pp. 1503–1519, 2010.

[42] H. Park, H. Lee, and C. Seok, "High-resolution protein–protein docking by global optimization: recent advances and future challenges," *Current Opinion in Structural Biology*, vol. 35, pp. 24–31, 2015. Catalysis and regulation • Protein-protein interactions.

[43] E. Ramírez-Aportela, J. R. López-Blanco, and P. Chacón, "Frodock 2.0: fast protein–protein docking server," *Bioinformatics*, vol. 32, no. 15, pp. 2386–2388, 2016.

[44] D. Kozakov, D. R. Hall, B. Xia, K. A. Porter, D. Padhorny, C. Yueh, D. Beglov, and S. Vajda, "The cluspro web server for protein–protein docking," *Nature protocols*, vol. 12, no. 2, pp. 255–278, 2017.

[45] Y. Yan, H. Tao, J. He, and S.-Y. Huang, "The hdock server for integrated protein–protein docking," *Nature protocols*, vol. 15, no. 5, pp. 1829–1852, 2020.

[46] C. Geng, Y. Jung, N. Renaud, V. Honavar, A. M. Bonvin, and L. C. Xue, "iscore: a novel graph kernel-based function for scoring protein–protein docking models," *Bioinformatics*, vol. 36, no. 1, pp. 112–121, 2020.

[47] Y. Cao and Y. Shen, "Energy-based graph convolutional networks for scoring protein docking models," *Proteins: Structure, Function, and Bioinformatics*, vol. 88, no. 8, pp. 1091–1099, 2020.

[48] D. Morozov, A. Melnikov, V. Shete, and M. Perelshtein, "Protein-protein docking using a tensor train black-box optimization method," *arXiv preprint arXiv:2302.03410*, 2023.

[49] S. Hubbard, S. Campbell, and J. Thornton, "Molecular recognition: Conformational analysis of limited proteolytic sites and serine proteinase protein inhibitors," *Journal of Molecular Biology*, vol. 220, no. 2, pp. 507–530, 1991.

[50] L. Krippahl and P. Barahona, "Applying constraint programming to rigid body protein docking," in *Principles and Practice of Constraint Programming - CP 2005. Lecture Notes in Computer Science, vol. 3709*, pp. 373–387, 10 2005.

[51] B. Selman and C. P. Gomes, "Hill-climbing search," *Encyclopedia of cognitive science*, vol. 81, p. 82, 2006.