VINÍCIUS DE ALMEIDA PAIVA

GASS-METAL: UM SERVIDOR WEB PARA IDENTIFICAÇÃO DE SÍTIOS METÁLICOS SIMILARES EM PROTEÍNAS BASEADO EM ALGORITMOS GENÉTICOS PARALELOS

Itabira
26 de fevereiro de 2021

Universidade Federal de Itajubá
Programa de Pós-Graduação em
Ciência e Tecnologia da Computação

GASS-METAL: UM SERVIDOR WEB PARA IDENTIFICAÇÃO DE SÍTIOS METÁLICOS SIMILARES EM PROTEÍNAS BASEADO EM ALGORITMOS GENÉTICOS PARALELOS

Dissertação submetida ao Curso de Pós-Graduação em Ciência e Tecnologia da Computação da Universidade Federal de Itajubá como requisito parcial para a obtenção do grau de Mestre em Ciência e Tecnologia da Computação.

VINÍCIUS DE ALMEIDA PAIVA

Itabira
26 de fevereiro de 2021

UNIVERSIDADE FEDERAL DE ITAJUBÁ

GASS-Metal: um servidor web para identificação de sítios metálicos similares em proteínas baseado em algoritmos genéticos paralelos

VINÍCIUS DE ALMEIDA PAIVA

Dr. Sandro Carvalho Izidoro – Orientador
Universidade Federal de Itajubá

Dra. Sabrina de Azevedo Silveira – Co-orientadora
Universidade Federal de Viçosa

Itabira, 26 de fevereiro de 2021

Resumo Estendido

Metais estão presentes em mais de 30% das proteínas encontradas na natureza e desempenham funções biológicas importantes, além de atuarem na manutenção da estrutura de proteínas. Íons metálicos em proteínas estão ligados a grupos de átomos e a esse conjunto é dado o nome de sítio metálico. Um sítio metálico pode exercer funções catalíticas, estruturais, transporte e transferência de elétrons em uma proteína.

Métodos tradicionais e experimentais para a predição de sítios metálicos geralmente encontram empecilhos relacionados a tempo e custo de execução, e por isso cresce a necessidade de ferramentas computacionais que possam auxiliar em predições. Diversos métodos na literatura têm empenhado esforços na predição de sítios metálicos e tem mostrado grandes resultados, porém ainda encontram barreiras por questões relacionadas ao tamanho da proteína, tipo de íons e ligantes, capacidade de encontrar resíduos interdomínio e até mesmo ao obter taxas de acerto não satisfatórias.

O objetivo desta dissertação é adaptar o algoritmo GASS (*Genetic Active Site Search*), inicialmente proposto para a predição de sítios catalíticos, para a busca de sítios metálicos. O método criado, GASS-Metal, divide os resíduos de uma proteína no espaço tridimensional e utiliza paralelismo de algoritmos genéticos para encontrar sítios metálicos candidatos que sejam próximos em relação à distância de *templates* curados provenientes dos M-CSA e MetalPDB.

Os resultados dos testes de *sanidade* e com proteínas homólogas mostraram que o GASS-Metal é um método robusto, capaz de encontrar sítios metálicos em diversos tipos de íons diferentes e não restringe sua busca a uma única cadeia. Além disso, ao usar mutações conservativas, a taxa de acerto na predição melhora ainda mais, ajudando a encontrar sítios em situações onde antes era inviável, pela falta de resíduos em determinadas proteínas.

Em comparação a preditores estado da arte, o GASS-Metal conseguiu desempenho satisfatório na predição de sítios metálicos de diferentes íons. Os resultados mostraram que o método foi superior na predição em 5 dos 12 íons metálicos avaliados e ainda obteve performance equivalente em outros 6 sítios metálicos diferentes.

Abstract

Metals are present in more than 30% of proteins found in nature and perform important biological functions, in addition they act in the maintenance of protein structure. Metal ions in proteins are bounded to groups of atoms and this set is called a metal-binding site. Metal-binding sites can perform catalytic, structural, transport and electron transfer functions in a protein.

Traditional and experimental techniques for metal-binding site prediction usually find obstacles related to time and cost of execution, making computational tools that can assist in predictions become even more important. Several methods in the literature have made efforts to predict metal-binding sites and have shown great results, but they still encounter barriers due to issues related to protein size, type of ions and ligands, ability to find inter-domain residues and even when obtaining not good accuracy rates.

The main goal of this master thesis is to adapt GASS algorithm (*Genetic Active Site Search*), initially proposed for the prediction of catalytic sites, to search for metal-binding sites. The method developed, GASS-Metal, divides residues of a protein in three-dimensional space and uses parallelism of genetic algorithms to find candidate sites that are close in relation to the distance of cured *templates* from M-CSA and MetalPDB.

The results of the *sanity* and homologous protein tests showed that GASS-Metal is a robust method, capable of finding metal-binding sites in different types of ions and does not restrict its search to a single chain. In addition, when using conservative mutations, the prediction accuracy rate improves even more, helping to find sites in situations where it was previously impossible, due to the lack of residues in certain proteins.

In comparison to state-of-the-art predictors, GASS-Metal achieved satisfactory performance in predicting metal-binding sites of different ions. The results showed that the method was superior in the prediction in 5 of the 12 metal ions evaluated and still obtained equivalent performance in other 6 different metal-binding sites.

Agradecimentos

Dedico primeiramente este trabalho à minha família, principalmente aos meus pais, Fátima e Paulo (in memoriam), e meu irmão, Matheus, sem eles nada seria possível. Agradeço à dedicação que sempre tiveram comigo. Além da melhor convivência que eu poderia ter, sempre me apoiaram em todas as decisões e projetos de minha vida. Vocês sempre foram e serão um exemplo para mim e esta realização pessoal certamente é uma realização de todos nós.

Ao meu orientador e amigo, professor Sandro Carvalho Izidoro, pelos incontáveis ensinamentos que me passou, sem ele este trabalho não teria acontecido. Nossas reuniões sempre foram muito produtivas e o trabalho rendeu frutos que irei colher pelo resto da vida. Agradeço também à minha co-orientadora, professora Sabrina de Azevedo Silveira, pelo total suporte que sempre nos deu ao trabalho, mesmo estando distante nunca hesitou em colaborar. Fico muito feliz em poder trabalhar com ambos no doutorado em um futuro próximo.

A todos os colegas de curso da Unifei pelos anos de convivência muito próxima, com compartilhamento de conhecimentos e experiências pelas quais me lembro até hoje. Aos professores do programa de pós-graduação em Ciência e Tecnologia da Computação, por todo conhecimento transmitido durante o curso de mestrado. As disciplinas lecionadas ajudaram em muito a realização deste trabalho.

Por fim agradeço a todos que de alguma forma ajudaram na condução deste trabalho e realização do mestrado. Uma dissertação nunca é fruto dos esforços de uma pessoa apenas e fico feliz que estive cercado de pessoas dispostas a ajudar e de grande competência.

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

A bioinformática trata de aplicar o conhecimento e técnicas da Ciência da Computação para resolver problemas biológicos. Segundo (Luscombe et al., 2001) é uma forma de conceituar a biologia, em termos de macromoléculas e suas propriedades físico-químicas, onde se aplicam técnicas computacionais (incluindo conhecimentos matemáticos e estatísticos) para que as informações associadas a essas moléculas possam ser entendidas e organizadas. Portanto, a Bioinformática trata de uma área multidisciplinar, onde a computação age como meio e não como fim.

Uma das áreas de pesquisa na bioinformática é o estudo de proteínas, que são moléculas grandes e complexas que exercem várias funções nos organismos. Elas são formadas basicamente pela união de aminoácidos e podem assumir diversos tamanhos e formas. As proteínas realizam grande parte dos trabalhos nas células e são necessárias para a estrutura, função e regulação dos tecidos e órgãos de organismos (Keskin et al., 2008).

Inicialmente, informações sobre proteínas eram definidas basicamente sobre sua sequência. Porém, o rápido aumento no número de estruturas macromoleculares tridimensionais disponíveis em bases de dados, tais como o PDB (Berman et al., 2000), fez surgir uma nova subárea da bioinformática: a bioinformática estrutural. A Bioinformática Estrutural também trata da representação, armazenamento, recuperação, análise e apresentação de informações estruturais, mas em escalas atômicas. A bioinformática estrutural possui duas grandes metas: a criação de métodos de uso geral para a manipulação de informações sobre macromoléculas biológicas e a aplicação destes métodos para resolver problemas da biologia gerando novos conhecimentos (Altman e Dugan, 2009).

Entre os desafios da área, pode-se destacar a predição de função de uma proteína. Apesar dos esforços para se definir e anotar as funções das proteínas, o Pfam (base de dados de famílias de proteínas) ainda contém cerca de 22% (3961) entradas definidas como domínios de função desconhecida (Roberts, 2004; El-Gebali et al., 2019; Santana et al., 2020). A função de uma proteína depende de sua estrutura. Proteínas que compartilham de uma mesma estrutura podem ter funções similares. Na falta de dados experimentais, a função de uma proteína pode ser inferida comparando sua estrutura com a estrutura de

uma proteína de função conhecida (Zvelebil e Baum, 2008). Contudo, esse método falha nos casos em que uma proteína tem um tipo de enovelamento muito comum e conservado para famílias funcionalmente diversas.

Neste trabalho, o foco é um grupo determinado de proteínas conhecidas como metaloproteínas. Metaloproteínas são proteínas que contém um ou mais íons metálicos em sua estrutura (Finkelstein, 2009) e estão ligadas a vários processos biológicos, como catálises, transporte, tradução e reconhecimento (Song et al., 2017). Com as estruturas atualmente conhecidas, sabe-se que cerca de 30% das proteínas contém pelo menos um íon metálico (Sobolev e Edelman, 2013). Os íons metálicos presentes nessas estruturas exercem um papel crucial desde a estabilização da estrutura tridimensional da metaloproteína até agindo como cofatores em suas catálises. Íons metálicos em proteínas são ligados por grupos de átomos que doam um par de elétrons para a ligação. Estes átomos geralmente tem carga neutra ou negativa e formam os sítios metálicos.

Devido à sua importância, os aminoácidos de sítios metálicos, assim como de sítios catalíticos e de ligação, foram mais conservados durante a evolução do que as sequências como um todo, e podem ser utilizados com sucesso na busca de sítios similares. Eles são especialmente úteis quando as sequências são muito diferentes ou quando a inferência da função com base unicamente na homologia da sequência não é possível (Izidoro et al., 2014). Métodos tradicionais de busca e predição geralmente encontram empecilhos relacionados ao custo, tempo e automação de processos. Dessa, forma, vários métodos computacionais foram propostos com o objetivo de buscar sítios similares ou mesmo predizer a função de uma proteína, proporcionando a diminuição de custos e do tempo em procedimentos experimentais.

Uma abordagem que já apresentou bons resultados foi a utilização de algoritmos genéticos na busca de sítios ativos similares (catalíticos e de ligação). O GASS (*Genetic Active Site Search* - método de busca de sítios catalíticos e de ligação em proteínas baseado em algoritmos genéticos) (Izidoro et al., 2014) se mostrou muito eficiente em vários testes. Baseado em anotações do Catalytic Site Altas (CSA), o método foi capaz de identificar corretamente mais de 90% dos sítios catalíticos catalogados. No conjunto de dados da competição CASP 10, o método ficou em quarto lugar dentre os 18 métodos participantes.

Viu-se então a oportunidade de verificar se o GASS pode ser adaptado e aplicado na busca de sítios metálicos. Pretende-se com este trabalho de dissertação adaptar o GASS para a busca de sítios metálicos em proteínas. Além disso, vários métodos computacionais de busca de sítios metálicos encontram limitações relacionadas às características dos sítios (e.g. tamanho, diferentes cadeias ou tipos de íons). O trabalho aqui descrito procura trabalhar em tais limitações.

1.1 Objetivos

O objetivo deste trabalho de dissertação é adaptar o GASS para a busca de sítios metálicos em metaloproteínas, criando assim o GASS-Metal. Com a definição de *templates* provenientes de anotações curadas do M-CSA (Ribeiro et al., 2017) em conjunto de sítios metálicos do MetalPDB (Putignano et al., 2017), busca-se utilizar algoritmos genéticos para encontrar sítios metálicos similares a esses *templates*, a fim de prover informações sobre estruturas e funções de metaloproteínas.

Os objetivos específicos são:

- Implementar rotinas para filtrar e organizar dados de proteínas do PDB, M-CSA e MetalPDB;
- Definir um conjunto de *templates* com informações curadas sobre proteínas que contém sítios metálicos;
- Construir um repositório de dados organizados de acordo com sua localização no espaço tridimensional a fim de formar o repositório do algoritmo genético;
- Adaptar o algoritmo genético GASS para o contexto de busca de sítios metálicos similares;
- Analisar e comparar os resultados obtidos a partir do algoritmo genético com os resultados de outras técnicas já implementadas verificando a eficácia e eficiência da modelagem proposta;
- Desenvolver e disponibilizar um servidor web com a ferramenta GASS-Metal.

1.2 Organização do Texto

No Capítulo 2, Revisão da Literatura, foi feito um levantamento bibliográfico de trabalhos da área, discutindo propriedades de sítios metálicos, busca de sítios metálicos similares, bases de dados de proteínas e metaloproteínas, além de trazer um referencial teórico sobre algoritmos genéticos. O Capítulo 3, Metodologia, descreve a estrutura metodológica proposta, pré-processamento de dados, a modelagem do algoritmo genético e os conjuntos de dados e métricas de avaliação utilizados nos testes. Os resultados são apresentados e discutidos no Capítulo 4. O Capítulo 5 é referente à conclusão e traz aspectos relevantes a serem considerados em trabalhos futuros.

Capítulo 2

Revisão da Literatura

Neste capítulo são apresentadas as propriedades de sítios metálicos em proteínas e trabalhos relacionados à busca destes sítios. São tratados tópicos referentes às propriedades dos sítios metálicos, busca de sítios metálicos similares e bases de dados de proteínas. A última seção deste capítulo discute sobre Algoritmos Genéticos (AG), técnica computacional na qual o GASS-Metal se baseia.

2.1 Propriedades dos sítios metálicos

Metais estão presentes em mais de um terço de todas as proteínas encontradas na natureza e são importantes para a função biológica e manutenção da estrutura da proteína (Harding et al., 2010). Diversos processos biológicos e químicos na natureza envolvem a ligação de íons metálicos em proteínas, também conhecidas como metaloproteínas. Muitas reações celulares biológicas requerem metaloproteínas e parte das notáveis e complexas transformações químicas são catalisadas por metaloenzimas (Akcapinar e Sezerman, 2017).

Íons metálicos desempenham diversos papéis em proteínas, como: estabilização, catálise, transdução de sinal, fixação de nitrogênio, fotossíntese e respiração. O metal pode tanto ligar quanto orientar o substrato em sítios ativos de proteínas. O íon metálico faz com que a reação química catalisada em uma enzima possa ser realizada de uma maneira mais rápida, pois provê uma organização de cadeias laterais de grupos funcionais com grupos preferidos de enzimas (Medhavi Mallick e Shankaracharya, 2011).

Íons metálicos em proteínas são ligados por grupos de átomos que doam um par de elétrons para a ligação e geralmente tem carga neutra ou positiva, formando assim os sítios metálicos. Estes sítios podem ser descritos pelo tipo de metal que contém e pelo número de coordenação, que é o número de átomos ligados ao metal (Tainer et al., 1992). Átomos geralmente estão em uma distância bem próxima dos íons metálicos, como mostrado por Zheng et al. (2017), a distância entre o íon metálico e átomos de oxigênio ficam na faixa de 1,86 Å a 2,19 Å e a interação metal-nitrogênio tem uma distância entre 1,67 Å a 2,29

Å.

Metais podem ser divididos em duas categorias: elementos de transição (Mn, Fe, Co, Cu, Mo, W) e não transição (Na, K, Mg, Ca, Zn). Os metais de transição são caracterizados por um estado de oxidação (valência) variado e sua formação de íons com a camada D de elétrons preenchida. Já os metais não definidos como de transição são descritos por sua constância no estado de oxidação e formação de íons preenchida de elétrons de forma incompleta na camada P ou completamente preenchida na camada P (Denesyuk et al., 2020).

Em grande parte das metaloproteínas, o íon metálico implica em uma função biológica específica, a qual é relacionada a uma afinidade de ligação. Embora existam sítios metálicos que desempenhem outros papéis em proteínas (e alguns casos onde os sítios não exercem função alguma), as principais funções que um sítio metálico exerce em proteínas são (Banaszak, 2000):

- Estrutural: quando o íon metálico é necessário para a conformação de uma região da proteína. Em alguns casos servindo como função regulatória;
- Transporte e armazenamento: o íon metálico está ligado a uma proteína como meio de transporte e armazenamento;
- Transferência de elétrons: o íon serve como uma região redox para a transferência, armazenamento e captação de elétrons;
- Catalítica: o metal é necessário para a ligação ou ativação de um substrato e também para o desenvolvimento de estados de transição.

Metais de transição do quarto período geralmente desempenham um papel catalítico em metaloenzimas: (I) através de ligações ao substrato, fazendo com que se orientem de maneira adequada para a reação, (II) fazendo a mediação de reações de oxi-redução através de mudanças reversíveis no estado de oxidação do íon metálico ou (III) agindo na estabilização eletrostática ou blindagem de cargas negativas. Os elementos de metal de transição mais encontrados em estruturas proteicas são Mn, Fe, Co, Ni, Cu e Zn (Zheng et al., 2017). Os átomos ligantes geralmente observados em metais de transição em estruturas macromoleculares são o oxigênio (O), nitrogênio (N) e enxofre (S). Os aminoácidos ASP, GLU, HIS e CYS são aminoácidos que geralmente coordenam a maioria dos metais de transição do quarto período (Zheng et al., 2008).

Metais alcalinos (Li, Na, K, Rb, Cs e Fr) e metais alcalinos terrosos (Be, Mg, Ca, Sr, Ba e Ra) geralmente desempenham papel estrutural na proteína. Mas isso não é regra, pois por exemplo o Magnésio também atua de forma catalítica em várias enzimas, como as T4 ligase. Geralmente tem o número de coordenação 6 e geometria octaedral (Zheng et al., 2017).

Sítios metálicos em proteínas contém uma variedade de números de coordenação, geometria, íons metálicos e átomos que ligam aos metais, mas apesar disso compartilham de uma característica comum: estão centralizados em um local de ligantes hidrofílicos acompanhados de uma região que contém grupos de carbono (Yamashita et al., 1990).

A coordenação de íons metálicos geralmente é definida pelo maior número possível, e os átomos ligados ao metal são organizados de maneira com que haja a menor quantidade de interações repulsivas entre eles e o metal. A geometria de um sítio metálico depende de dois principais fatores: a quantidade de átomos (este que varia de acordo com o tamanho do metal e dos átomos da ligação) e seu arranjo estereoquímico (Tainer et al., 1992).

Os aminoácidos que compõem uma proteína seguem padrões bem definidos de comportamentos químicos. Sua estereoquímica leva a características de estruturas secundárias, como α-hélices e folhas-β, e juntamente com a sequência a estruturas terciárias (e em alguns casos quaternárias) permite catálise e outras funções na proteína (Harding et al., 2010). Grupos carboxilato de cadeia lateral (ASP e GLU), grupos tiol e tioéter (CYS e MET) e grupos hidroxila (SER, THR e TYR) são os aminoácidos que mais aparecem coordenando íons metálicos em proteínas, embora o oxigênio carbonílico de cadeia principal participa em alguns casos, particularmente os de cálcio (Ca) (Tainer et al., 1992). A Tabela 2.1 apresenta um resumo das propriedades dos 20 principais aminoácidos que compõem as proteínas.

Apesar de haver similaridades entre sítios metálicos de íons diferentes, existem também várias particularidades. Nas Subseções 2.1.1 a 2.1.6 são descritos os sítios metálicos de zinco, ferro, cálcio, magnésio, cobre e sódio respectivamente. Na subseção 2.1.7 são descritos outros íons metálicos.

2.1.1 Sítio metálico de zinco

Zinco (Zn) é um metal que está presente em diversos processos biológicos em seres vivos, sendo um componente crucial presente em proteínas estruturais, enzimas, fatores de transcrição e proteínas ribossômicas (Zhang e Zheng, 2020). Diferentemente de outros metais de transição, o Zn não realiza oxirredução e é um ácido de Lewis que polariza água, características que fazem com que este íon metálico seja um dos mais versáteis, estando presente em várias metaloenzimas de todas as classes de enzimas (Zheng et al., 2017).

Sítios metálicos de zinco podem desempenhar papel tanto catalítico quanto estrutural em proteínas. A característica catalítica de um sítio de Zn geralmente acontece quando o sítio está em contato com algum ligante externo (geralmente água) e contém três resíduos, sendo principalmente ASP, GLU e HIS (Harding et al., 2010; Zhao et al., 2011). A razão principal para que o zinco desempenhe papel catalítico são suas propriedades químicas bem distintas, que combinam a força de ácido de Lewis, falta de oxirredução e troca rápida de ligantes (Andreini et al., 2011).

Já um sítio de Zn é dito estrutural quando no arranjo íon metálico e átomos ligados

Tabela 2.1: Abreviaturas e propriedades padrão dos aminoácidos.

Aminoácidos	Símbolo	Abreviação	Polaridade	Carga elétrica (pH 7,4)	Ocorrência em proteínas(%)
Alanina	ALA	A	não polar	neutro	8,76
Arginina	ARG	R	básic polar	positiva	5,78
Asparagina	ASN	N	polar	neutro	3,93
Ácido aspártico	ASP	D	ácid polar	negativa	5,49
Cisteína	CYS	C	não polar	neutro	1,38
Ácido glutâmico	GLU	E	ácid polar	negativa	6,32
Glutamina	GLN	Q	polar	neutro	3,9
Glicina	GLY	G	não polar	neutro	7,03
Histidina	HIS	H	básic polar	positiva(10%) neutro(90%)	2,26
Isoleucina	ILE	I	não polar	neutro	5,49
Leucina	LEU	L	não polar	neutro	9,68
Lisina	LYS	K	básic polar	positiva	5,19
Metionina	MET	M	não polar	neutro	2,32
Fenilalanina	PHE	F	não polar	neutro	3,87
Prolina	PRO	P	não polar	neutro	5,02
Serina	SER	S	polar	neutro	7,14
Treonina	THR	T	polar	neutro	5,53
Triptofano	TRP	W	não polar	neutro	1,25
Tirosina	TYR	Y	polar	neutro	2,91
Valina	VAL	V	não polar	neutro	6,73

Fonte: Adaptada de Cooper e Hausman (2004) e Kozlowski (2016).

não há espaço suficiente para uma outra molécula, além disso o sítio contém geralmente quatro aminoácidos ligados, sendo os mais comuns HIS e CYS (Zhao et al., 2011). Em mais detalhes, em mais de 96% dos sítios de Zn estruturais pelo menos dois resíduos são CYS que são preferidas em relação a outros resíduos pelo fato de serem capazes de transferir carga negativa para o íon de Zn^{2+}, formando assim ligações mais fortes (Andreini et al., 2011).

2.1.2 Sítio metálico de ferro

Ferro (Fe) é o metal mais comum encontrado em proteínas de estrutura conhecida, sendo estas proteínas principalmente do grupo heme. Sítios metálicos de Fe estão envolvidos em transporte de O_2 ou processos de transferência de elétrons, onde o Fe(II) é coordenado por quatro átomos de N do anel de porfirina do grupo heme, e um átomo de N da histidina da cadeia de proteína (globina) (Tainer et al., 1992). Os aminoácidos doadores mais comuns encontrados em sítios metálicos de Fe são HIS, ASP, GLU, CYS e TYR (Akcapinar e Sezerman, 2017).

Um grande número de sítios metálicos de Fe está localizado em clusters de Fe/S

(ferro/enxofre), como em sistemas redox, e alguns deles em sítios catiônicos com grupos de doadores de diversos aminoácidos de cadeia lateral, como as transferrinas, que transportam o Fe para a membrana celular. Em proteínas que contém ferro e enxofre, os átomos de Fe estão ligados ao S da cisteína e geralmente também a um íon sulfeto (S^{2-}), formando assim clusters como os de Fe_2S_2 ou Fe_4S_4 (Harding et al., 2010).

O grupo heme pertencente está embutido a proteínas de tal maneira que existe o acesso à molécula de O_2 a uma pequena cavidade (conhecida como "bolsa de óleo") que coordena o íon de Fe(II) na sexta posição do octaedro (Harding et al., 2010). As proteínas com Fe não pertencentes ao grupo heme contém íons de Fe(II) e Fe(III) que são encontrados principalmente em geometria octaédrica ou trigonal bipiramidal. Os clusters com um a quatro íons de ferro com geometria tetraédrica são ligados principalmente por cisteínas e ligantes inorgânicos de enxofre (Tainer et al., 1992).

2.1.3 Sítio metálico de cálcio

Cálcio (Ca) é o quinto elemento mais abundante no planeta e está presente em diversos organismos, processos e estruturas biológicas, sendo um componente essencial na biomineralização de ossos, dentes e conchas, por exemplo (McPhalen et al., 1991). O íon de Ca é considerado "pesado" e dessa forma tem preferência por átomos também "pesados" com baixa polarizabilidade, sendo o oxigênio o átomo de coordenação mais comum, seguido pelo nitrogênio (Denesyuk et al., 2020). Os aminoácidos mais encontrados em sítios metálicos de Ca são: ASP, GLU, ASN e GLY (Akcapinar e Sezerman, 2017).

Um dos aspectos mais interessantes de proteínas que contêm sítios metálicos de Ca é a variabilidade. Existem, pelo menos, 16 formas diferentes de dobrar uma cadeia polipeptídica em torno do íon de Ca^{2+}, e essa diversidade dos sítios de Ca^{2+} reflete a diversidade de funções realizadas pelo sítio em estruturas biológicas. Em nível intracelular, íons de cálcio desempenham diversos papéis como regulação metabólica, transmissão nervosa, concentração muscular, divisão e crescimento celular, além de papel catalítico e estabilizador (McPhalen et al., 1991).

Íons de Ca^{2+}, assim como os de Na^{+} e K^{+}, são coordenados principalmente por átomos de oxigênio de carga negativa, tendo uma interação puramente eletrostática. Íons de Ca^{2+} têm um número de coordenação alto comparado a outros íons metálicos, sendo os valores de sete a nove os mais comuns para tais números. O raio de coordenação do Ca é relativamente grande, tendo distância entre o íon central e átomos de oxigênio entre 2,3 Å a 2,6 Å (Denesyuk et al., 2020).

2.1.4 Sítio metálico de magnésio

Magnésio (Mg) é um dos mais versáteis cofatores na bioquímica celular, exercendo funções catalíticas e estruturais em ambientes intra e intercelular. Pode-se citar algumas

das principais funções exercidas pelo íon de Mg em uma proteína: estabilização da estrutura da proteína e membranas biológicas, além de atuar na ativação de enzimas que regulam a bioquímica de ácidos nucleicos (Dudev e Lim, 2007).

Íons de Mg, assim como os de Ca, são chamados de "pesados", pois tem preferência por átomos também "pesados", além de também ter preferência por ligantes com baixa polarizabilidade e possuir raio iônico pequeno. Em contraste com íons de Ca, que geralmente tem um número de átomos doadores de quatro até nove, íons de Mg frequentemente têm apenas um ou dois (Tainer et al., 1992).

Em um sítio metálico, o íon de Mg tende a ligar diretamente com a cadeia lateral dos aminoácidos ASP e GLU ou ASN e GLN (Dudev e Lim, 2007). As restantes posições de coordenação do íon (que tem preferência pelo número de coordenação seis, octaedral) geralmente são preenchidas por moléculas de água juntamente com átomos de oxigênio ou outras pequenas moléculas, como ADP e ATP (Harding et al., 2010). É interessante destacar que o íon de manganês (Mn^{2+}) pode substituir o íon de magnésio (Mg^{2+}) em sítios ativos de várias enzimas (Khrustalev et al., 2016).

2.1.5 Sítio metálico de cobre

Cobre (Cu) é um metal importante na ativação de diversas enzimas que participam de processos biológicos fundamentais como a respiração e a fotossíntese e é encontrado em proteínas presentes em organismos procariotos e eucariotos (Zhang e Zheng, 2020). Sítios metálicos de Cu geralmente desempenham papéis de transferência de elétrons (citocromos e plastocianinas, por exemplo) e catalítico (superóxido dismutase e galactose oxidase, por exemplo) em proteínas (Tainer et al., 1992).

O íon de cobre é geralmente associado a reações de oxidação e ambos Cu(I) e Cu(II) formam complexos com diferentes grupos doadores. O cobre no estado de oxidação 2 é normalmente de geometria quadrada planar, enquanto no estado de oxidação 1 é geralmente de geometria tetraédrica (Harding et al., 2010).

Sítios metálicos de cobre em estruturas macromoleculares geralmente têm o número de coordenação igual a quatro, com geometria tetraédrica, principalmente quando está ligado com átomo de enxofre (S). Os aminoácidos mais comuns em sítios metálicos de Cu são a cisteína, com distância entre o átomo de S e o íon de Cu entre 2,2 Å a 2,3 Å, e metionina, tendo 2,5 Å de distância máxima entre o átomo de S e o íon de Cu (Zheng et al., 2017).

2.1.6 Sítio metálico de sódio

Sódio (Na) é um dos elementos mais abundantes no planeta, estando relacionado a processos biofísicos e fisiológicos em diversos seres vivos. Os sítios metálicos de Na podem desempenhar papel catalítico, estrutural e transporte em proteínas. Por ter algumas características bem similares, o íon de sódio muitas vezes é modelado, de maneira errada,

como uma molécula de água ou um íon de magnésio, quando se leva em consideração apenas um mapeamento de densidade de elétrons. Íons de magnésio carregam mais carga que os de sódio e assim precisam estar em volta a resíduos mais ácidos, como ASP e GLU (Zheng et al., 2017).

Íons de sódio são predominantemente ligados por átomos de oxigênio a uma distância média de 2,42 Å e seu número de coordenação é geralmente seis. A geometria de sítios metálicos de sódio é variada, podendo ser encontrados na forma tetraédrica, bipiramidal ou octaédrica. Pelo fato de que átomos da cadeia principal de resíduos do sítio metálico de sódio estarem envolvidos na coordenação do íon Na^+, a predição de sítios de Na em proteínas de estrutura desconhecida torna-se um desafio (Lev et al., 2013).

2.1.7 Outros sítios metálicos

Segundo dados do MetalPDB (Putignano et al., 2017), dos 64 tipos diferentes de metais que existem em estruturas catalogadas no PDB (Berman et al., 2000), os 5 íons mais comuns correspondem a mais de 86% das ocorrências, fazendo com que a maior parte dos íons metálicos tenham pouca representatividade em relação a sítios metálicos em proteínas.

Ainda sim é válido mencionar alguns tipos de sítios metálicos, como os de potássio (K), íon que tem o maior raio iônico entre todos os metais, geralmente encontrados em proteínas, com distâncias entre o potássio e oxigênio maiores do que 2,7 Å (Zheng et al., 2017). Íons de potássio geralmente estão ligados a funções de estabilização de proteínas, homeostase celular e atividade elétrica. No aspecto humano, mutações que afetam sítios metálicos de potássio estão relacionadas a algumas doenças como os cânceres de mama e pâncreas, doenças cardiovasculares e Alzheimer (Durdagi et al., 2013).

Níquel (Ni) é um metal essencial a diversas formas biológicas. Enzimas utilizam o níquel para realizar uma série de funções como hidrólises e reações redox. O íon de Ni é geralmente encontrado com número de coordenação igual a seis com configuração octaédrica (Boer e Hausinger, 2013).

O cobalto (Co) tem características similares ao íon de ferro, porém aparecendo em menor quantidade em sistemas biológicos. Existem exemplos onde o íon de cobalto pode, inclusive, substituir o íon de ferro, como em anéis de porfirinas. O íon de cobalto geralmente aparece com geometria octaédrica e tem número de coordenação igual a seis (Zheng et al., 2017).

Manganês (Mn) é um metal de transição do quarto período comumente encontrado em metaloproteínas na forma Mn^{2+} e suas propriedades são similares ao Zn^{2+} e Mg^{2+}. Existem exemplos de enzimas com esse íon metálico em todas as classes de enzimas (Harding et al., 2010). De acordo com Lu et al. (2012) os aminoácidos preferencialmente pertencentes aos sítios de Mn são: ASP, HIS e GLU.

2.2 Busca de sítios metálicos similares

Atualmente há um crescente número de estruturas e sequências de proteínas depositadas em banco de dados especializados, sendo provido principalmente pelo avanço de tecnologias de sequenciamento genômico e métodos de determinação de estruturas. Aumentou também a necessidade de abordagens que sejam capazes de predizer as funções biológicas de tais proteínas (Akcapinar e Sezerman, 2017).

Alinhado ao fato de que métodos experimentais ainda são desafiadores, por conta de problemas relacionados a custo, tempo e automação de processos, a demanda por técnicas computacionais cresce ainda mais. Essas técnicas criadas vêm sendo capazes de buscar e predizer sítios metálicos, bem como sua geometria, função e várias outras informações que podem ajudar diversos aspectos de pesquisa na área (Medhavi Mallick e Shankaracharya, 2011). Vários métodos para busca e identificação de sítios metálicos foram propostos e podem ser agrupados em três categorias principais: algoritmos baseados em sequência, em estrutura ou híbridos.

Os algoritmos baseados em sequência utilizam múltiplos alinhamentos de vários organismos que já foram estudados para verificar a conservação de resíduos que podem ser importantes tanto no sentido funcional como no estrutural. Como exemplo desses algoritmos, (Passerini et al., 2011) Passerini et al. (2012) propõe a criação de uma ferramenta, o MetalDetector, que utiliza apenas informações de sequência de proteínas para a determinação de coordenadas dos sítios de metais de transição ligados a CYS e HIS. A busca dos sítios metálicos é guiada por uma função que pontua as estruturas candidatas que são construídas através de subestruturas geradas a partir de uma busca gulosa. O algoritmo utiliza Support Vector Machine (SVM), cadeia oculta de Markov e grafos para fazer a modelagem e solução do problema de predição. A base de dados é formada por estruturas não redundantes retiradas do Protein Data Bank (PDB)(Berman et al., 2000) pelo UniqueProt (Mika e Rost, 2003).

Foi desenvolvido por Qiao e Xie (2019) o algoritmo MIonSite para a predição de sítios metálicos para ligantes específicos, a partir da sequência da proteína. O MIonSite emprega informações evolutivas das proteínas, estrutura secundária prevista, acessibilidade ao solvente e informações de conservação para extrair a característica discriminativa de cada resíduo. O algoritmo AdaBoost (Schapire, 1999) é aprimorado e projetado para lidar com o problema de desequilíbrio oculto na previsão do local de ligação de íons metálicos, onde o número de locais de não ligação é muito maior do que o dos locais de ligação de íons metálicos. Um dataset foi criado para testes e validações e o mesmo continha metaloproteínas com os seguintes íons metálicos: Zn^{2+}, Ca^{2+}, Mg^{2+}, Mn^{2+}, Fe^{3+}, Cu^{2+}, Fe^{2+}, Co^{2+}, Na^{+}, K^{+}, Cd^{2+} e Ni^{2+}.

Cao et al. (2017) apresenta um método de identificação de resíduos ligados ao íon metálico em uma proteína. O algoritmo utiliza SVM e uma variação de Position Weight Matrix, PWSM, para analisar a sequência através de uma janela deslizante de tamanho

variável. A ferramenta trabalha com 10 tipos de íons metálicos diferentes (Zn^{2+}, Cu^{2+}, Fe^{2+}, Fe^{3+}, Ca^{2+}, Mg^{2+}, Mn^{2+}, Na^{+}, K^{+} e Co^{2+}) e utilizou sequências que continham identidade inferior a 95% obtidas no BioLiP Database (Yang et al., 2012).

Hu et al. (2016) traz os algoritmos IonCom e IonSeq. Ambos consistem em métodos de predição de sítios de pequenos ligantes (metais e ácidos). O IonCom utiliza informações de sequência de proteínas, baseando a predição em *templates* e trabalha com informações que envolvem propriedades estruturais locais, aspectos de conservação de íons e propensões específicas de cada sítio. Já o IonSeq (que assim como o IonCom também utiliza informações de sequência de proteínas) utiliza uma janela deslizante para extrair diversas características da sequência. Visando o desbalanceamento de classes que ocorre em problemas de predição de sítios metálicos, os autores utilizam o algoritmo AdaBoost para gerar dados de treinamento e teste. Ambos os métodos são capaz de realizar a predição dos sítios metálicos de Zn^{2+}, Cu^{2+}, Fe^{2+}, Fe^{3+}, Ca^{2+}, Mg^{2+}, Mn^{2+}, Na^{+}, e K^{+}.

Levy et al. (2009) aborda um estudo que analisa se modelos estruturais baseados em homologia remota são eficazes para prever sítios metálicos em 3 dimensões baseando-se apenas em sequências de proteínas. O trabalho descreve o algoritmo SeqCHED (variação do algoritmo CHED que realiza a predição de sítios metálicos baseado em estrutura) que checa a sequência da proteína alvo com as sequências de modelo do PDB e então faz a modelagem da cadeia lateral em 3D na cadeia principal do modelo selecionado. A base de dados utilizada para realização de testes é formada de proteínas retiradas do PDB que não tinham mais do que 30% de similaridade de sequência e com resoluções melhores que 2,5 Å.

O TargetS (Yu et al., 2013) é um algoritmo que realiza a predição de sítios metálicos baseado na sequência de proteínas. O método realiza a predição a partir de características provenientes da matriz de conservação e matriz da estrutura secundária da proteína e propensões de ligantes específicos, utilizando a técnica SVM. O TargetS trabalha em duas etapas: primeiramente busca os resíduos dos sítios metálicos através de módulos de predição e depois os agrupa de acordo com a estrutura 3D da proteína.

Uma outra forma de se representar uma proteína é através de sua visualização tridimensional. Para buscar os sítios metálicos é necessário algoritmos que realizam uma busca estrutural, trabalhando assim com informações espaciais. Lin et al. (2016) é um exemplo de trabalho que utiliza informações de estruturas de proteínas ao descrever o algoritmo MIB, que busca sítios metálicos em 12 tipos de metais diferentes: Zn^{2+}, Ca^{2+}, Mg^{2+}, Mn^{2+}, Fe^{3+}, Cu^{2+}, Fe^{2+}, Co^{2+}, Cd^{2+}, Ni^{2+}, Hg^{2+} e Cu^{+}. A ferramenta prevê não somente a região dos resíduos ligados ao metal como o próprio local de encaixe do íon metálico. Os *templates* utilizados para a busca consistem em resíduos com distâncias de até 3,5 Å. Foi considerado como um sítio metálico as estruturas que continham 1 íon metálico e pelo menos 2 resíduos. Para a realização de treinamento e testes foram retiradas estruturas do PDB que continham pelo menos um íon metálico dos 12 abordados no artigo e depois retiradas as proteínas que continham uma similaridade superior a 30%.

He et al. (2015) aborda o algoritmo mFASD, que compara dois sítios metálicos usando a métrica FASD, a qual avalia a similaridade dos átomos em contato com o íon metálico em relação ao ambiente químico deste local. Átomos a uma distância R do íon metálico são selecionados a fazerem parte do FAS (functional atom set). A distância escolhida para a composição de um FAS foi de 3.5 Å e os vizinhos de cada átomo do FAS que estavam em até 5.0 Å também foram utilizados para o cálculo. Todo esse ambiente foi avaliado de acordo com as distâncias e propriedades químicas. A base de dados utilizada foi a Base PDBSelect25 (Griep e Hobohm, 2009) que contém estruturas do PDB com até 25% de similaridade. Foram mantidas estruturas que continham pelo menos uma ocorrência de cada um dos seguinte metais: Cu, Fe, Mg, Mn, Zn e Ca.

Sciortino et al. (2019) propõe uma alteração no algoritmo GaudiMM (uma ferramenta de modelagem molecular multi-uso) para a predição de sítios metálicos. Não trabalha com *templates* estruturais extraídos de bases de dados, mas sim com regras de coordenação. Cada iteração do algoritmo genético trabalha em dois passos: exploração e avaliação, que basicamente é a aplicação da função *fitness* nos indivíduos da população e a seleção desses indivíduos para a próxima geração. Para testes foram utilizadas 105 estruturas de alta qualidade adquiridas por raio-X, via MetalPDB web server (Putignano et al., 2017).

Ainda se tratando sobre algoritmos que utilizam a estrutura de proteínas para realizar buscar e predição, Brylinski e Skolnick (2011) faz uma adaptação do algoritmo FINDSITE, que originalmente faz a busca por sítios ativos em proteínas, para fazer a predição de sítios metálicos. A ferramenta trabalha com proteínas homólogas fracas e utiliza *templates* relacionados a distâncias. O FINDSITE trabalha com SVM e um classificador Bayesiano para calcular as distâncias entre o modelo e a estrutura nativa. A base de dados utilizada contém sítios com pelo menos um dos seguintes íons metálicos: Ca, Co, Cu, Fe, Mg, Mn, Ni, e Zn. Estruturas redundantes foram removidas, totalizando 860 proteínas que foram obtidas no PDB.

Por fim, tem-se os algoritmos que são a combinação dos baseados em estrutura e os baseados em sequência, os híbridos. Os algoritmos híbridos trabalham com um número maior de informações e geralmente utilizam as duas formas de busca de informações em conjunto, para assim gerar os resultados. Song et al. (2017) descreve o algoritmo MetalExplorer que utiliza machine learning para predição de sítios metálicos. O funcionamento da ferramenta consiste em quatro etapas: preparo da base de dados, extração de features, seleção de features e avaliação de desempenho. O MetalExplorer utiliza a técnica de floresta aleatória e trabalha com oito tipos de íons metálicos, sendo eles: Ca, Co, Cu, Fe, Ni, Mg, Mn, e Zn. A base de dados utilizada para testes e avaliação foi criada a partir de estruturas não redundantes já conhecidas do PDB.

Yang et al. (2013) propõe dois métodos de predição de sítios de ligação: TM-SITE e S-SITE. O primeiro utiliza informações da estrutura da proteína e através de *templates* compara o alinhamento de estruturas juntamente com análises do ambiente do sítio para a predição. Já o S-SITE utiliza informações da sequência de proteínas que também utiliza

o alinhamento (agora de sequências) e templates. Os autores utilizam a base de dados BioLiP para a obtenção de informações das proteínas utilizadas nos treinos e testes.

O COACH (Yang et al., 2013) é uma abordagem que utiliza vários métodos de predição de maneira conjunta a fim de encontrar sítios de ligação em proteínas. O método utiliza os métodos TM-SITE e S-SITE para primeiramente gerar predições de resíduos do sítio que depois são verificadas de acordo com a base de dados BioLiP, através de comparações de informações de sequência e estrutura. Depois, outros três métodos (COFACTOR, FINDSITE e ConCavity) são utilizados para refinar os resultados das predições.

Ajitha et al. (2018) propõe a criação do METAL ACTIVE INTERACTION e do ZINC-CLUSTER. O primeiro é um banco de dados que armazena informações de estruturas do PDB que contém íons metálicos (49 tipos de metal). Já o segundo é uma ferramenta de predição de sítios metálicos de zinco, baseando-se tanto na estrutura 3D da proteína como também em sua estrutura primária, a sequência. O algoritmo ZINCCLUSTER trabalha de duas formas: identifica os resíduos associados aos metais, em estruturas já conhecidas, ou encontra grupos de sítios metálicos a partir da sequência primária da proteína. O ZINCCLUSTER utiliza SVM para a predição de sítios e trabalha com proteínas obtidas no PDB.

Apesar de utilizarem abordagens muitas vezes bem diferentes e que as próprias informações sobre as metaloproteínas (sequência ou estrutura) serem bem particulares, ainda sim existem certas maneiras em comum de trabalharem.

Como visto anteriormente, uma grande parte dos algoritmos para busca e predição de sítios metálicos utiliza técnicas de Inteligência Artificial (IA) em sua implementação. Pode-se observar diversos trabalhos que utilizam algumas destas técnicas de IA, como redes neurais em Lippi et al. (2012); Passerini et al. (2011), algoritmos evolutivos em Sciortino et al. (2019) e principalmente o uso de Support Vector Machine (SVM), em Brylinski e Skolnick (2011); Cao et al. (2017); Lippi et al. (2012). Encontra-se também, na modelagem dos algoritmos de busca e predição, modelos estatísticos empregados à técnicas computacionais, como, por exemplo, o Modelo Oculto de Markov (Passerini et al., 2012; Andreini et al., 2009). Outros trabalhos também fazem a modelagem do problema usando grafos (He et al., 2015; Passerini et al., 2011) e realizam buscas e predições nesse tipo de estrutura.

Sobre a forma em que foram montados as bases para a realização de testes e treinamentos de ferramentas de busca e predição de sítios metálicos, pode-se citar algumas semelhanças. A primeira delas diz respeito à semelhança entre as macromoléculas que são selecionadas. Geralmente os autores trabalham com proteínas que contém pouca similaridade entre si, com percentuais variando em torno de 30% de similaridade máxima entre elas (Levy et al., 2009; Lin et al., 2016). Os trabalhos tendem a utilizar estruturas com um certo grau de resolução, que variam em torno de 1,5 Å a 2 Å (Ångström), como visto em Levy et al. (2009) e Zheng et al. (2017).

De forma geral, muitas ferramentas utilizam tanto informações sobre as distâncias dos

resíduos ligados ao metal (que juntamente com o íon metálico formam o sítio metálico), como também de resíduos vizinhos aos sítios. A maneira como os algoritmos realizam as buscas é peculiar de cada abordagem, mas sabe-se também que certas ferramentas utilizam propriedades físico-químicas para ajudar a encontrar sítios metálicos de uma maneira mais precisa além da maneira padrão de calcular as distâncias entre os resíduos que compõe a região (He et al., 2015).

Outro ponto sobre a forma que os métodos computacionais realizam a busca e identificação dos sítios metálicos é em relação aos algoritmos baseados em sequência, que utilizam uma janela deslizante de tamanho variável para percorrer toda a sequência da metaloproteína e assim identificar as partes que compõem o sítio metálico, como visto em Cao et al. (2017).

2.3 Bases de dados

Bases de dados são coleções sistemáticas que suportam armazenamento e gerenciamento de dados. Elas armazenam grandes quantidades de informação e possibilitam consultas rápidas e facilitadas a diversos tipos de itens e documentos. Em Bioinformática existem diversas bases de dados para diferentes tipos de contexto, como por exemplo o PDB, que traz informações sobre macromoléculas biológicas em geral, o MetalPDB, que atua trazendo informações envolvendo moléculas e íons de metal, e o M-CSA que trabalha com sítios catalíticos. A seguir, são descritas com mais detalhes as bases de dados utilizadas neste trabalho.

2.3.1 Protein Data Bank (PDB)

O Protein Data Bank [1](PDB) é um repositório que contém dados estruturais de macromoléculas biológicas, criado no Brookhaven National Laboratories, Upton, EUA, em 1971. Foi a primeira base de dados de acesso livre de história da medicina e biologia e atualmente abriga mais de 173 mil estruturas de macromoléculas biológicas. Os dados das macromoléculas depositados no PDB são extraídos a partir de técnicas como raio-x, ressonância magnética nuclear, microscopia eletrônica por criogenia e também de modelos teóricos (Berman et al., 2000).

Existem quatro principais serviços prestados pelo PDB: depósito de dados, gerenciamento de arquivo e integração, entrega de dados e exploração, e divulgação e educação (Burley et al., 2020). Para ajudar no funcionamento de todos os processos do repositório, diversos parceiros, colaboradores e plataformas estão envolvidos, como o Worldwide Protein Data Bank (wwPDB), que atua na área de depósito, validação e acurácia das biomoléculas catalogadas, e também o PDB101, um portal online para professores, estudantes e público em geral explorar o mundo das macromoléculas biológicas.

[1] `https://www.rcsb.org`

2.3.2 MetalPDB

O MetalPDB[2] é uma base de dados que provê informações de sítios metálicos de ligação em estruturas tridimensionais de macromoléculas biológicas. A base de dados é voltada para o estudo de metais na biologia e é composta por *templates* que representam os sítios metálicos e descrevem o ambiente em que o íon metálico se encontra.

Os sítios metálicos depositados no MetalPDB são estruturas na forma de Sítios Funcionais Mínimos (MFS, em inglês), onde cada MFS é a junção de átomos do cofator de metal, dos ligantes de metal e outros resíduos ou espécies químicas a uma distância de até 5 Å de um ligante. O conceito de MFSs é importante pois pode prover informações sobre as funções ou mecanismos de ação de uma metaloproteína, além de ser útil para predizer a função de estruturas 3D na ausência de dados bioquímicos experimentais (Putignano et al., 2017).

Atualmente o MetalPDB armazena informações sobre mais de 300 mil sítios metálicos de diferentes íons em mais de 55 mil estruturas providas pelo PDB. A base de dados conta ainda com informações sobre funções, geometria e domínios metálicos.

2.3.3 Mechanism and Catalytic Site Atlas (M-CSA)

O Mechanism and Catalytic Site Atlas[3] (M-CSA) é uma base de dados que documenta sítios ativos de enzimas e resíduos catalíticos em enzimas de estruturas 3D. Foi criado com o intuito de prover anotações curadas de um pequeno número de resíduos altamente conservados e que estão diretamente relacionados com atividades enzimáticas de estruturas depositadas no PDB (Furnham et al., 2013).

Os dados curados que o M-CSA disponibiliza são fontes de informação para construir *templates* 3D de sítios catalíticos e podem ser usados, por exemplo, para buscar e identificar novas estruturas. Em sua versão mais recente (M-CSA 2.0), o M-CSA conta com 968 estruturas de enzimas curadas e anotadas e utiliza um método de confiabilidade que extrapola as anotações e identificação para estruturas homólogas dos resíduos catalíticos (Ribeiro et al., 2017).

2.3.4 BioLiP

O BioLiP[4] é uma base de dados curada semi-manualmente e que contém informações biológicas de interações proteína-ligante (Yang et al., 2012). Cada entrada do BioLiP compreende por anotações como resíduos catalíticos, resíduos de ligação, afinidades dos resíduos, termos da ontologia genética, dentre outras informações. Para a obtenção dos dados da base, o BioLiP utiliza o algoritmo COACH (Yang et al., 2013) para a predição de sítios de ligação.

[2] `https://metalpdb.cerm.unifi.it`

[3] `https://www.ebi.ac.uk/thornton-srv/m-csa/`

[4] `http://zhanglab.ccmb.med.umich.edu/BioLiP/`

Para avaliar a relevância biológica dos ligantes, o BioLiP filtra cada ligante candidato em quatro passos. O primeiro deles consiste em verificar se o ligante candidato está na lista de artefatos e aparece mais do que 15 vezes na estrutura, caso ambos sejam verdade, o ligante candidato é considerado irrelevante. O segundo passo consiste em calcular a distância entre o ligante e o resíduo. Se essa distância for menor do que a soma dos raios de Van der Waals dos átomos envolvidos somado a uma tolerância de 0.5 Å considera-se o ligante como válido. O terceiro passo verifica se o ligante, agora dentro da distância definida, não está na lista de artefatos. Tendo passado pelos três filtros anteriores, o quarto passo utiliza *abstracts* do PubMed para filtrar ligantes irrelevantes.

Atualmente, o BioLiP conta com 529.047 entradas totais, sendo elas compostas por mais de 109 mil proteínas originadas do PDB e mais de 146 mil resíduos ligados a metais. É possível, por meio de seu endereço eletrônico, realizar buscas de ligantes e proteínas, além de fazer o download das informações contidas na base de dados.

2.4 Algoritmos genéticos

Algoritmos genéticos (AG) são métodos meta-heurísticos baseados na teoria de seleção natural de Charles Darwin e foram inicialmente propostos por J. H. Holland em 1992. Os elementos básicos de um AG são: indivíduos, seleção por aptidão, e operadores genéticos (mutação e cruzamento) (Katoch et al., 2020).

Um AG tem início com a definição de uma população de indivíduos (população inicial) que representam possíveis soluções para um determinado problema. Os indivíduos são então selecionados com base em uma função de aptidão (fitness). Em seguida, operadores genéticos (cruzamento e mutação) são aplicados nos indivíduos selecionados a fim de produzir uma nova população. Este processo iterativo (geração) é realizado até que uma condição de parada seja satisfeita, podendo ser um número definido de gerações, uma detecção de convergência ou tempo de execução do AG (Izidoro et al., 2014).

AGs são técnicas de otimização estocástica que não garantem a solução ótima do problema todas as vezes em que são executados, se tratando assim de um processo não-determinístico. Apesar disso, existem vários problemas associados à utilização de AGs, como por exemplo problemas que envolvem o tamanho e a diversidade de populações, e maldição de dimensionalidade (Umbarkar e Joshi, 2013).

Existem diversos contextos em que AGs podem ser utilizados, e a Bioinformática é um exemplo. Otovic et al. (2020) desenvolveram um AG para busca em espaços químicos de pequenos peptídeos. O trabalho permitiu a definição de bibliotecas de peptídeos capazes de cobrir uma grande área do espaço de pesquisa de novos peptídeos ativos.

Liu e Tao (2008) propôs a utilização de AG para a predição de estrutura de proteínas baseando-se em sua sequência. Os autores utilizaram valores hidrofóbicos de proteínas como um modelo de otimização matemático e um AG foi utilizado para resolver o problema de otimização.

AGs também podem ser utilizados em conjunto com outras técnicas computacionais, como as redes neurais profundas, ou DNN (*Deep Neural Networks*). No trabalho de Shukla e Singh (2021), um AG foi utilizado como ferramenta de otimização, encontrando a melhor arquitetura de uma DNN. O processo incluiu a definição da quantidade de camadas, nós por camadas e as funções utilizadas em cada camada.

Uma outra maneira de se utilizar AGs foi descrita por Morris et al. (1998), ao trabalhar juntamente com a técnica de SVM (*Support Vector Machine*). A acurácia de problemas de classificação utilizando SVM depende de uma boa definição de parâmetros, e nesse trabalho, um AG foi utilizado para a seleção de features e otimização de parâmetros.

A seguir, são descritos com mais detalhes cada componente de um AG, bem como a sua execução. Os tópicos abordados são: indivíduo e população, função fitness, seleção e operadores genéticos, parâmetros, complexidade, condição de parada e a abordagem paralela de AGs.

2.4.1 Representação do indivíduo e população

A representação de um indivíduo corresponde à primeira etapa da modelagem de um AG. Um indivíduo de um AG é uma abstração de um indivíduo do mundo real. A definição de um indivíduo envolve simplificar aspectos do mundo real e representa uma possível solução para o problema em questão. Os indivíduos devem ser criados de tal forma que possam ser avaliados, selecionados e manipulados pelos operadores genéticos, e geralmente são definidos por especialistas na área (Eiben e Smith, 2007).

Um indivíduo, ou solução candidata, pode ser representado de acordo com alguns modelos descritos por Eiben e Smith (2007): binário, inteiro, real e permutação. A representação binária é a mais simples, onde um indivíduo consiste em uma simples string binária de dígitos, ou bit-string. O tamanho da string vai depender do contexto do problema e como se dará o mapeamento do indivíduo do mundo real para o indivíduo do AG. Um problema desta abordagem é que diferentes bits tem diferentes significados e uma simples alteração em um dos bits (através de uma mutação, por exemplo), pode trazer resultados muito variados.

A representação inteira é uma maneira de definir indivíduos de um AG quando o problema naturalmente mapeia diferentes genes (características de um indivíduo) em um elemento de um conjunto. Um exemplo desta representação é o problema de se encontrar valores ótimos de um conjunto de variáveis que tem valores inteiros.

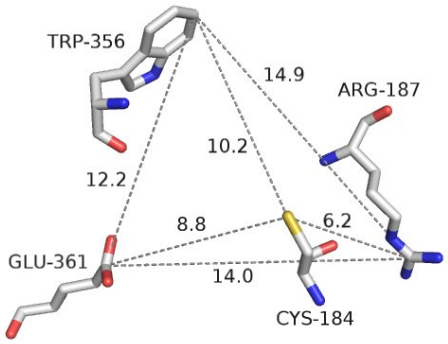
Uma outra maneira de representar indivíduos de um AG é através de valores reais ou de ponto flutuante. Essa forma consiste em usar números reais para compor a string e é utilizada para representar genes com valores contínuos, e não mais discretos como na representação inteira. É útil para descrever valores de altura, distâncias ou peso, por exemplo.

A representação de permutação é útil para problemas que envolvem ordenação, como ordenação de tarefas ou o problema do caixeiro viajante. Nessa representação, cada indivíduo é formado por uma string de números que representam a sequência para a solução do problema.

Um conjunto de indivíduos forma a população de um AG. Essa população contém possíveis soluções para o problema e pode ser gerada de maneira aleatória ou através de sementes (*seeds*). Diaz-Gomez e Hougen (2007) cita uma série de fatores a serem levados em consideração ao definir uma população inicial gerada aleatoriamente: o espaço de busca, função *fitness*, diversidade, dificuldade do problema, seleção e o número de indivíduos. Populações iniciadas através de sementes geralmente são criadas a partir de pré-processamento feito com indivíduos aleatórios, onde os mesmos são avaliados de acordo com a função *fitness*. Aqueles mais bem avaliados irão compor a população inicial (Meadows et al., 2013).

Figura 2.1: Modelagem de um indivíduo utilizado pelo GASS. (a) Indivíduo do GASS - (b) Sítio catalítico da enzima 3NOS com as distâncias (em Angstroms) entre os últimos átomos mais pesados da cadeia lateral de cada resíduo.

(a)



(b)

Fonte: Adaptado de Izidoro et al. (2014)

Izidoro et al. (2014) define como o indivíduo utilizado em seu AG (GASS) um grupo de aminoácidos que compõem um sítio ativo candidato. Um indivíduo é codificado como um vetor onde cada posição contém informações referentes a um aminoácido, como seu nome, posição na sequência e coordenadas do átomo mais pesado no espaço tridimensional. A Figura 2.1 mostra a modelagem do indivíduo utilizado pelo GASS.

Uma outra modelagem de indivíduo de um AG é mostrada por Morris et al. (1998), onde os autores buscam prever a conformação de ligantes a macromoléculas. Os indi-

víduos são mapeados em ligantes de macromoléculas como uma string, que contém as informações de coordenadas no espaço tridimensional para a tradução do ligante, variáveis que especificam a orientação do ligante e também um valor real para cada torção do ligante.

2.4.2 Função de avaliação (*fitness*)

Uma função de avaliação ou função *fitness* deve ser capaz de representar os requisitos necessários a que uma população deve se adaptar a fim de avançar para a geração seguinte (Eiben e Smith, 2007). Todos os indivíduos da população de cada geração do AG são avaliados pela função *fitness*.

É importante que a função *fitness* seja representativa e possa diferenciar com precisão os indivíduos (soluções) bons dos ruins. Uma função *fitness* não ajustada na avaliação de indivíduos pode acabar descartando um indivíduo promissor, que poderia ajudar a encontrar soluções melhores para o problema, além do fato de consumir recursos em indivíduos que agregam pouco no desenvolvimento do AG.

Funções *fitness* multi-objetivas levam em consideração diversos aspectos do problema ao avaliar indivíduos, e podem tratá-los de maneira igualitária ou dando pesos diferentes para cada um. Izidoro et al. (2015) utiliza uma função *fitness* multi-objetiva ao combinar características de profundidade juntamente com distâncias de resíduos de aminoácidos para avaliar os indivíduos em problemas de busca de sítios ativos em proteínas.

2.4.3 Seleção e operadores genéticos

A pressão seletiva diz respeito à influência que o meio exerce sobre a seleção de indivíduos de um AG. Representa um conjunto de características impostas à população a fim de direcionar a evolução de determinadas características para se adaptarem a esse meio.

A pressão seletiva é dada pela distribuição de probabilidades dos indivíduos sobreviverem ao meio. Em um cenário onde as probabilidades dos indivíduos estão com uma distribuição uniforme, a pressão seletiva é baixa e assim todos os indivíduos têm probabilidades semelhantes de sobreviverem. No caso onde a distribuição das probabilidades é distinta, a pressão seletiva é alta, fazendo que os indivíduos com alta aptidão tenham mais chances de sobreviverem em relação àqueles com baixa aptidão (Camargo, 2006).

A próxima etapa de um AG, após a fase de avaliação de indivíduos (através da função *fitness*), é a de seleção, que diz como deverá ser feita a escolha de indivíduos que formarão descendentes para a próxima geração (Mitchell, 1998). O termo pressão seletiva é muitas vezes usado para mostrar quanto um método de seleção considera o valor de avaliação de indivíduos (Jonson e Petersen, 2001). O objetivo da seleção de um AG é destacar indivíduos mais aptos na população para que possam gerar descendentes ainda melhores.

Eiben e Smith (2007) citam diversas maneiras de realizar a seleção de indivíduos de um AG, como a seleção baseada no valor absoluto de *fitness*, a seleção por torneio ou seleção por *rankings*.

O princípio da seleção por *fitness* baseia-se em que os indivíduos são selecionados apenas de acordo com o valor absoluto em que a função *fitness* o avalia. Indivíduos mais aptos tendem a ocupar toda a população de maneira muito rápida, fazendo com que o processo de busca seja mais focado em uma região do espaço de busca específica. Dessa forma fica mais difícil para o AG cobrir todas as possíveis soluções do problema. Esse fenômeno é conhecido como convergência prematura (Eiben e Smith, 2007).

A seleção pelo método de torneio faz com que um número N de indivíduos aleatórios sejam selecionados. Depois, um número aleatório R é escolhido entre 0 e 1. Se $R < K$ (onde K é um parâmetro, como por exemplo 0,75) o indivíduo mais apto é selecionado, caso contrário o menos apto (Mitchell, 1998). Quanto maior é o valor de K, maior será a pressão seletiva imposta à população.

A seleção feita através do método de ranking classifica os indivíduos com base em sua *fitness* e depois são alocadas probabilidades de seleção de acordo com o ranqueamento (e não em relação ao valor de *fitness*) (Eiben e Smith, 2007). Essa abordagem evita que a maior parte da seleção seja feita por indivíduos mais aptos, reduzindo a pressão seletiva. É uma alternativa para evitar a convergência prematura (Mitchell, 1998).

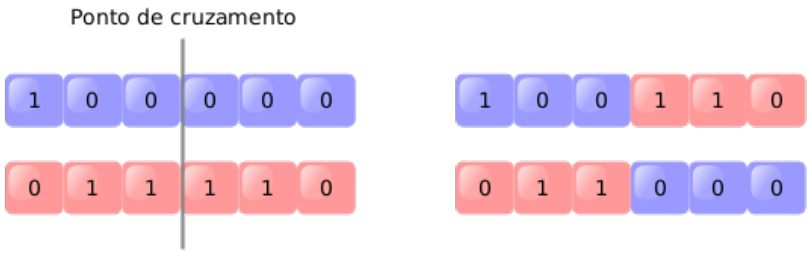
Existe também o método da roleta que consiste em uma maneira comum de seleção de indivíduos a partir de seu valor esperado de *fitness*. Para cada indivíduo é assegurado um pedaço da roleta, onde o tamanho do pedaço é proporcional ao valor fitness do indivíduo. A roleta então é girada N vezes, onde N é a quantidade de indivíduos da população, e ao final de cada vez em que a roleta é girada, o indivíduo marcado é selecionado para ser um pai da próxima geração (Mitchell, 1998).

Tendo sido feita a seleção de indivíduos, dois operadores genéticos são utilizados para gerar uma nova população (próxima geração) do AG: cruzamento e mutação. Esses operadores genéticos têm como finalidade refinar e espalhar a busca, trazendo também mais variabilidade genética.

O operador de cruzamento utiliza a combinação entre dois indivíduos (aqui definidos como pais) para gerar indivíduos descendentes (definidos como indivíduos filhos) (Dréo et al., 2006). Basicamente, a operação de cruzamento acontece quando dois indivíduos são selecionados e partes aleatórias destes são trocadas entre eles, formando assim novos indivíduos. Pode-se citar três principais formas de cruzamento: ponto simples (ou um ponto), multiponto (ou k-pontos) e uniforme.

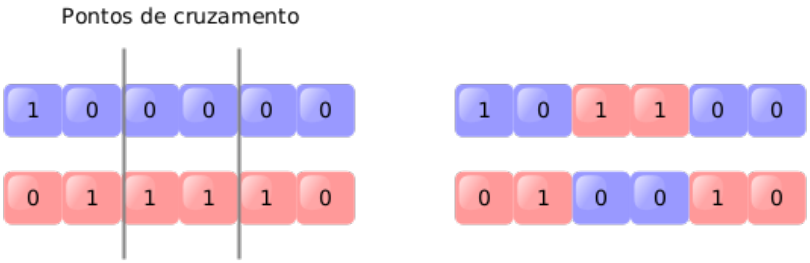
O cruzamento do tipo ponto simples seleciona dois indivíduos pais para o cruzamento e seleciona aleatoriamente um ponto Pi nesses indivíduos (onde $i >= 0$ e $i < n$, sendo n o tamanho do indivíduo) e então dois indivíduos filhos são criados pela combinação das partes criadas pela divisão dos indivíduos pais pelo ponto Pi (A. J. Umbarkar, 2015). A Figura 2.2 mostra o resultado do cruzamento do tipo um ponto.

Figura 2.2: Exemplo de um cruzamento do tipo um ponto.



O cruzamento multiponto atua de forma bem similar ao cruzamento de ponto simples, porém, nesse caso mais do que um ponto é criado. O método seleciona dois indivíduos pais e também seleciona aleatoriamente um valor de K, que determina os pontos $P1i$ a $Pk - 1i$ (onde $i >= 0$ e $i < n$, sendo n o tamanho do indivíduo) que serão os locais onde haverá o cruzamento (A. J. Umbarkar, 2015). A Figura 2.3 ilustra o cruzamento multiponto.

Figura 2.3: Exemplo de um cruzamento do tipo multiponto.



Por último, tem-se o cruzamento uniforme, que como o próprio nome diz, faz com que haja uniformidade na combinação das partes de ambos os indivíduos pais. Esse tipo de operador cria os indivíduos filhos a partir de um número real u (entre 0 e 1) e a partir daí cada gene dos indivíduos filhos decide de acordo com o número u se serão herdados do primeiro ou segundo pai (A. J. Umbarkar, 2015).

O operador de mutação ocorre alterando aleatoriamente algumas características genéticas de certos indivíduos que foram selecionados por um critério probabilístico (Goldberg et al., 1989). A mutação é uma operação que utiliza apenas um indivíduo pai para criar o indivíduo filho, aplicando algum tipo de modificação aleatória em sua representação (Eiben e Smith, 2007). Diversos tipos de mutação são descritos por Soni e Kumar (2014), como a mutação de inserção, de inversão e uniforme.

A mutação de inserção seleciona dois genes aleatórios do indivíduo e então move o primeiro gene para seguir o segundo, movendo todos os outros genes de acordo. Esse

tipo de mutação acaba por não modificar muito a ordem em que os genes aparecem e é utilizada em problemas de permutação.

Na mutação de inversão, dois genes aleatórios são escolhidos, e para realizar a operação, realiza-se a inversão de todos os genes entre os escolhidos. Isso faz com que seja preservada a informação adjacente entre os genes, porém, perde-se informação de ordem. Também é utilizado em problemas de permutação.

Já a mutação uniforme realiza a mudança de um gene aleatório de acordo com um valor específico em que esse gene pode assumir. Ou seja, um gene G escolhido para sofrer a mutação pode receber um valor i, onde i corresponde a um elemento do conjunto de valores que G pode assumir. Esse tipo de mutação é usado em casos de representação real e inteira de indivíduos.

2.4.4 Parâmetros

Os AGs possuem alguns parâmetros que impactam diretamente em seu funcionamento. Apesar de haver na literatura valores padrão a serem utilizados, a configuração de parâmetros é particular no contexto em que o AG está inserido (Izidoro et al., 2014). Alguns dos parâmetros utilizados nos AGs são:

- **Número de gerações**: é um dos critérios de parada de um AG. Ao executar um número muito pequeno de gerações um AG pode não encontrar uma resposta satisfatória. Por outro lado, um número de gerações muito grande pode impactar negativamente no tempo computacional gasto. Em Izidoro (2005) o número máximo de gerações do AG foi definido em 10 pois o objetivo era encontrar não apenas o máximo global de uma função, mas também os máximos locais. Como dito anteriormente sobre os parâmetros de um AG, em muitos casos, sua configuração depende diretamente do problema a ser resolvido.

- **Tamanho da população**: é a quantidade de indivíduos presentes em cada geração do AG. Pode ser estática, se mantendo a mesma durante toda a execução do algoritmo, ou pode sofrer alterações em seu tamanho de acordo com a execução. Populações maiores tendem a consumir mais tempo de execução enquanto populações menores podem acabar não cobrindo todo o espaço de busca do problema. Novamente é importante salientar que o ajuste dos parâmetros de um AG estão ligados diretamente ao contexto do problema a ser resolvido. Em Izidoro (2005) o tamanho da população foi definido em 6 indivíduos, pois o objetivo era encontrar não apenas o máximo global de uma função, mas também os máximos locais.

- **Probabilidade de Cruzamento**: é um percentual que indica a chance de um indivíduo trocar material genético com outro dentro da população a fim de gerar indivíduos descendentes. Tem como finalidade refinar a busca de indivíduos. AGs

com taxas de cruzamento altas tendem a inserir novas características mais rapidamente à população, porém pode-se acabar perdendo bons indivíduos que possam ser substituídos. O uso de operadores de cruzamento com probabilidade alta ou baixa depende do contexto do problema.

- **Probabilidade de Mutação**: determina a chance de um indivíduo sofrer alteração em sua(s) característica(s) e tem como finalidade evitar com que o AG fique preso em mínimos locais, sendo responsável por inserir diversidade à população. Geralmente possui taxas baixas, mas como acontece com o cruzamento, a utilização de taxas altas ou baixas depende do contexto.

- **Tamanho do Torneio**: é um parâmetro da seleção por torneio que controla a pressão seletiva, evitando uma convergência precoce do AG. O operador de torneio trabalha selecionando aleatoriamente N indivíduos da população e selecionando a melhor solução entre eles para seguir para a geração seguinte.

- **Tamanho do Ranking:** Número das melhores soluções encontradas após a execução do AG. Pode variar de um (1) até o tamanho da população.

2.4.5 Condição de parada

Existem duas principais formas para o término da execução de um AG (Eiben e Smith, 2007). A primeira é em relação às características dos indivíduos que compõem a solução do problema. Quando é possível identificar um padrão ótimo em relação aos indivíduos da população, não existe mais a necessidade de se continuar executando o AG, podendo assim encerrar sua execução.

A segunda forma de condição de parada de um AG acontece quando não se sabe identificar um padrão ótimo dos indivíduos. Pode-se citar alguns fatores que podem fazer com que um AG termine sua execução:

- Tempo máximo de execução do algoritmo ou número de gerações é excedido;

- Número total de avaliações feitas pela função fitness é alcançado;

- Melhorias em indivíduos feitas através de operadores genéticos e seleção já alcançaram um certo limite, não havendo mais mudanças;

- A densidade da população cai para um certo limite.

Geralmente, um AG tem sua execução terminada quando se satisfaz uma das formas descritas anteriormente: um certo valor ótimo (ou satisfatório) é alcançado pelos indivíduos ou uma condição de parada é satisfeita.

2.4.6 AGs paralelos

AGs são uma importante abordagem computacional para resolver diversos problemas de busca e otimização. Problemas que muitas vezes estão inseridos em contextos amplos e que por isso acabam requerendo uma grande quantidade de recursos computacionais. Existem casos onde executar AGs em máquinas em série pode levar dias ou até semanas até completar sua execução, e uma abordagem paralela pode trazer ganhos consideráveis em tempo de execução e utilização de recursos (Umbarkar e Joshi, 2013).

Computação paralela diz respeito a vários processos que trabalham simultaneamente a fim de resolver um determinado problema. O paralelismo funciona decompondo a carga de trabalho, ou tarefas, entre os vários recursos computacionais disponíveis, a fim de ter ganhos em relação a tempo e/ou melhora nos resultados. Abordagens de problemas que utilizam o paralelismo devem levar em consideração a comunicação entre os processos, pois muitas vezes apenas ajustar um problema serial não garante a melhor abordagem paralela (Madhuri e Deep, 2009).

Algoritmos evolutivos têm como característica implícita uma busca naturalmente paralela por uma solução. Isto se evidencia ao notar-se que cada indivíduo dentro de uma população busca por si só otimizar sua fitness (Cantu-Paz, 1998). É esta propriedade que permite em uma mesma população, com todos os indivíduos expostos aos mesmos operadores, o surgimento de soluções boas diversas. Este fato, atrela o conceito de Algoritmos Genéticos ao conceito de paralelização, indicando intuitivamente a ideia de um AG paralelo.

Tendo isso em mente, AGs paralelos trabalham, por exemplo, com problemas de multi-população, onde vários processos diferentes trabalham de maneira independente com suas respectivas populações e AG. Ao final de cada execução paralela ou até mesmo após algumas gerações, processos podem trocar mensagens entre si, compartilhando e integrando soluções (Majd et al., 2013).

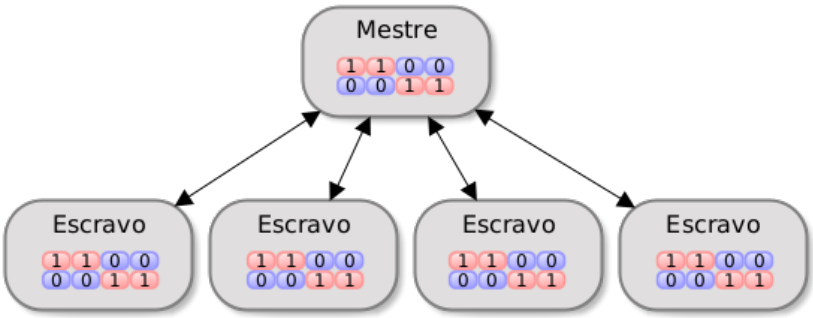
Embora AGs sequenciais têm mostrado sucesso em diversos contextos e problemas diferentes, existem casos onde essa abordagem não é o suficiente, sendo necessário partir para uma implementação paralela do AG. Pode-se citar como casos onde AGs paralelos são mais vantajosos (Nowostawski e Poli, 1999):

- Quando a população é demasiadamente grande;
- Quando a função fitness consome muitos recursos (tempo e/ou memória);
- Quando AGs sequenciais caem em regiões subótimas do espaço de busca.

Existem três classes gerais de AGs paralelos: mestre-escravo, granulado-grosso (ilha) e granulado-fino (célula) (Fauzi Mohd Johar et al., 2013). Os AGs paralelos do tipo mestre-escravo utilizam a mesma ideia de um AG sequencial e aplicam o paralelismo de maneira simples, onde não se altera nem restringe nenhum dos operadores genéticos. No

modelo mestre-escravo geralmente apenas a avaliação de indivíduos pela função fitness é feita de maneira paralela, todo o controle de gerações, seleção e operadores de mutação e cruzamento ainda são feitos de forma serial. Este tipo de implementação é buscada principalmente quando a análise da fitness é complexa e consome muita carga computacional. Uma desvantagem desse método é que muitas vezes o processo mestre fica ocioso, esperando pelos outros processos. A Figura 2.4 mostra a topologia de um AG paralelo do tipo mestre-escravo (Fauzi Mohd Johar et al., 2013).

Figura 2.4: Topologia de um AG paralelo do tipo mestre-escravo.



Fonte: Adaptado de Fauzi Mohd Johar et al. (2013).

Um outro tipo de abordagem paralela em AGs é a de granulado grosso. Esta forma apresenta uma diferença significativa na modelagem de um AG, pois trata-se da paralelização de populações, dividindo em subgrupos que são direcionados a processos diferentes (ilhas). Cada ilha tem seu próprio processo evolutivo, com suas respectivas gerações e operadores de seleção, cruzamento e mutação. Os processos trabalham como um AG sequencial, separadamente, e durante as execuções dos processos, indivíduos de uma ilha migram para outra a fim de gerar uma variabilidade genética maior (Madhuri e Deep, 2009).

Por último, tem-se os AGs paralelos do tipo granulado-fino ou celular, onde indivíduos são distribuídos em processos diferentes e as operações de cruzamento ficam restritas a processos (ou células) vizinhas. AGs paralelos do tipo granulado-fino requerem uma topologia bem definida e geralmente decaem em performance à medida que a população aumenta (Fauzi Mohd Johar et al., 2013). Esta é uma alternativa viável principalmente quando implementada sobre um dispositivo de processamento SIMD (Single Instruction Multiple Data).

Existem ainda AGs paralelos híbridos, que utilizam duas ou mais abordagens descritas anteriormente para criar um novo modelo. Esse modelo é também chamado de modelo hierárquico. Embora AGs paralelos híbridos adicionem um nível de complexidade ao problema, podem ser essenciais em contextos onde as abordagens anteriores não consigam ser utilizadas da melhor maneira.

Como exemplo de utilização de AG paralelo, Kurdi (2016) utiliza uma abordagem baseada no modelo de ilhas (ou granulado grosso) para resolver o problema de escalonamento de tarefas. Este problema, que é NP-difícil, consiste em determinar uma atribuição de tarefas para processadores a fim de otimizar o tempo de execução total das tarefas. No trabalho, cada ilha tem seus próprios operadores genéticos e a cada período de migração, os indivíduos menos aptos migram primeiro para outras ilhas, na tentativa de encontrar um meio em que possam se adaptar melhor.

Cai et al. (2016) traz o modelo de AG paralelo do tipo mestre-escravo no contexto de otimização industrial. Os autores trabalham no problema de encontrar rotas seguras e eficientes para o levantamento e movimentação de cargas pesadas em ambientes industriais, em situações de construção, manutenção ou instalação de equipamentos. O AG paralelo é implementado com Unidades de Processamento Gráficos (GPU, do inglês *Graphics Processing Units*) utilizando programação CUDA para executar as tarefas de avaliação de fitness, seleção, cruzamento e mutação. Já o processo mestre é executado na CPU (*Central Processing Unit*), coordenando toda a execução do AG.

Capítulo 3

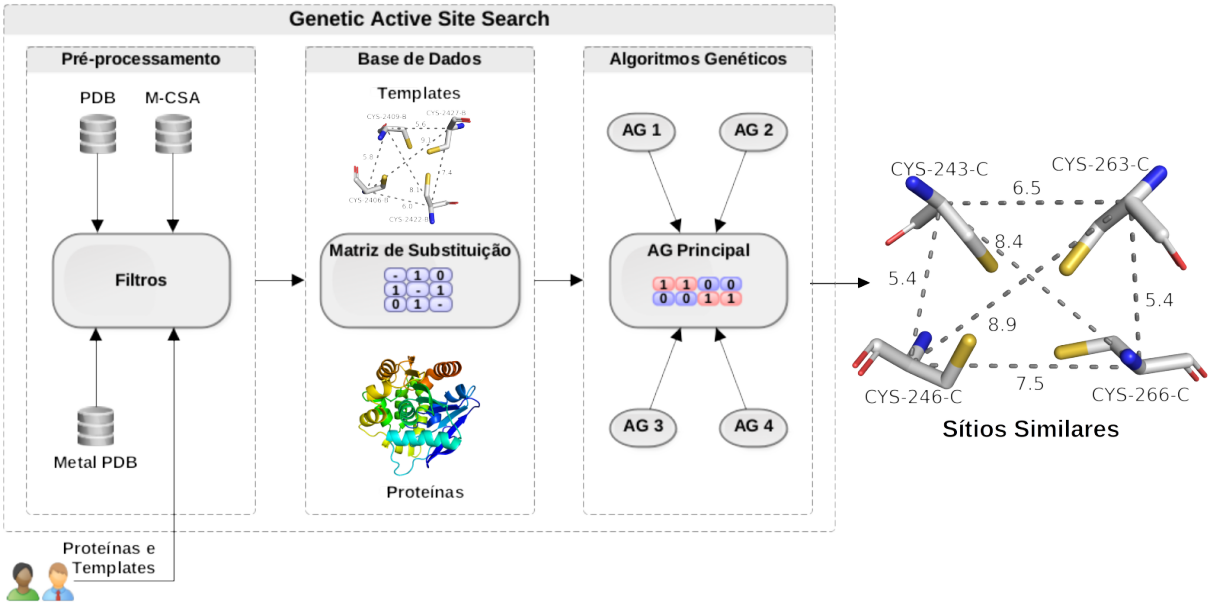
Metodologia

Este capítulo discute a metodologia utilizada para a busca de sítios metálicos similares através de informações estruturais de proteínas. O método de busca é baseado no *Genetic Active Site Search* (GASS) (Izidoro et al., 2014), um método que utiliza um algoritmo genético para a busca de sítios ativos baseados em *templates*.

Izidoro et al. (2014) define o problema de busca baseada em *templates* da seguinte forma: dado um conjunto de N aminoácidos que compõe o sítio ativo A de uma enzima de função conhecida (*template*), e uma proteína hipotética B com M aminoácidos de função desconhecida, o método procura o padrão A em B.

O GASS-Metal utiliza do método GASS em um contexto de sítios metálicos em metaloproteínas. A Figura 3.1 mostra a metodologia proposta para a busca de sítios metálicos baseada em dados estruturais.

Figura 3.1: Metodologia proposta para a busca de sítios metálicos similares.



Na etapa de pré-processamento o usuário pode apresentar uma proteína alvo para efetuar a busca de sítios metálicos similares. Utilizando dados do M-CSA, PDB e MetalPDB,

o GASS-Metal gera uma base de dados contendo um conjunto de *templates* de sítios metálicos, quatro repositórios de resíduos de aminoácidos provenientes da proteína alvo (cada um representando um quadrante no espaço tridimensional da proteína), e uma matriz de substituição de resíduos de aminoácidos para tratar problemas de mutação conservativa.

A proteína alvo é dividida em quadrantes, e cada quadrante corresponde a um AG paralelo que contém sua própria população e operadores genéticos. Os quatro AGs funcionam para produzir uma população inicial uniforme para o AG principal. Ao final da execução do último AG tem-se uma lista de indivíduos candidatos para solução do problema ordenados pelo valor de *fitness* (aptidão). Assim como em Izidoro et al. (2014), o GASS-Metal também pode lidar com sítios interdomínio (resíduos de aminoácidos em várias cadeias) e com mutação conservativa (busca não exata de resíduos).

As seções a seguir tratam com mais detalhes as etapas de pré-processamento e modelagem do GASS-Metal, bem como características de sua implementação utilizando computação paralela.

3.1 Pré-processamento

Uma das etapas do pré-processamento do GASS-Metal consiste na busca e seleção de sítios metálicos que irão compor um repositório de *templates*. Cada *template* contém informações sobre os resíduos (aminoácidos) que compõem um determinado sítio. Essas informações sobre os resíduos são: nome, átomo de referência (neste trabalho foi utilizado o carbono alfa como referência), cadeia em que o resíduo (aminoácido) se encontra, as coordenadas no espaço tridimensional do átomo de referência e a função que o sítio metálico relacionado desempenha na proteína.

Assim como o GASS, o GASS-Metal também trabalha com sítios interdomínios (cadeias diferentes), onde é possível utilizar *templates* como o da proteína 3NOS (*Human Endothelial Nitric Oxide Synthase With Arginine Substrate* - EC: 1.14.13.39), que contém os resíduos do sítio metálico de zinco (4 cisteínas) distribuídos em 2 cadeias diferentes (cadeias A e B).

Outra etapa do pré-processamento está relacionada à geração do repositório de resíduos que será utilizado para a geração da população inicial do AG, e fornecer resíduos para as operações de mutação (mutação normal e mutação conservativa). Uma mutação é classificada como conservativa quando um resíduo é substituído por outro com propriedades físico-químicas semelhantes, não influenciando na estabilidade e função da proteína (Jonson e Petersen, 2001).

As subseções seguintes descrevem em detalhes os procedimentos do pré-processamento.

3.1.1 Geração dos *templates* de sítios metálicos

Para a geração dos *templates* do GASS-Metal, foram utilizados dados do M-CSA (Ribeiro et al., 2017), PDB (Berman et al., 2000) e MetalPDB (Putignano et al., 2017). O M-CSA possui basicamente dois conjuntos de dados: um conjunto de proteínas anotado manualmente e derivado de literatura primária (LIT), e um segundo conjunto formado por proteínas homólogas (encontradas por alinhamento com as proteínas LIT utilizando PSI-BLAST). Para cada proteína LIT foi verificada a existência de sítios metálicos no MetalPDB. Uma vez encontrado um sítio metálico, este foi anotado utilizando mais informações do PDB. Para cada sítio encontrado é anotado o nome da proteína, o tipo de íon metálico, os resíduos e suas posições na sequência e cadeia. Ao mesmo tempo, são gerados os arquivos com as coordenadas espaciais (X, Y, Z) do átomo de carbono alfa que serão utilizados para o cálculo da *fitness*. As listas com todos os *templates* utilizados pelo GASS-Metal estão no GitHub[1].

3.1.2 Geração da matriz de substituição de resíduos

Assim como no GASS, o GASS-Metal também utiliza uma matriz de substituição para lidar com a mutação conservativa. O objetivo é permitir ao AG gerar indivíduos com resíduos diferentes aos do template. A matriz de substituição de resíduos foi baseada em Nilmeier et al. (2013), e sua diferença com relação a outras matrizes de substituição existentes, como a Blosum62 (Henikoff e Henikoff, 1992) ou a MIQS (Yamada e Tomii, 2013), é que ela é específica para cada *template*.

Para criar a matriz de substituição, as proteínas LIT do M-CSA (que são a base dos *templates*) são comparadas com as respectivas proteínas homólogas, seguindo os mesmos dois critérios encontrados em Nilmeier et al. (2013):

1. O número de resíduos na proteína homóloga deve ser o mesmo na proteína LIT;
2. O número do *Enzyme Commission* (EC) deve ser igual entre as duas proteínas.

O EC é um sistema numérico e hierárquico de classificação de enzimas. Existem 7 principais categorias: (1) Oxidoredutases, (2) Transferases, (3) Hidrolases, (4) Liases, (5) Isomerases, (6) Ligases e (7) Translocase. Formado por 4 números (#.#.#.#), o número EC classifica as enzimas, fornecendo detalhes sobre suas reações enzimáticas (enz, 1999).

As substituições observadas envolvendo as proteínas LIT e suas homólogas são anotadas para cada entrada LIT, gerando uma lista de possíveis substituições. Por exemplo, o *template* de zinco da proteína LIT 2BMI (*Metallo-Beta-Lactamase* - EC: 3.5.2.6) possui uma substituição CYS $\leftrightarrow$ HIS, indicando que o resíduo CYS pode ser substituído pelo resíduo HIS ou vice-versa.

[1] https://github.com/sandroizidoro/GASS-METAL

Ao todo, foram observadas 184 substituições possíveis entre os *templates* (entradas LIT). Assim, cada *template* terá sua matriz de substituição. A lista de substituição contendo todas substituições para todos os *templates* está junto com as listas de *templates* no GitHub.

3.1.3 Geração do repositório de resíduos

Para efetuar a busca de sítios metálicos similares na proteína alvo, é necessário selecionar seus resíduos para formar um repositório de resíduos. Este repositório será utilizado tanto para gerar a população inicial do AG quanto fornecer resíduos para a operação de mutação. Ele é formado com base nos resíduos do mesmo tipo do *template* e dos resíduos definidos na matriz de substituição.

Considere o *template* de magnésio (Mg) ASP, 444, A; ASN, 471, A; GLY, 473, A (resíduo, posição na sequência, cadeia) obtido da proteína LIT 1PVD (*Crystal Structure of the Thiamin Diphosphate Dependent Enzyme Pyruvate Decarboxylase From The Yeast Saccharomyces Cerevisiae* - EC: 4.1.1.1), e a substituição de resíduos prevista para esse *template* (GLY $\leftrightarrow$ ILE). O repositório será formado por todos os resíduos ASP, ASN, GLY e ILE da proteína alvo. Desta forma, para cada *template* diferente ou para cada proteína alvo diferente, se faz necessário redefinir o repositório.

A definição e utilização de um único repositório para todos os resíduos da proteína alvo envolvidos na busca pode não garantir uma boa variabilidade genética para a formação da população inicial do AG. A geração da população inicial é feita de forma aleatória, podendo assim não conter vários resíduos de partes da proteína. Para resolver esse problema, o GASS-Metal trabalha com um AG paralelo (Seção 3.2.5) e quatro repositórios com resíduos distintos.

Inicialmente é calculado o centróide $(X, Y, Z)_C$ da proteína que é obtido por meio do ponto médio das distâncias entre os dois pontos mais distantes em cada uma das coordenadas X, Y e Z (Equação 3.1).

$$(X, Y, Z)_C = \frac{maior(X, Y, Z) - menor(X, Y, Z)}{2} \tag{3.1}$$

Com base no centróide, a proteína alvo é dividida em quadrantes. Um resíduo é pertencente a um quadrante específico quando suas coordenadas X e Y se encaixam seguindo as seguintes regras:

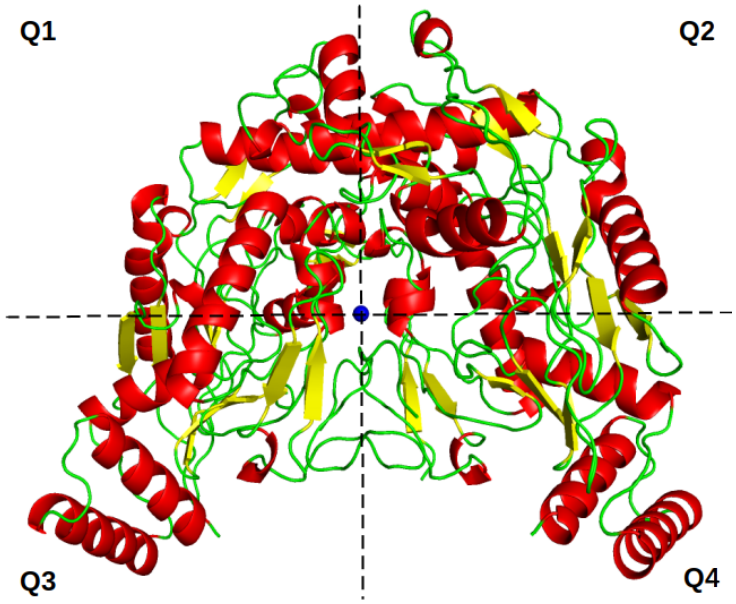
- Quadrante 1: se a coordenada X do resíduo $>$ a coordenada X do centróide e a coordenada Y do resíduo $>$ a coordenada Y do centróide;
- Quadrante 2: se a coordenada X do resíduo $>$ a coordenada X do centróide e a coordenada Y do resíduo $<$ a coordenada Y do centróide;
- Quadrante 3: se a coordenada X do resíduo $<$ a coordenada X do centróide e a coordenada Y do resíduo $<$ a coordenada Y do centróide;

- Quadrante 4: se a coordenada X do resíduo $<$ a coordenada X do centróide e a coordenada Y do resíduo $>$ a coordenada Y do centróide.

Por se tratar de quadrantes em um espaço tridimensional, uma das três coordenadas (X, Y e Z) pode ser descartada. Neste caso a coordenada Z não foi utilizada.

Assim, ao final desta definição, todos os resíduos da proteína estão classificados em seus respectivos quadrantes e prontos para serem utilizados na operação de mutação (mais detalhes na Seção 3.2.3) e na definição da população inicial (mais detalhes na Seção 3.2.1). A Figura 3.2 mostra um exemplo da divisão em quadrantes utilizando a proteína 3NOS.

Figura 3.2: Proteína 3NOS com o centróide (esfera azul no centro) e seus quadrantes.



3.2 Modelagem do algoritmo genético

A Seção 2.4 trouxe diversas definições e características de AGs, bem como contextos em que são utilizados. Esta seção descreve o AG utilizado pelo GASS-Metal, apresentando desde a definição do indivíduo, passando pela função *fitness*, operadores genéticos, parâmetros e sua implementação paralela.

O GASS-Metal utiliza o método GASS (*Genetic Active Site Search*), que foi proposto inicialmente para busca de sítios ativos (catalíticos e de ligação) em proteínas, em um contexto de sítios metálicos em metaloproteínas. O GASS-Metal busca por resíduos em uma proteína baseando-se em seus *templates* de sítios metálicos, retornando indivíduos que sejam próximos em distância em relação aos seus *templates*.

Resumidamente, o GASS-Metal utiliza *templates* de sítios metálicos para encontrar resíduos que possam representar um sítio metálico em uma proteína alvo. Essa proteína alvo é dividida em quadrantes, e cada quadrante corresponde a um AG paralelo que contém sua própria população e operadores genéticos. Durante a execução dos AGs

nos quadrantes, uma outra população é criada, que será usada por um último AG geral (principal). Ao final da execução do último AG tem-se uma lista de indivíduos candidatos para solução do problema de busca de sítios metálicos em proteínas.

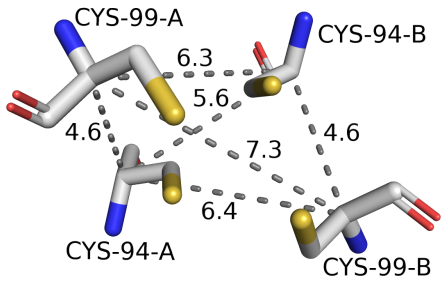
As subseções seguintes descrevem os indivíduos e população do GASS-Metal, bem como a função *fitness* utilizada, os métodos de seleção, mutação e cruzamento e também a descrição da implementação de AG paralelo utilizada.

3.2.1 Representação do indivíduo e população

O primeiro passo para a modelagem de um AG é a definição dos indivíduos, que são possíveis soluções para o problema. É uma etapa crucial pois um indivíduo de um AG deve ser uma abstração do indivíduo do mundo real, e dessa forma representações ruins podem influenciar no resultado final. Para o problema de busca de sítios metálicos, um indivíduo é um vetor que contém características do sítio. Essas características compreendem de informações sobre os resíduos (aminoácidos) que compõem o sítio metálico.

Para cada resíduo pertencente a um sítio metálico são armazenadas as seguintes informações: o nome do resíduo, a posição do carbono α no espaço tridimensional (coordenadas X, Y e Z), a posição do aminoácido na sequência e a cadeia em que se encontra. A Figura 3.3 mostra a representação de um indivíduo utilizado pelo GASS-Metal.

Figura 3.3: Representação de um candidato a sítio metálico - (a) Sítio de zinco da enzima 3NOS com as distâncias (em Angstroms) entre os átomos de carbono α de cada resíduo - (b) Indivíduo do GASS-Metal.



(a)

CYS CA A 94	CYS CA A 99	CYS CA B 94	CYS CA B 99
22.176 -4.032 42.155	21.369 -0.507 45.039	16.714 -4.431 43.236	16.810 -1.800 39.465

(b)

Como visto anteriormente, na etapa de pré-processamento os resíduos são separados em repositórios diferentes, cada um correspondente a um quadrante no espaço tridimensional da proteína. Cada um dos quatro AG paralelos trabalha com um quadrante diferente, e a população de cada AG é constituída somente por indivíduos que estejam naquele quadrante, selecionados aleatoriamente do repositório correspondente. Ou seja, dado o

indivíduo Ai (com i variando de 1 a 400), com N resíduos, da população Pj (com j variando de 1 a 4), todos os N resíduos do indivíduo Ai pertencem ao quadrante j. Esta regra de manter apenas indivíduos referentes a um quadrante tem apenas uma exceção: quando não há um determinado resíduo em um dos quadrantes. Neste caso, um resíduo é selecionado aleatoriamente de outro quadrante para compor o indivíduo. Os demais resíduos do indivíduo permanecem atrelados ao quadrante atual.

Um exemplo onde a regra *indivíduo-quadrante* não ocorre em todos os casos é na proteína 1EUU (*Sialidase or Neuraminidase, Large 68Kd Form* - EC: 3.2.1.18). Ao buscar o sítio metálico de sódio (Na) composto por 6 resíduos (2 ASN, 1 ASP, 1 ALA, 1 GLU e 1 THR) na proteína 1EUU, observa-se que não existe nenhum GLU no quadrante 2, impossibilitando assim a geração de uma população exclusivamente com indivíduos desse quadrante. Quando esse tipo de situação ocorre, um outro resíduo GLU é selecionado aleatoriamente de qualquer outro quadrante para compor os indivíduos do quadrante 2 da 1EUU. Assim, a população inicial do AG referente ao quadrante 2 é formada por indivíduos com os resíduos ASN, ASP, ALA e THR sempre do quadrante 2, porém os resíduos de GLU são dos quadrantes 1, 3 e 4.

A população inicial do último AG é composta por indivíduos provenientes das execuções dos AGs paralelos anteriores. Em cada um dos quadrantes, e a cada 20 gerações, os 10 indivíduos mais bem ranqueados são inseridos na população inicial do último AG. Dessa forma, ao final das 200 gerações de cada AG por quadrante, 100 indivíduos são inseridos no AG final, totalizando 400 indivíduos de sua população.

3.2.2 Função de avaliação (*fitness*)

Com a população inicial gerada, tem início a etapa de avaliação dos indivíduos. A função *fitness* utilizada pelo GASS-Metal para essa avaliação é a mesma do GASS, onde se trabalha com as distâncias entre as coordenadas dos átomos de carbono alfa de cada resíduo do *template* e as coordenadas de cada átomo de carbono alfa dos resíduos dos candidatos a sítios metálicos encontrados.

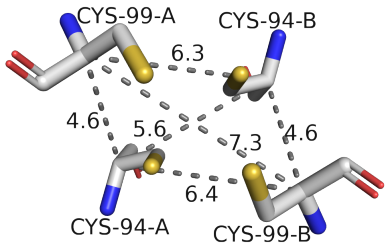
Como apresentado por Izidoro et al. (2014), a Equação 3.2 mostra o valor n sendo o número de resíduos no *template* e no indivíduo, v sendo as coordenadas 3D para cada par de resíduos no *template* e o valor w as coordenadas de cada par de resíduos do sítio metálico candidato. Baseada na conhecida função *Root Mean Square Deviation* (RMSD) (Maiorov e Crippen, 1994; Laskowski et al., 2005), a função de avaliação do GASS-Metal se diferencia por não calcular o quadrado da média das distâncias dos resultados.

$$Fit(\mathbf{v}, \mathbf{w}) = \sqrt{\sum_{i=1}^{(n^2-n)/2} \|v_i - w_i\|^2} \tag{3.2}$$

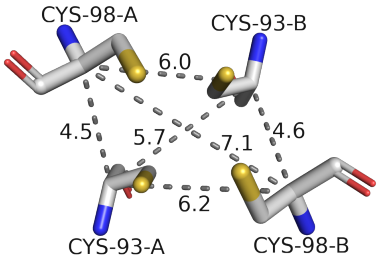
A Figura 3.4 mostra os sítios metálicos das proteínas 3NOS, 3NLX, 3E7S e 5NSE, e as distâncias em Angstroms medidas a partir da posição do carbono α de cada resíduo.

Abaixo de cada sítio metálico existe um valor *fitness* que foi calculado utilizando o sítio metálico da proteína 3NOS como template. Ao calcular o *fitness* do *template* 3NOS na própria proteína, seu valor aparece como zero pois são exatamente os mesmos resíduos, não existe nenhuma diferença entre as suas distâncias. Já nas outras proteínas, é possível ver um valor diferente de zero. Quanto mais próximo de zero o valor do *fitness*, maior a similaridade, em termos de distância, entre o *template* e o sítio candidato.

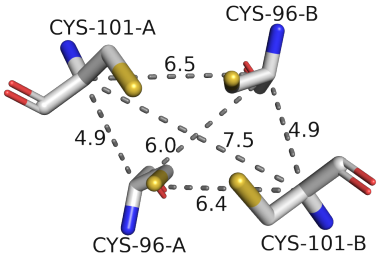
Figura 3.4: Sítios metálicos de zinco das enzimas 3NOS, 3E7S, 5NSE e 3NLX, e seus respectivos valores de *fitness* utilizando a 3NOS como template.



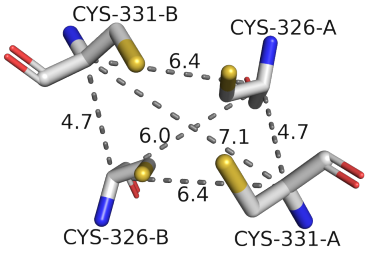
(a) 3NOS - Fitness 0



(b) 3E7S - Fitness 0,628



(c) 5NSE - Fitness 0,556



(d) 3NLX - Fitness 0,714

3.2.3 Seleção e operadores genéticos

Depois que os indivíduos são avaliados pela função *fitness*, vem a fase de seleção. Essa etapa é importante para a evolução da população do AG, pois dá uma chance maior de sobrevivência aos melhores indivíduos. Os métodos de seleção e operadores genéticos utilizados pelo GASS-Metal são os mesmos utilizados pelo GASS.

Na literatura existem diversos métodos de seleção (Eiben e Smith, 2007). Neste trabalho, foi utilizado o método de seleção por torneio, onde um subconjunto K de indivíduos é sorteado aleatoriamente da população, e o melhor indivíduo (de acordo com seu valor obtido pela função fitness) desse subconjunto é selecionado. A seleção por torneio tem a característica de não requerer conhecimento sobre a população, utilizando apenas uma relação de ordem para classificar K indivíduos.

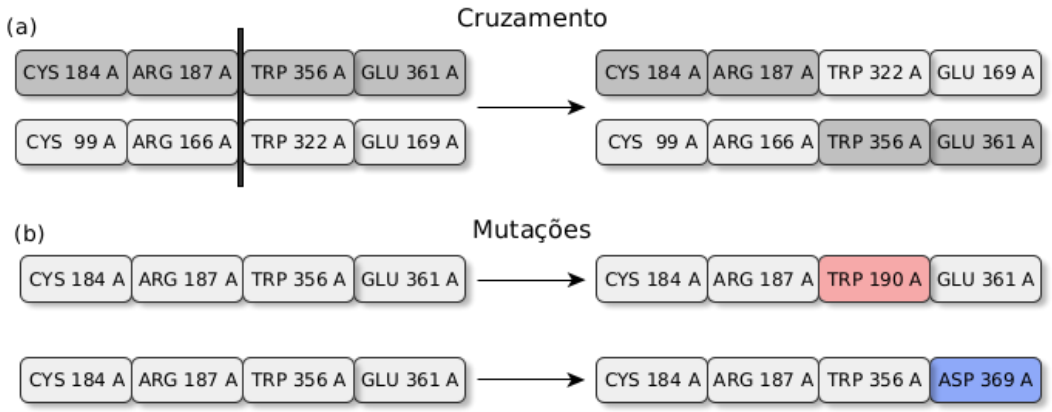
Outro ponto positivo neste método de seleção é a capacidade de controlar a pressão seletiva (convergência prematura) por meio do parâmetro K. A convergência prematura ocorre quando, em poucas gerações, o AG estabiliza, ficando limitado a explorar apenas uma parte do espaço de busca. Isso faz com que o AG encontre apenas soluções ótimas

locais, e não globais. Desta forma, quanto maior o valor de K, maior será a pressão seletiva.

Tendo sido feita a seleção, dois operadores genéticos são utilizados para gerar uma nova população: cruzamento e mutação. O GASS-Metal utiliza o cruzamento do tipo um ponto (ou simples) e mutação do tipo uniforme. Na operação de cruzamento, são necessários dois indivíduos da população. Uma posição aleatória dos indivíduos é selecionada e os resíduos antes desse ponto no primeiro pai são concatenados com os resíduos depois desse ponto no segundo pai, gerando assim um novo indivíduo (Figura 3.5 (a)).

Na mutação, um ponto é escolhido e substituído por um resíduo aleatório, que pode ser do mesmo tipo ou por um tipo diferente de resíduo, que é o caso da mutação conservativa. A mutação conservativa é indicada pela matriz de substituição de resíduos da mesma proteína (Figura 3.5 (b)).

Figura 3.5: Representação dos operadores de cruzamento e mutação. Na representação resumida do indivíduo tem-se o nome do resíduo, a posição na sequência e a cadeia. (a) Cruzamento do tipo um ponto. (b) Mutação de um Triptofano por outro Triptofano (TRP 365 por TRP 190), e mutação de um Glutamato por um Aspartato (GLU 361 por ASP 369 - mutação conservativa).



Fonte: Izidoro et al. (2014)

3.2.4 Parâmetros

Para chegar aos valores finais dos parâmetros utilizados pelo GASS-Metal, uma série de *testes de sanidade* foram realizados. Em um *teste de sanidade*, um respectivo *template* é usado como base para ser buscado na mesma proteína em que se encontra. Para exemplificar, a proteína 1EUU contém um sítio metálico de sódio (Na) com seis resíduos na cadeia A e o *template* que corresponde a esse sítio foi usado como base para a busca na própria proteína 1EUU.

No cenário ideal, o *teste de sanidade* deveria retornar sempre o próprio *template* base como um dos indivíduos da população final e o resultado da função *fitness* para esse indivíduo deveria ser 0 Å. Porém, pela própria característica não-determinística de um AG, nem sempre isso acontece, e os valores dos parâmetros do AG influenciam diretamente nos resultados.

Os testes compreenderam 30 execuções do GASS-Metal para cada um dos 928 *templates* de sítios metálicos, e o menor valor de *fitness* dentre as 30 execuções foi salvo. Dessa forma, os 928 *templates* foram classificados em 3 grupos: o grupo 1 corresponde aos *templates* onde o menor valor de *fitness* encontrado foi igual a 0 Å, o grupo 2 aos *templates* que o valor de *fitness* era maior que 0 Åe menor que 3 Åe o grupo 3 aos *templates* em que o menor valor de *fitness* encontrado era igual ou maior a 3 Å. Assim, dado dois conjuntos de parâmetros (taxas de cruzamento, mutação e torneio) de AG diferentes A e B, o conjunto de parâmetros A é melhor que B caso o número de *templates* do grupo 1 de A for maior que o grupo 1 de B, se o grupo 2 de A for maior que o grupo 2 de B e o mesmo para o grupo 3 de ambos os parâmetros.

O número de gerações e o tamanho da população foram inicialmente configurados com os mesmos valores utilizados na versão web do GASS (GASS-WEB (Moraes et al., 2017)). Testes variando esses valores também foram realizados visando melhorar os resultados sem impactar no tempo de processamento.

Após a execução de 35 testes diferentes (cada um com 30 execuções de todos os 928 templates) com 20 parâmetros diferentes, chegou-se aos valores finais de cada um dos parâmetros do AG utilizado pelo GASS-Metal. Esses valores se mantiveram constantes, independente do template, proteína ou quadrante em que o AG estava sendo executado. A seguir são listados cada um dos parâmetros e seus respectivos valores:

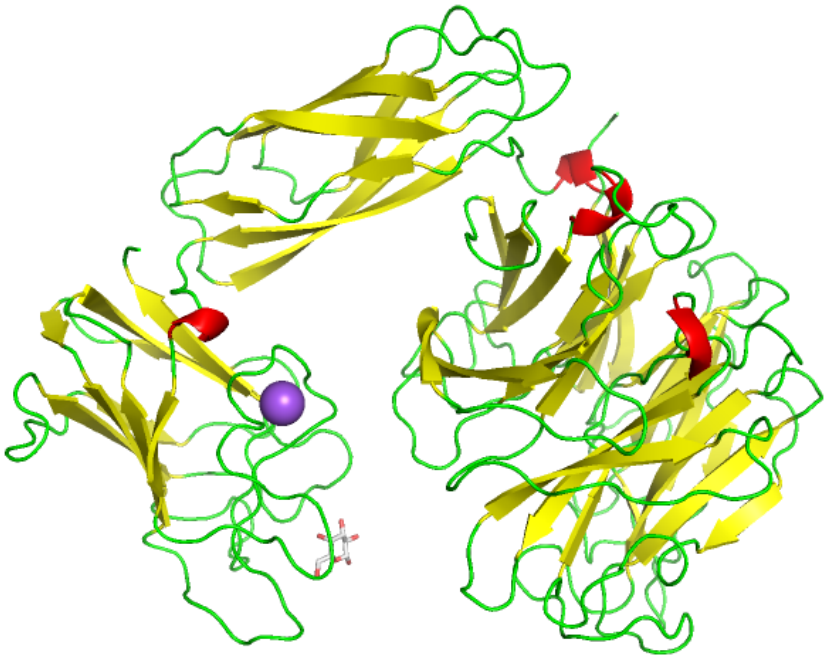
- Número de gerações: 200;
- Tamanho da população: 400;
- Seleção por torneio: 20 indivíduos (5% da população);
- Taxa de mutação: 90%;
- Taxa de cruzamento: 90%;
- Tamanho do ranking: 100.

Como mencionado na Seção 2.4.4, a probabilidade de mutação geralmente possui taxas baixas, mas é dependente do contexto da aplicação do AG. Apesar do valor da taxa de mutação utilizada pelo AG do GASS-Metal se apresentar muito acima dos valores reportados na literatura, ele está em conformidade com os resultados dos experimentos e com outras fontes na literatura, tais como Cetin e Gundogmus (2019); Doerr et al. (2015); Cervantes e Stephens (2006).

3.2.5 AG paralelo

Ao utilizar o método GASS no contexto de sítios metálicos, foram observados casos onde a densidade dos resíduos das proteínas no espaço tridimensional era irregular. É o caso da proteína 1EUU, que como pode ser observada na Figura 3.6, contém três principais regiões com densidades diferentes de resíduos.

Figura 3.6: Estrutura da proteína 1EUU.



Esse tipo de padrão observado em algumas proteínas faz com que a seleção de resíduos para compor os indivíduos da população inicial do AG tenha uma probabilidade maior de escolher resíduos de uma região mais densa da proteína. Os poucos representantes das regiões menos densas geralmente formam indivíduos com resíduos de outras regiões muito distantes, fazendo com que esses indivíduos não sobrevivam aos processos de seleção do AG e se "percam" no meio das gerações. O problema maior é quando os resíduos que compõem as melhores solução do problema estão em uma região não muito densa.

Por isso, para o GASS-Metal, viu-se a necessidade de abordar o problema de inicialização da população do AG, bem como o desenvolvimento das gerações e operadores de seleção, cruzamento e mutação, em partes (de maneira paralela). Cada AG trabalha apenas com indivíduos de uma região da proteína, ou quadrantes. Dessa forma, é possível garantir que mesmo regiões não muito densas nas proteínas possam ter indivíduos presentes apenas nessas regiões, fazendo com que os mesmos possam se adaptar de acordo com a direção da evolução de determinadas características.

A modelagem paralela apresentada neste trabalho trata-se de uma versão não padrão de uma granulação grossa. Estratégias não padrão são utilizadas quando certos benefícios são buscados e abordagens clássicas não são capazes de fornecê-los. No caso do GASS-

Metal, a paralelização em ilhas é utilizada de forma que a população de cada ilha seja gerada e mantida apenas em um quadrante. Assim, quatro ilhas são executadas simultaneamente, sem migração entre essas, porém seus melhores indivíduos são enviados a uma população final, onde os operadores do AG serão aplicados de forma global, para que os melhores resultados possíveis sejam alcançados.

Existem diversas ferramentas para de fato implementar concorrência em um programa, cada uma destas com características e propostas únicas. Em sistemas distribuídos de maior porte, que geralmente contam com memória primária distribuída, se torna mais comum o uso de uma biblioteca de MPI (*Message Passing Interface*), já em sistemas modelados com memória primária única, a ferramenta mais básica de paralelização se tornam as *threads*.

A paralelização de um processo por meio de *threads* é um modelo possível devido a forma como a maioria dos sistemas operacionais modernos administra e executa suas tarefas. Isso somado ao fato de que existe um direcionamento do desenvolvimento de unidades de processamento, fazendo com que cada processador tenha mais de um núcleo. Desta forma, um processo é capaz de lançar outros processos que executam de forma assíncrona (Butenhof, 1997), cada um sendo imposto ao escalonamento de processos do sistema operacional e recebendo uma fatia de tempo de execução. Este processo se repete em todos os núcleos, balanceando assim a carga computacional ao executar de forma concorrente.

Ao buscar a execução de um processo em um sistema totalmente distribuído um obstáculo é encontrado. Cada nó de processamento possui sua própria memória principal, onde são alocados os recursos necessários para sua execução, e possivelmente seus resultados. A comunicação entre instâncias de um mesmo processo então fica dificultada, por ter dados alocados em outra máquina com endereçamento inacessível. Neste caso faz-se necessário um protocolo de comunicação entre esses nós, e esta é a proposta das bibliotecas MPI (Gabriel et al., 2004), transmitir dados e informações entre máquinas durante a execução de um processo em comum.

No caso da implementação realizada neste trabalho, a ferramenta escolhida foi a paralelização por *threads*. Esta escolha se deu pelos critérios de desempenho, facilidade de implementação e pela própria modelagem proposta para estudo. Embora sistemas distribuídos tenham maior escalabilidade, essa grande capacidade de processamento não é necessária para a implementação de um servidor web para este contexto. Nesta situação, as vantagens do uso de MPI se tornam pequenas, já que há apenas uma memória principal, não havendo necessidade de troca de mensagens entre as instâncias deste processo. As vantagens de uma implementação por *threads* se tornam mais evidentes, já que pode-se escolher exatamente o ponto a ser executado em paralelo, evitando repetir processos de leitura de arquivos e pré-processamentos necessários para o AG.

Após gerar o código propriamente modelado para o uso por *threads*, o uso deste modelo pode ser aplicado à implementação através de poucas chamadas de funções de bibliotecas.

Para manter a implementação mais prática e replicável foi escolhida a biblioteca padrão de *threads* da linguagem C++, presente desde a versão C++11 (ISO, 2012). Esta é fortemente baseada no sistema de *threads* do sistema operacional Unix, sistema chamado de POSIX Threads. A versão padrão oferece diversas vantagens, como o encapsulamento total nos padrões da linguagem e suporte multiplataformas, não se restringindo apenas aos sistemas Unix.

As *threads* padrão também são implementadas no paradigma de programação orientado a objetos, desta forma o trecho de código a ser executado pela *thread* é informado no seu construtor e o processo é automaticamente iniciado. Assim como o término da execução de uma *thread* pode ser assegurado pela *thread* lançadora utilizando o método *join()*.

Como em qualquer implementação concorrente em memória única, deve-se proteger os acessos e escritas a recursos que são compartilhados entre as instâncias em execução. Neste caso, existe a situação que o AG de cada quadrante insere indivíduos na população inicial do último AG. Logo, este trecho de código é protegido por um sistema de exclusão mútua (mutex), garantindo que apenas uma thread possa acessar esta população do último AG de cada vez. Este efeito é obtido utilizando os métodos *lock()* e *unlock()* da classe mutex padrão da linguagem.

Os benefícios de performance do método de paralelização escolhido podem ser observados primeiramente pela redução de 33,98% no tempo de execução médio obtido. O valor base para o cálculo deste percentual foi o tempo de execução do AG para 20 proteínas homólogas da proteína 3NOS, utilizando seu sítio metálico (Zn) como *template*. A execução deste teste pode ser visto para a versão sequencial na Figura 3.7 e para a versão paralela em *threads* na Figura 3.8. Os valores de *fitness* obtidos indicam bom funcionamento do método em ambas as versões testadas e validam o uso deste teste.

Na região direita das Figuras 3.7 e 3.8 também pode ser observado a execução do comando *htop*, que indica de maneira gráfica o uso dos recursos do sistema por cada processo. Na Figura 3.7 pode-se perceber o uso intenso de um núcleo do sistema, enquanto outros permanecem inativos, ou com carga de execução baixa. Esse tipo de uso é característica de execuções sequenciais, totalmente em uma única thread. Desta forma, enquanto o uso de um núcleo chega a 79,3% no momento da captura de tela, o uso de recursos de computação total do sistema é de apenas 12,6%.

Já para a versão em múltiplas threads (Figura 3.8), é possível observar o uso de todos os núcleos de processamento do sistema de forma paralela, característica de execuções concorrentes de várias threads. Desta forma, mesmo cada núcleo tendo uma porcentagem de uso mais baixa, essa execução atinge o uso total de computação de 21,7%.

Esta diferença de eficiência no uso de recursos computacionais pode ser observada na Figura 3.9, com estatísticas de uso capturadas durante a execução das 20 proteínas selecionadas. Percebe-se que na execução sequencial, para cada dado tempo, apenas um núcleo apresenta uso significativo (este apenas troca de núcleo devido ao escalonamento

Figura 3.7: Execução do AG com 20 proteínas - Versão Sequencial.

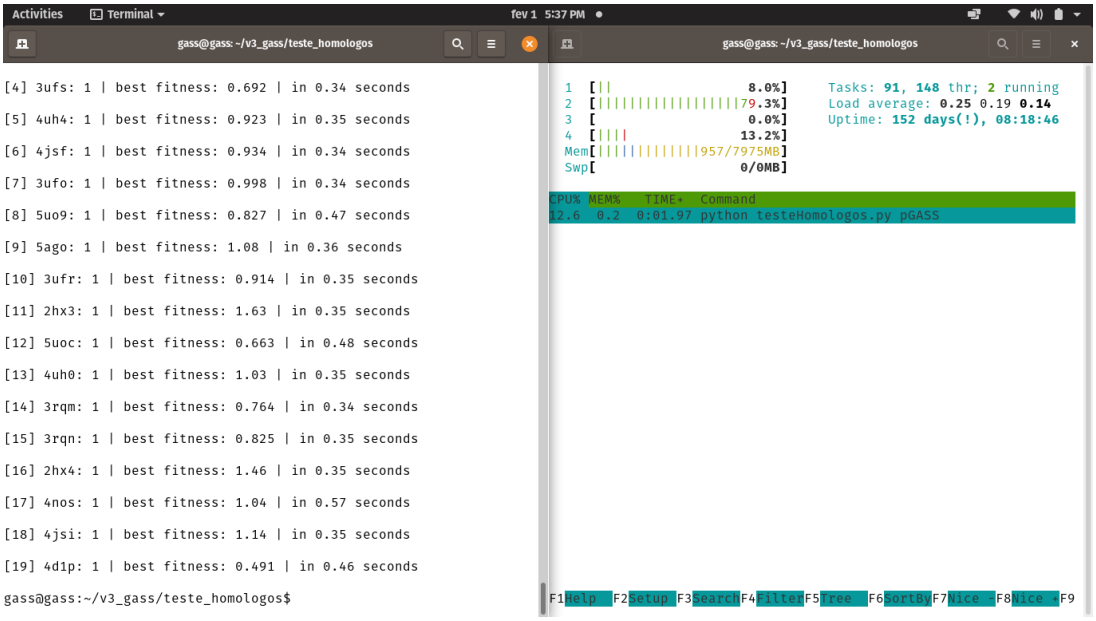
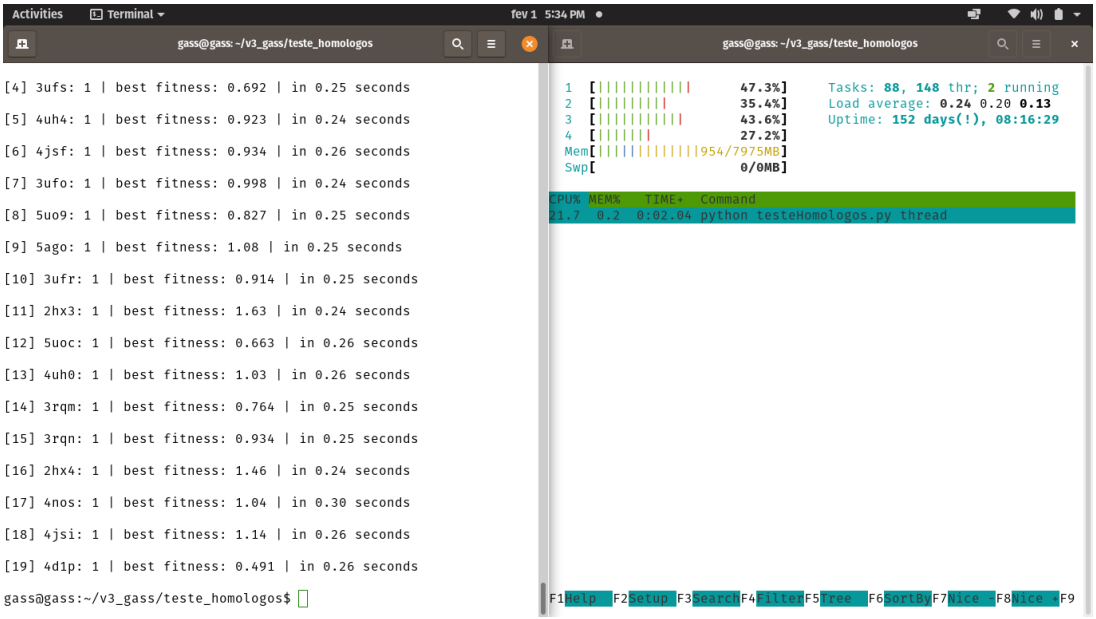
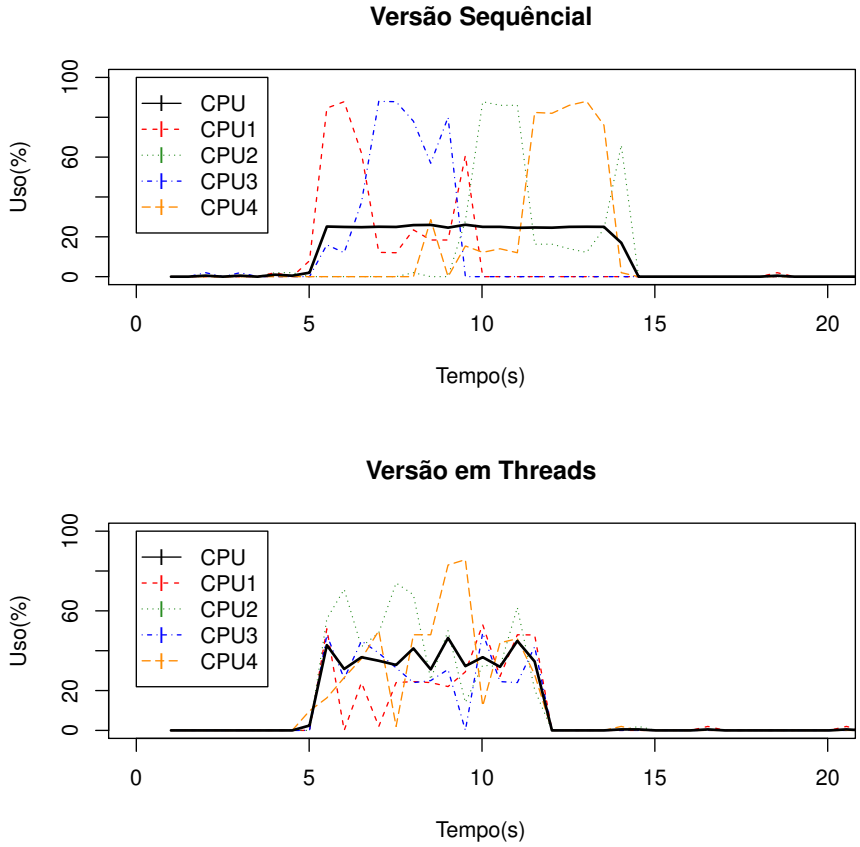


Figura 3.8: Execução do AG com 20 proteínas - Versão *thread*



de processos do sistema operacional). Já na execução por *threads*, ocorrem alta em todos os núcleos simultaneamente, levando assim a um menor tempo de execução total.

Figura 3.9: Execução do AG - Sequencial x Paralelo



3.3 Conjuntos de dados

Com o objetivo de testar e validar a metodologia implementada para a identificação de sítios metálicos similares, três conjuntos de dados foram utilizados nos experimentos. Todos os dados foram extraídos do PDB, M-CSA e MetalPDB, e pré-processados (Seção 3.1). Os conjuntos de dados (CD) foram utilizados para os seguintes experimentos:

- **Teste de sanidade (CD1):** 928 *templates* provenientes de 320 proteínas LIT do M-CSA. Cada *template* foi utilizado para buscar a si próprio na proteína de origem.
- **Teste com homólogos (CD2):** 8 *templates* de diferentes íons metálicos escolhidos aleatóriamente e seus respectivos homólogos. No total, o conjunto de homólogos possui 649 proteínas. A Tabela 3.1 apresenta em detalhes os dados do CD2.
- **Testes com outros métodos (CD3):** 274 proteínas com sítios metálicos de Zn, Ca, Mg, Mn, Fe, Cu, Co, Na, K, Cd e Ni, utilizado por Qiao e Xie (2019) em seu *Independent validation dataset*, ou base de dados de validação. Apesar de trazer o nome das proteínas e a quantidade de resíduos que compõem os sítios metálicos, os autores não informam exatamente quais são os resíduos, ou *targets*, de cada sítio metálico. Com os dados utilizados para testes do MIonSite provenientes da base

de dados BioLip (Yang et al., 2012), seria possível obter os resíduos, porém, até a realização dos testes, a base de dados estava fora do ar devido a problemas técnicos, o que também impossibilitou a obtenção dos resíduos de cada sítio.

Ainda sim é possível deduzir, com alguma semelhança, quais são os resíduos que compõem cada sítio. Para tal, foi utilizada a mesma métrica de definição de sítios metálicos utilizado pelo BioLip, que, como visto na Seção 2.3.4, define que um resíduo pertence a um sítio metálico caso o mesmo esteja a uma distância da soma dos raios de Van der Waals (Haynes, 2014) dos átomos envolvidos somado a uma tolerância de 0.5 Å. Desta forma, dadas as proteínas que compõem a base de dados de validação, calculou-se a posição de cada íon metálico e as distâncias de cada resíduo em relação ao íon. Todos os resíduos que tinham a distância igual ou menor do valor descrito anteriormente foram definidos como pertencentes ao sítio.

A Tabela 3.2 mostra o número de resíduos gerados a partir desse procedimento e compara com o os valores originais utilizado pelo MIonSite. O número de resíduos difere em alguns casos pelo fato do BioLiP não utilizar apenas distâncias para definir os resíduos de um sítio metálico. Existe também um processo manual para avaliar os candidatos. Apesar disso, na grande maioria dos casos a diferença é pequena ou nenhuma, o que indica que os resíduos aqui obtidos são bem próximos aos definidos pelo autor desta base de dados.

Tabela 3.1: Templates e homólogos do CD2.

Íon metálico	Template	Numero de proteínas homólogas
Ca	1MHL	72
Cu	2JCW	43
Fe	3PCA	27
K	1YRC	55
Mg	1VID	41
Mn	1NHX	22
Na	1A50	18
Zn	3NOS	371
Total de proteínas		649

Tabela 3.2: Quantidade de proteínas e resíduos pertencentes a sítios metálicos do CD3 gerados neste trabalho em comparação aos valores originais.

Íon Metálico	Nº de resíduos gerados	Nº de resíduos originais	Nº de proteínas
Zn^{2+}	475	494	69
Ca^{2+}	490	482	64
Mg^{2+}	471	574	93
Mn^{2+}	87	95	23
Fe^{3+}	21	21	7
Cu^{2+}	12	12	3
Fe^{2+}	9	9	3
Co^{2+}	18	18	5
Na^{+}	5	7	1
K^{+}	32	37	3
Cd^{2+}	14	14	2
Ni^{2+}	4	4	1

3.4 Métricas de avaliação

A seguir são apresentados os métodos que foram aplicados nos experimentos.

- *Receiver Operating Characteristic* (ROC): é uma ferramenta para avaliação de algoritmos de aprendizado e predição. A criação deste gráfico é baseada nos valores da matriz de confusão, relacionados como verdadeiro positivo (TP), verdadeiro negativo (TN), falso positivo (FP) e falso negativo (FN). O gráfico ROC é um gráfico bidimensional, onde a taxa de TP (Equação 3.3) é representada no eixo de Y e a taxa de FP (Equação 3.4) é representada no eixo de X, mas que também pode ser representado pelo valor da área abaixo da curva ROC (*Area Under the ROC Curve* - AUC) (Hand, 2009; Fawcett, 2006).

$$Taxa\ \ TP = \frac{Positivos\ \ classificados\ \ corretamente}{Total\ \ de\ \ positivos} \tag{3.3}$$

$$Taxa\ \ FP = \frac{Negativos\ \ classificados\ \ incorretamente}{Total\ \ de\ \ negativos} \tag{3.4}$$

- Sensibilidade (Sen): também chamada de revocação ou *recall*, reflete o quanto um método é eficaz em identificar corretamente, dentre todos os indivíduos avaliados, aqueles que realmente apresentam a característica de interesse. A sensibilidade é calculada pela Equação 3.5.

$$Sen = \frac{TP}{TP + FN} \tag{3.5}$$

- Especificidade (Spe): reflete o quanto um método é eficaz em identificar corretamente os indivíduos que não apresentam a condição de interesse. Esta métrica é também chamada de taxa de verdadeiros negativos. Seu valor é descrito pela Equação 3.6.

$$Spe = \frac{TN}{TN + FP} \tag{3.6}$$

- Acurácia (Acc): indica o grau de concordância que há entre o resultado da medição e o valor verdadeiro (aquele que é aceito, desde que estabelecido por uma definição ou consenso) da grandeza. A Equação 3.7 traz o cálculo da acurácia.

$$Acc = \frac{TP + TN}{TP + FN + TN + FP} \tag{3.7}$$

- *Matthew Correlation Coeficient* (MCC): métrica que avalia a qualidade das predições. Os resíduos do sítio metálico são classificados como verdadeiros positivos (TP: resíduos do sítio metálico preditos corretamente), verdadeiros negativos (TN: predição correta para resíduos não vinculados ao sítio), falsos negativos (FN: resíduos do sítio metálico incorretamente não preditos), falsos positivos (FP: resíduos não vinculados ao sítio metálico incorretamente predito). O valor de MCC varia de +1 (predição perfeita) a -1 (predição inversa), onde um MCC de valor 0 corresponde a uma predição aleatória (Equação 3.8).

$$MCC = \frac{(TP.TN - FP.FN)}{\sqrt{(TP + FP).(TP + FN).(TN + FP).(TN + FN)}} \tag{3.8}$$

Capítulo 4

Resultados e Discussões

Este capítulo apresenta e discute os resultados obtidos ao se avaliar o GASS-Metal na busca de sítios metálicos. A Seção 4.1 apresenta os resultados do *teste de sanidade* e a Seção 4.2 apresenta os resultados dos testes com proteínas homólogas. A Seção 4.3 apresenta os testes comparando o GASS-Metal com outros métodos. Por último, a Seção 4.4 apresenta o servidor web implementado com dois recursos disponíveis: (i) busca de sítios metálicos similares utilizando *templates* LIT do M-CSA; (ii) busca de sítios similares *um-para-um*.

4.1 Teste de sanidade

Como já mencionado na Seção 3.3, o *teste de sanidade* utiliza um *template* de sítio metálico de uma proteína para buscar o mesmo sítio na mesma proteína. Em um cenário ideal, o *teste de sanidade* deveria retornar sempre o próprio *template* base como um dos indivíduos da população final e o resultado da função *fitness* para esse indivíduo deveria ser 0 Å. Porém, devido a natureza não-determinística de um AG, nem sempre isso acontece, e os valores dos parâmetros do AG influenciam diretamente nos resultados.

O *teste de sanidade* foi utilizado com dois objetivos: configurar os parâmetros do AG utilizado no GASS-Metal e validar o uso de um AG paralelo em relação ao AG original utilizado pelo GASS.

Uma vez que o espaço de busca (tamanho das proteínas) e o tamanho dos *templates* de sítios metálicos podem variar muito, a ideia era conseguir uma configuração que pudesse retornar bons resultados e ao mesmo tempo não comprometer o tempo de execusão. O objetivo é que o GASS-Metal fucione como um servidor web, e conseguir um tempo de execusão razoável é fundamental para o bom funcionamento da ferramenta. Dessa forma, como descrito na Seção 3.2.4, o AG foi configurado para trabalhar com uma população fixa de 400 indivíduos e um número fixo de 200 gerações. Ao final da execução, são selecionados os 100 melhores indivíduos da população final (ranking 100).

Após a definição dos parâmetros via *teste de sanidade* foi a vez de validar o uso de

um AG paralelo em relação ao AG original. Um teste utilizando a versão original do AG (GASS-WEB) com a população inicial sendo formada de maneira aleatória, e um teste com a nova versão do AG (paralelo) foram executados para se obter os valores finais (percentagem de acertos).

Ao todo foram 928 *templates* provenientes de 320 proteínas LIT do M-CSA. Cada *template* foi utilizado para buscar a si próprio na proteína de origem. O AG foi executado 30 vezes para cada um dos 928 templates. O teste envolvendo o AG original encontrou corretamente o sítio metálico em 76,92% nas 27.840 execuções realizadas. Já o novo AG paralelo encontrou corretamente o sítio metálico em 94,67% no mesmo número de execuções.

A Tabela 4.1 apresenta os resultados do *teste de sanidade* com o AG original agrupados por tipo de íon metálico. O percentual de acertos corresponde ao número de vezes em que o AG encontrou corretamente o sítio metálico. Como exemplo, ao utilizar os *templates* de Zn (193 *templates*), o AG obteve um percentual de acertos de 95,21%. Foram no total 5.790 execuções do AG. Utilizando os *templates* de Mg (179 *templates*), o AG obteve um percentual de acertos de 64,61% em 5.370 execuções.

Na Tabela 4.2 estão os resultados do *teste de sanidade* com o AG paralelo também agrupados por tipo de íon metálico. Como exemplo, ao utilizar os *templates* de Zn (193 *templates*), o AG obteve um percentual de acertos de 99,66%. Foram no total 5.790 execuções do AG praticamente sem erros. Utilizando os *templates* de Mg (179 *templates*), o AG obteve um percentual de acertos de 90,6% em 5.370 execuções. Com este teste fica claro a melhoria nos resultados utilizando o AG paralelo.

Tabela 4.1: Resultados do teste de sanidade - AG original.

Metal	Quantidade de Templates	Fitness Médio	Posição Média no Ranking	Posição Máxima no Ranking	Acertos (%)
Al	3	0	1	1	100
Ca	118	2,36	1,10	8	62,57
Cd	1	0	1	1	100
Co	3	1,03	1	1	100
Cu	30	5,07	1,27	26	89,78
Fe	203	3,45	2,77	98	88,41
Gd	2	0	1	1	100
Hg	2	0,04	1	1	16,67
K	54	2,01	1,52	33	56,23
Mg	179	1,61	2,42	94	64,61
Mn	85	4,74	1,81	22	65,53
Na	40	0,89	1,02	2	59,67
Ni	10	1,27	1,15	6	81,37
Sb	1	0	1	1	100
U	4	0	1	1	1,67
Zn	193	1,99	1,23	46	95,21
Média					76,92

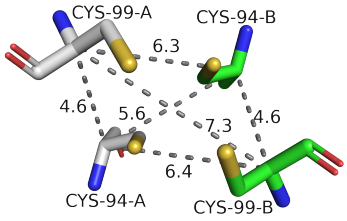
Tabela 4.2: Resultados do teste de sanidade - AG paralelo.

Metal	Quantidade de Templates	Fitness Médio	Posicão Média no Ranking	Posição Máxima no Ranking	Acertos (%)
Al	3	0	1	1	100
Ca	118	0,55	1,08	15	92,01
Cd	1	0	1	1	100
Co	3	0	1	1	100
Cu	30	1,18	1,06	41	96
Fe	203	0,88	1,29	76	98,03
Gd	2	0	1	1	100
Hg	2	0,01	1,05	2	100
K	54	0,7	1,35	36	78,7
Mg	179	0,44	1,37	56	90,6
Mn	85	0,89	1,43	75	94,84
Na	40	0,07	1,03	3	92,92
Ni	10	0,25	1,75	63	94,71
Sb	1	0	1	1	100
U	4	3,92	3,87	98	47,5
Zn	193	0,48	1,07	52	99,66
Média					94,67

As Tabelas 4.1 e 4.2 também apresentam os valores de *fitness* médio, posição média e a posição máxima no ranking (posição na população final onde a solução foi encontrada). A variação nesses valores acontece pelo motivo de que o GASS-Metal pode encontrar um sítio correto que esteja em uma outra cadeia.

Diferente de outros métodos, o GASS-Metal consegue encontrar sítios metálicos similares mesmo em proteínas com mais de uma cadeia e até mesmo sítios interdomínio. Um exemplo de sítio interdomínio é o da proteína 3NOS (*Human Endothelial Nitric Oxide Synthase With Arginine Substrate* - EC: 1.14.13.39), que possui dois resíduos na cadeia A e dois resíduos na cadeia B (Figura 4.1).

Figura 4.1: Sítio de Zn da proteína 3NOS. Em branco os resíduos da cadeia A e em verde os resíduos da cadeia B.



Proteínas podem apresentar várias cadeias com sítios similares. A proteína 3D47 (*Crystal structure of L-rhamnonate dehydratase from Salmonella typhimurium complexed with Mg and D-malate* - EC: 4.2.1.90) possui 8 cadeias (A, B, C, D, E, F, G, H) onde cada uma abriga um sítio de magnésio (Mg) (sítio: ASP, 226; GLU, 252; GLU 280)(Figuras 4.2 e 4.3). No *teste de sanidade* foram utilizados os oito *templates* nessa proteína. Em

uma das execuções utilizando o *template* da cadeia G (Tabela 4.3 - Execução 01) o AG encontrou corretamente o sítio esperado (primeira posição no ranking com *fitness* 0 Å) e mais 4 sítios corretos (posições 2 a 5 com *fitness* abaixo de 1 Å) em outras cadeias (D, B, F e H). Em outra execução com o mesmo *template* (Tabela 4.3 - Execução 02), o AG não encontra o sítio na cadeia G, mas encontra os sítios corretos na cadeia A e D (posições 1 e 2 do ranking). O que não se trata exatamente de um erro do AG, uma vez que os sítios encontrados em outras cadeias estão corretos.

Figura 4.2: Proteína 3D47 com 8 cadeias e 8 *templates* de ferro - Formato Sticks.

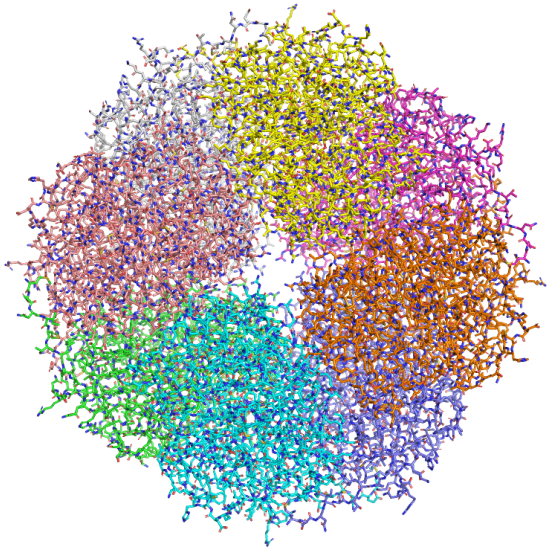
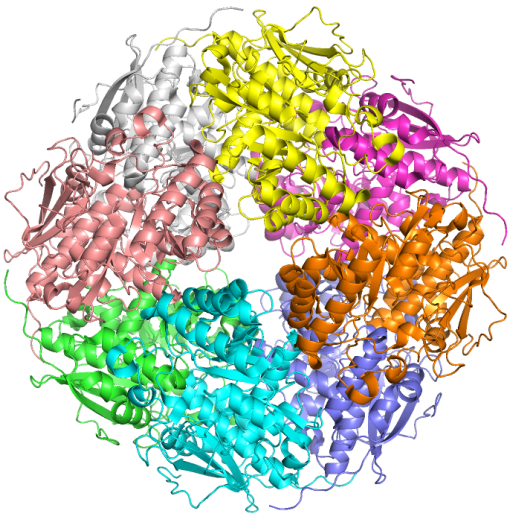


Figura 4.3: Proteína 3D47 com 8 cadeias e 8 *templates* de ferro - Formato Cartoon.



Muitos sítios similares em uma mesma proteína com pequenas diferenças entre os valores de *fitness*, e tipos de resíduos iguais em um mesmo *template* podem guiar o AG para

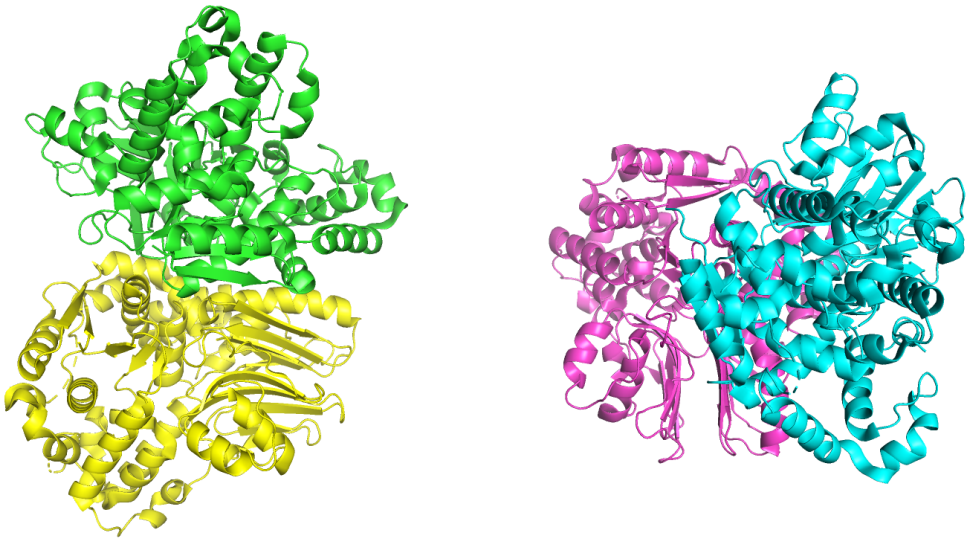
uma outra cadeia. Essa situação acontecia com maior frequência quando o AG utilizava apenas uma única população. Na implementação com uma população dividida (quadrantes - AG paralelo) a mesma situação diminuiu consideravelmente, mas eventualmente acaba ocorrendo.

Tabela 4.3: Resultados do *teste de sanidade* - proteína 3D47.

Execução 01				Execução 02			
Posição	Fitness	Cadeia	Resíduos	Posição	Fitness	Cadeia	Resíduos
1	0	G	ASP 226; GLU 252; GLU 280	1	0,08	D	ASP 226; GLU 252; GLU 280
2	0,08	D	ASP 226; GLU 252; GLU 280	2	0,09	A	ASP 226; GLU 252; GLU 280
3	0,11	B	ASP 226; GLU 252; GLU 280	3	3,18	B	ASP 302; GLU 280; GLU 252
4	0,16	F	ASP 226; GLU 252; GLU 280	4	3,21	B	ASP 302; GLU 280; GLU 349
5	0,19	H	ASP 226; GLU 252; GLU 280	5	3,24	F	ASP 302; GLU 280; GLU 252
6	3,21	B	ASP 302; GLU 280; GLU 252	6	3,28	D	ASP 302; GLU 280; GLU 252

Um caso em que o AG paralelo não consegue um bom resultado no *teste de sanidade* é com a proteína 1CT9 (*Crystal Structure of asparagine synthetase B from Escherichia Coli* - EC: 6.3.5.4) (Figura 4.4). São 4 cadeias (A, B, C e D) contendo um sítio com 4 íons de urânio (U) em cada cadeia.

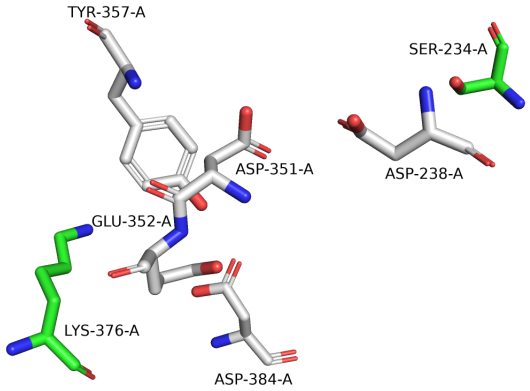
Figura 4.4: Proteína 1CT9 - Cadeias A (verde), B (ciano), C (magenta) e D (amarelo).



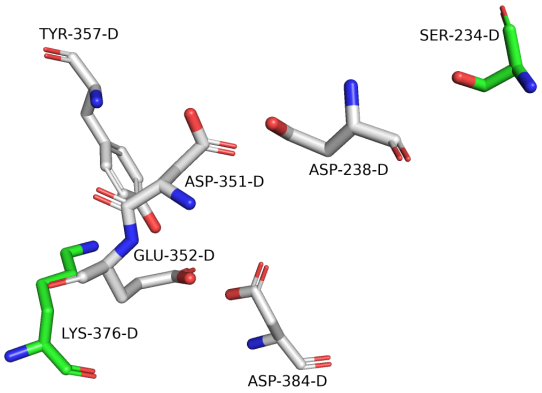
Segundo dados do MetalPDB (Putignano et al., 2017), os sítios das cadeias A e D são formados pelos resíduos ASP 238, ASP 351, GLU 352, TYR 357 e ASP 384 (Figura 4.5). Já o sítio da cadeia B é formado pelos resíduos SER 234, ASP 238, ASP 351, GLU 352, TYR 357 e ASP 384, enquanto na cadeia C tem-se os resíduos ASP 238, ASP 351, GLU 352, TYR 357, LYS 376 e ASP 384 (Figura 4.6).

Embora os resíduos SER 234 e LYS 376 (resíduos em verde na Figura 4.5) não façam parte dos sítios das cadeias A e D, ambos estão situados em conformidade com os mesmos resíduos nas cadeias B e C.

Figura 4.5: Sítios metálicos de urânio da proteína 1CT9 (Cadeias A e D). Os resíduos em verde não fazem parte dos sítios conforme dados do MetalPDB (Putignano et al., 2017).



(a) Sítio - Cadeia A



(b) Sítio - Cadeia D

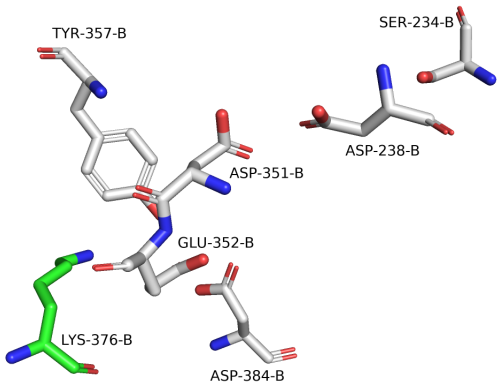
Situação semelhante acontece com as cadeias B e C. Conforme o MetalPDB (Putignano et al., 2017), na cadeia B a LYS 376 não faz parte do sítio metálico, e na cadeia C é a SER 234 que não faz parte (resíduos em verde na Figura 4.6).

Contudo, existe uma similaridade envolvendo as distâncias dos resíduos dos 4 sítios. Por exemplo, as distâncias entre os resíduos LYS 376 e ASP 384 nas cadeias A, B, C e D, são, respectivamente 10,1 Å, 10 Å, 9,9 Åe 10,2 Å. No geral, as distâncias entre os resíduos não variam mais que 1 Å.

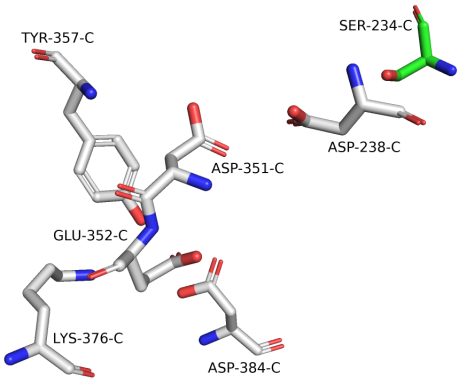
O motivo para esse resultado com a proteína 1CT9 está associado à divisão em quadrantes para a geração das populações iniciais dos AGs (paralelo). Novamente, são 4 cadeias (A, B, C e D) contendo um sítio com 4 íons de urânio (U) em cada cadeia. Contudo, a divisão em quadrantes (baseado no centróide da proteína) acaba por gerar quadrantes muito diferentes em quantidade de resíduos, impactando negativamente para o bom funcionamento do AG com esta proteína (Figura 4.7).

Uma vez que o GASS-Metal não trabalha com cadeias separadamente (para que possa encontrar os sítios interdomínio), uma alternativa seria avaliar novas formas da geração da população inicial que não apenas o uso do centróide/quadrantes.

Figura 4.6: Sítios metálicos de U da proteína 1CT9. (Cadeias B e C). Os resíduos em verde não fazem parte dos sítios conforme dados do MetalPDB (Putignano et al., 2017).

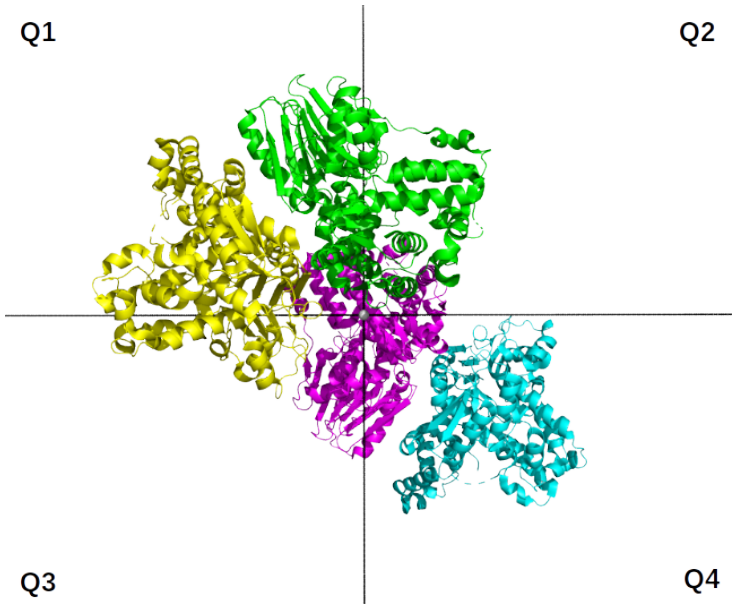


(a) Sítio - Cadeia B



(b) Sítio - Cadeia C

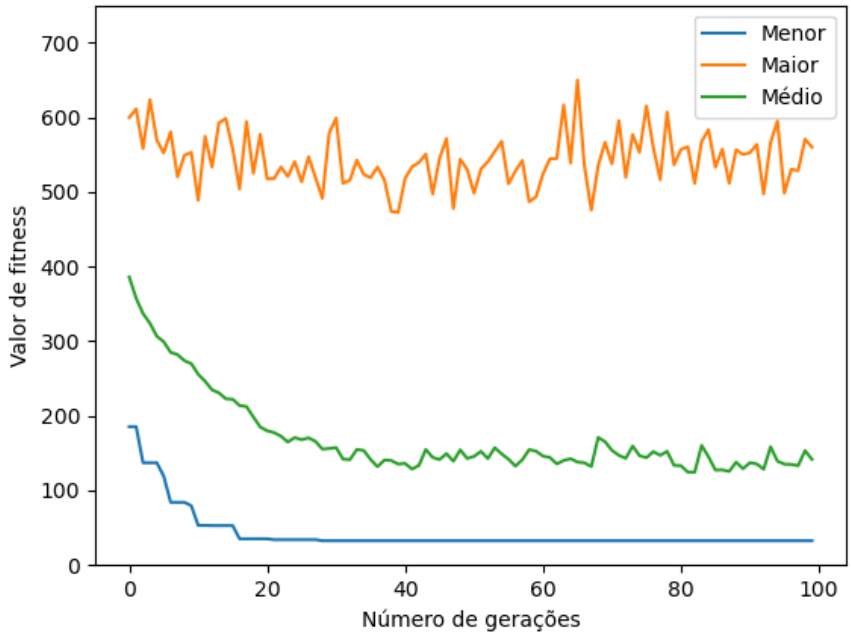
Figura 4.7: Proteína 1CT9 com o centróide (esfera cinza no centro) e seus quadrantes.



Uma análise importante a ser feita envolve a convergência dos indivíduos do AG durante as gerações. A Figura 4.8 traz as informações referentes ao *teste de sanidade* feito

na proteína 1EUU com o GASS original, sem a abordagem paralela. Neste caso é possível observar que o menor valor de *fitness* (linha azul) nunca chega a zero, tendo o menor valor alcançado de 32,57.

Figura 4.8: Indivíduos do AG e seus menores, maiores e valores médio de *fitness* durante as gerações do *teste de sanidade* da proteína 1EUU.

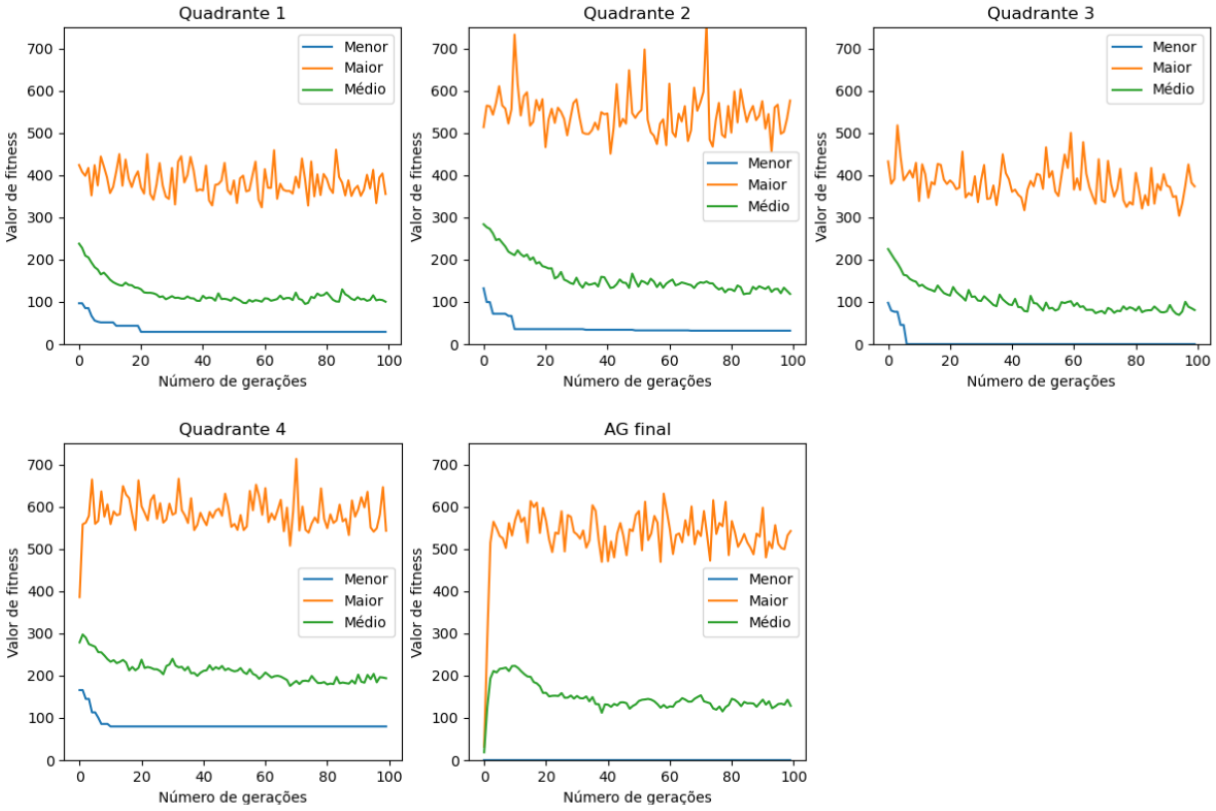


A Figura 4.9 mostra as execuções paralelas e do AG final do mesmo *teste de sanidade* da proteína 1EUU e seus valores de *fitness* dos indivíduos. Neste caso, o GASS-Metal encontra o sítio metálico correto na busca por quadrantes (mais precisamente ainda no quadrante 3). Mesmo já tendo encontrado o menor valor em uma das execuções paralelas, o AG final é executado, pois em alguns casos não é possível encontrar o melhor indivíduo nas execuções anteriores.

Tanto na execução paralela quanto na tradicional, os valores de *fitness* máximo (linha laranja) e médio (linha verde) sofrem alterações que variam para cima ou para baixo no decorrer das gerações. Isto se deve às mutações que ocorrem nos indivíduos, fazendo com que seu valor de *fitness* seja alterado tanto para mais quanto para menos. As gerações mostradas foram fixadas no número 100 pois neste caso a convergência ocorre antes do valor padrão do GASS-Metal de 200 gerações. Existem casos onde é necessário um número maior de iterações dos AGs.

Com isso é possível observar mais uma vez a importância da abordagem paralela do problema. A solução proposta de dividir a proteína em quadrantes e realizar a busca por partes faz com que seja possível buscar resíduos em regiões que a abordagem original do AG não conseguia.

Figura 4.9: Indivíduos do AG e seus menores, maiores e valores médio de *fitness* durante as gerações do *teste de sanidade* da proteína 1EUU.



4.2 Proteínas homólogas

A pergunta que se pretende responder com este teste é: O GASS-Metal pode encontrar sítios metálicos similares em proteínas homólogas (proteínas que derivam de um *ancestral comum*)? Para este teste foram selecionados 8 *templates* de diferentes íons metálicos e suas respectivas proteínas homólogos de maneira aleatória, conforme Tabela 3.1 na Seção 3.3. No total, foram selecionadas 649 proteínas homólogas com mesmo número EC de seus respectivos *templates* (Seção 3.1.2).

Foram realizadas 30 execuções do AG para cada proteína homóloga. A Tabela 4.4 apresenta os resultados do experimento. Além do percentual de acertos, também são apresentados os valores de *fitness* médio, posição média no ranking e posição máxima no ranking. Por exemplo, ao utilizar o *template* de Zn (3NOS), o AG teve um percentual de acerto de 100% em 11.130 execuções.

Este teste também serviu para avaliar a importância da matriz de substituição utilizada pelo AG do GASS-Metal. Na Tabela 4.4 estão anotados os resultados com e sem a utilização da matriz de substituição. Ao utilizar o *template* de Fe (3PCA) sem a matriz de substituição, o AG teve um percentual de acerto de 85,19% de acerto em 810 execuções. Com o uso da matriz de substituição, o percentual foi de 99,51%. Em todos os casos

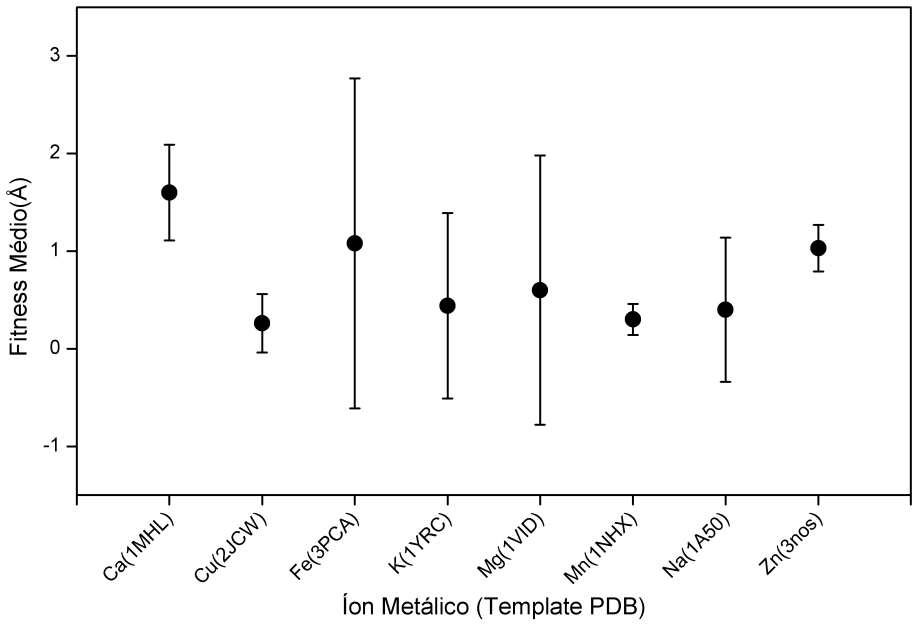
Tabela 4.4: Resultados do teste com proteínas homólogas.

Metal	*Template*	Homólogas	Mutações Conservativas	Fitness Médio	Posição Média no Ranking	Posição Máxima no Ranking	Acertos (%)
Ca	1MHL	72	N	1,6	1	2	100
Cu	2JCW	43	S	0,26	1,5	78	99,38
Cu	2JCW	43	N	0,24	1,33	6	97,67
Fe	3PCA	27	S	1,08	6,63	35	99,51
Fe	3PCA	27	N	1,21	1,33	25	85,19
K	1YRC	55	S	0,44	1,5	78	95,09
K	1YRC	55	N	0,38	1,05	2	90,85
Mg	1VID	41	S	0,6	2,24	29	100
Mg	1VID	41	N	0,29	1,06	2	88,89
Mn	1NHX	22	N	0,3	1	1	100
Na	1A50	18	S	0,4	2,09	73	92,78
Na	1A50	18	N	0,27	1,06	2	88,89
Zn	3NOS	371	N	1,03	1	1	100

onde havia a possibilidade de utilizar a matriz de substituição, o AG obteve um resultado melhor do que se sua utilização (Tabela 4.4).

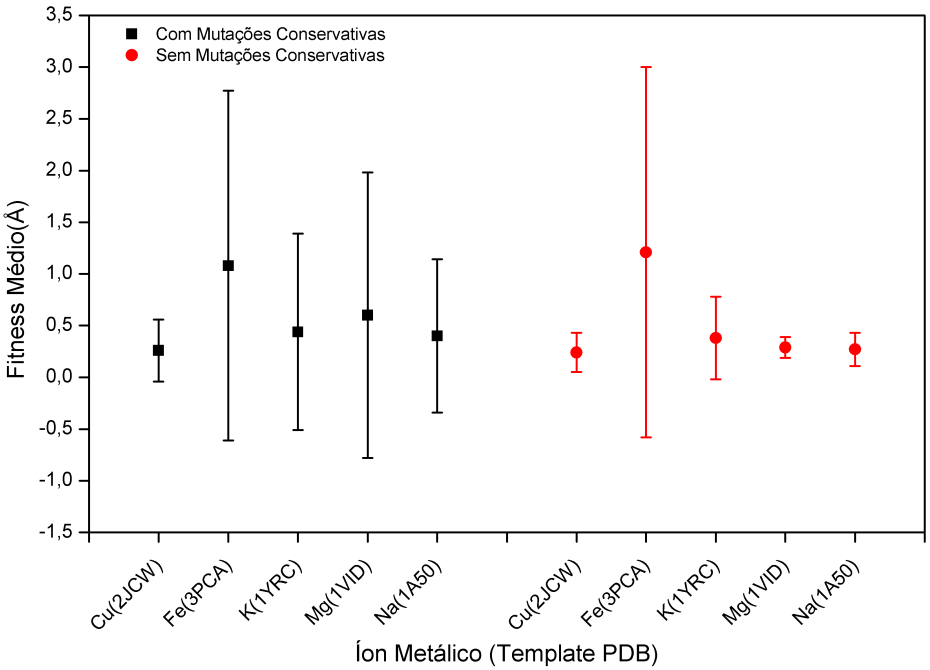
A Figura 4.10 mostra o desvio padrão do valor de *fitness*. Os *templates* de Cu, Fe, K, Mg e Na, tiveram um desvio padrão maior devido ao uso da matriz de substituição. A Figura 4.11 apresenta uma comparação entre o desvio padrão dos *templates* com mutações conservativas e o *templates* sem mutação. Pode-se perceber que ao utilizar as mutações conservativas, o valor do desvio padrão aumenta. Sem as mutações, o desvio padrão é menor, mas os resultados (percentual de acertos) diminuem.

Figura 4.10: Desvio padrão do valor de *fitness* do teste com proteínas homólogas.



As Figuras 4.12 e 4.13 mostram os indivíduos do GASS-Metal durante as gerações em cada um dos AGs paralelos e também no final, ambas utilizando como *template* o sítio

Figura 4.11: Comparação do desvio padrão do valor de *fitness* entre *templates* com e sem mutações conservativas.



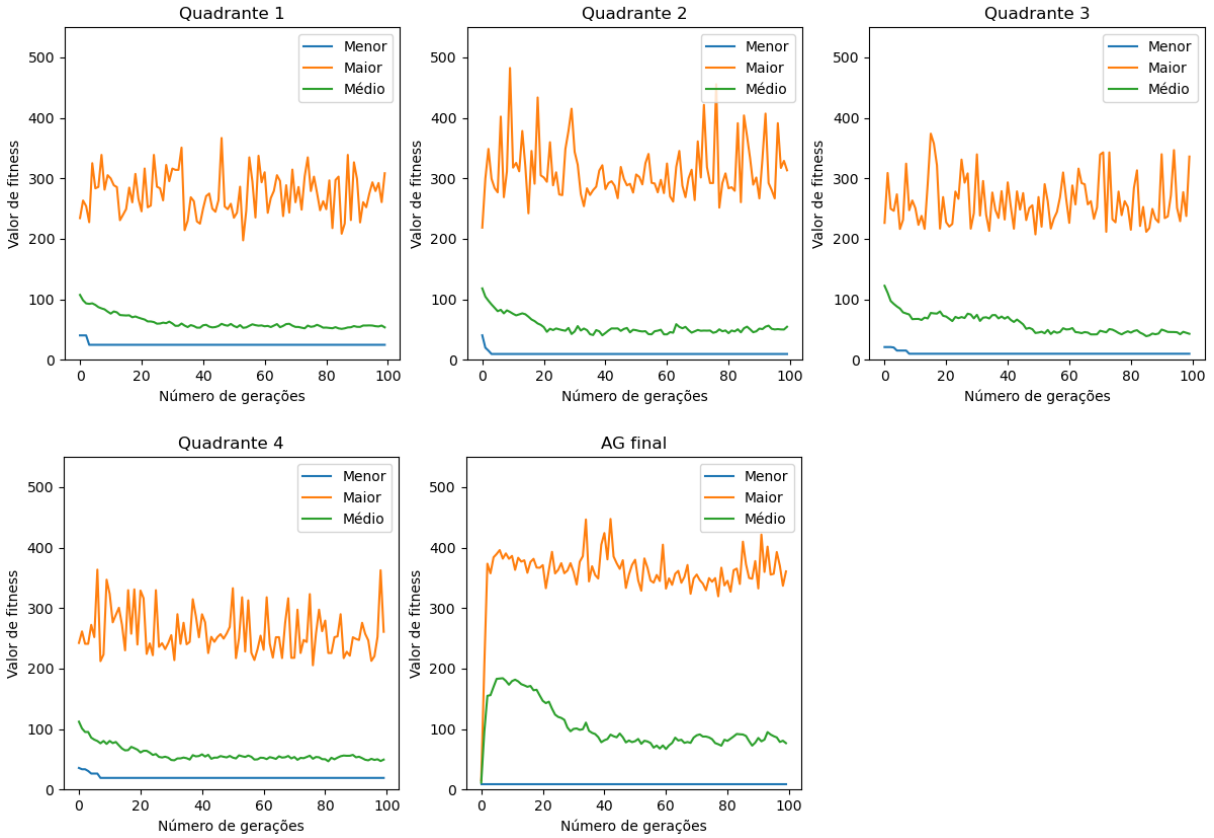
da proteína 1YRC (*X-ray Crystal Structure of Hydrogenated Cytochrome P450cam* - EC: 1.14.15.1) em uma de suas homólogas, a 2GQX (*Crystal structure of cytochrome P450cam Mutant (f87w/y96f/l244a/v247l/c334a) with Pentachlorobenzene* - EC: 1.14.15.1).

A Figura 4.12 não utiliza de mutações conservativas de resíduos e por isto é possível observar que seu menor valor de *fitness* encontrado em cada geração (linha azul) não se aproxima do valor zero (o mínimo encontrado é 9,38) durante toda a execução. Isto indica que os sítios encontrado não são similares e de fato, por não usar substituições, o GASS-Metal não consegue encontrar o sítio da proteína homóloga.

Já a Figura 4.13 utiliza mutações conservativas, e é possível observar que tanto no quadrante 1 como no quadrante 3 o GASS-Metal consegue encontrar os menores valores de *fitness* bem próximos de zero (0,21 para ser mais preciso) indicando que o sítio encontrado é similar ao *template*. Ao verificar este menor sítio encontrado confirma-se como o sítio da proteína homóloga buscado. Neste exemplo, mesmo já tendo encontrado o sítio correto nos AGs paralelos o AG final foi executado. Mas é importante frisar que existem casos onde não é possível encontrar os sítios durante as execuções paralelas, precisando assim de uma última execução.

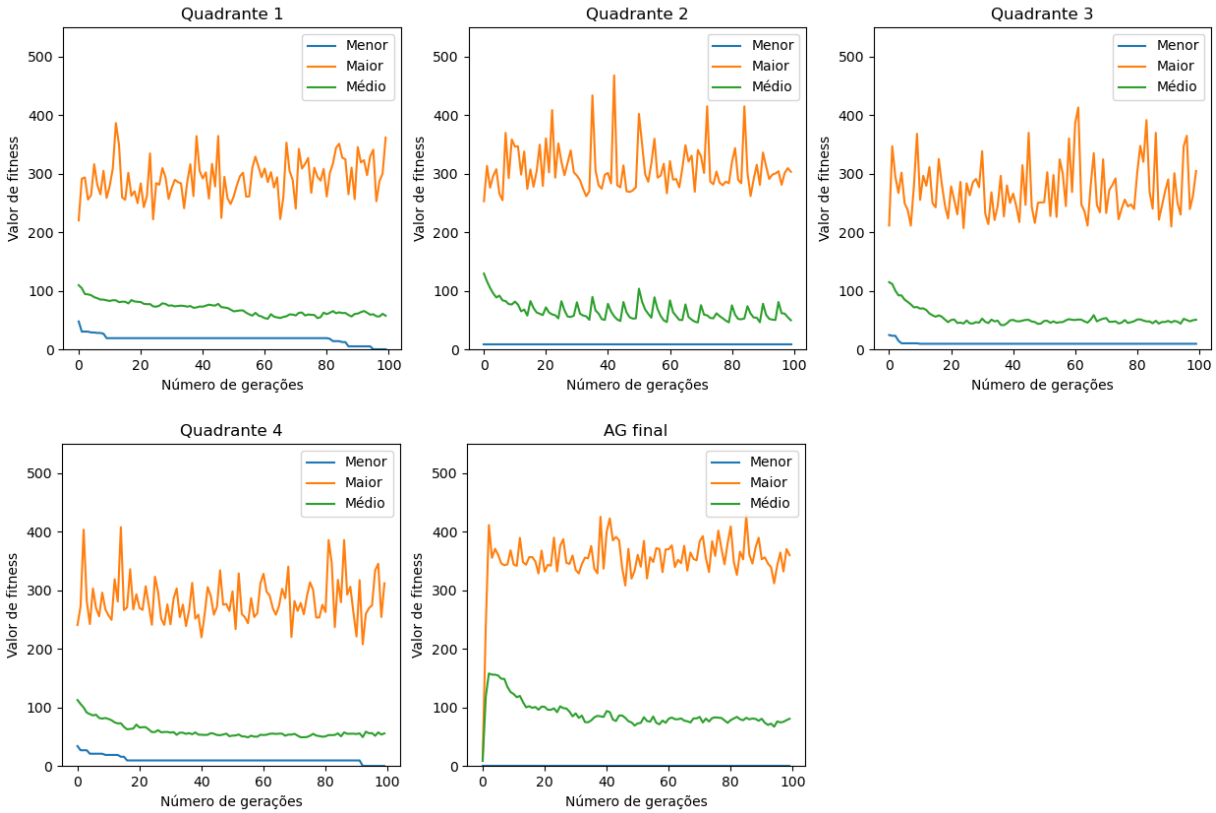
Em ambos os casos (com e sem substituições) é possível observar que os valores de *fitness* maior (linha laranja) e *fitness* médio (linha verde) convergem acompanhando o melhor indivíduo no decorrer das gerações. Os picos e variações que ocorrem são explicados pelo operador de mutação dos AGs que faz com que certos resíduos dos indivíduos sejam substituídos por outros resíduos aleatórios dos repositórios. Isso faz com que os novos indivíduos gerados possam ter um valor de *fitness* diferente a cada geração, ficando maior

Figura 4.12: Gráficos comparativos dos menores, maiores e valores médios de *fitness* dos indivíduos dos AGs durantes as gerações. Resultados obtidos utilizando o *template* da proteína 1YRC para a busca do sítio metálico de sua proteína homóloga 2GQX, sem o uso de mutações conservativas.



ou menor em relação à geração anterior. Assim como mostrado na Seção 4.1, as gerações mostradas nos gráficos foram limitadas a 100, pois neste exemplo o GASS-Metal já alcançou a convergência neste número de gerações. Em outras proteínas pode ser necessário um número maior de gerações para isto.

Figura 4.13: Gráficos comparativos dos menores, maiores e valores médios de *fitness* dos indivíduos dos AGs durantes as gerações. Resultados obtidos utilizando o *template* da proteína 1YRC para a busca do sítio metálico de sua proteína homóloga 2GQX, com o uso de mutações conservativas.



4.3 GASS-Metal comparado a métodos *estado da arte* para a predição de sítios metálicos

Esta seção avalia o GASS-Metal de acordo com as métricas descritas na Seção 3.4 e o compara com diversos outros métodos estado da arte. Os métodos de predição de sítios metálcicos utilizados para a comparação com o GASS-Metal foram: MetalDetector, S-SITE, TargetS, IonSeq, MIB, COACH e IonCom, todos descritos na Seção 2.2. Este teste utiliza o cojunto de dados CD3, mencionado na Seção 3.3, e tem como objetivo analisar como o GASS-Metal se comporta na predição de sítios metálicos em relação a outros preditores da literatura.

Como o MetalPDB e o BioLiP utilizam métricas diferentes para a definição dos resíduos que compõem um sítio metálico (Seções 2.3.2 e 2.3.4), para que os testes ficassem justos, tanto com o GASS-Metal quanto com os demais algoritmos, novos *templates* foram definidos, agora utilizando as mesmas métricas do BioLiP. Dessa forma, para cada proteína LIT que contém um íon metálico, ao invés de buscar seus respectivos sítios metálicos no MetalPDB, usou-se a mesma métrica de cálculo de sítios metálicos utilizado

pelo BioLiP, calculando as distâncias de cada resíduo em relação aos íons que a proteína contém. Aqueles resíduos que estiverem dentro da distância definida fazem parte do sítio metálico, formando assim novos *templates* do GASS-Metal.

Após a definição dos novos *templates*, o GASS-Metal foi executado para cada grupo de íons metálicos do conjunto de dados CD3. Os resultados dos metais Zn^{2+}, Ca^{2+}, Mg^{2+} e Mn^{2+} podem ser vistos na Tabela 4.5, os resultados dos metais Fe^{3+}, Cu^{2+}, Fe^{2+} e Co^{2+} aparecem na Tabela 4.6 e a Tabela 4.7 traz os resultados dos metais Na^{+}, K^{+}, Cd^{2+} e Ni^{2+}.

Os valores dos resultados estão dispostos de três maneiras diferentes: (1) 10 melhores do ranking, (2) 100 melhores do ranking e (3) todos os resultados obtidos na busca. Isso se explica através de três motivos principais. O primeiro é que pelo fato do GASS-Metal trabalhar com distâncias e ordená-los de acordo com seu valor de *fitness*, é interessante mostrar como são os 10 indivíduos mais bem ranqueados, visando a experiência que o usuário terá ao utilizar o servidor web (que será abordado na Seção 4.4). Os 10 primeiros sítios candidatos tendem a ser mais observados ao avaliar os resultados, e mesmo utilizando apenas esse pequeno conjunto, pode-se notar uma boa performance do GASS-Metal em relação a vários métodos, ganhando inclusive de todos os avaliados em alguns tipos de íons metálicos, como os de Co^{2+}, Na^{+} e Ni^{+}.

A escolha por informar os resultados dos testes do GASS-Metal dos indivíduos ranqueados no top 100 é justificada observando os *testes de sanidade* (Seção 4.1). Neste teste, que avaliou a taxa de acertos que o GASS-Metal ao utilizar um *template* de sítio metálico para buscar o mesmo sítio na própria proteína, mostrou também a posição mais elevada onde o sítio foi encontrado no *ranking*. Como visto nos testes, a posição máxima no *ranking* entre todos os *templates* de todos os metais avaliados foi de 98 (resultado obtido em um *template* de Fe). Todos os sítios encontrados no teste de sanidade estão dentro do intervalo entre 1 a 98, e por isso foi pertinente informar os resultados do teste com os outros métodos aqui descritos em um intervalo próximo, tendo sido fixado o valor de 100 posições.

Já a opção por trazer os resultados de todos os indivíduos encontrados se deve ao fato de mostrar a relação dos resultados entre o *target* e todos os *templates*. Soma-se ainda o fato de que, sejam dois sítios metálicos encontrados, A e B, caso o sítio A tenha um valor de *fitness* menor que B, não necessariamente o sítio A será mais próximo ao *target* em relação ao sítio B. O valor de *fitness* é calculado sobre os *templates* do GASS-Metal e, portanto, um sítio encontrado pode estar mais distante a um *template*, porém, mais próximo ao *target* do experimento. Isso faz com que sua posição no ranking nem sempre indique o quão melhor um resultado é em relação aos demais. Em todos as instâncias de resultados, os indivíduos encontrados são avaliados em relação a um *target* (sítio metálico definido como correto no experimento) com as métricas descritas na Seção 3.4.

Tabela 4.5: Resultados da comparação entre GASS-Metal com outros métodos *estado da arte* dos metais Zn^{2+}, Ca^{2+}, Mg^{2+} e Mn^{2+}.

Íon Metálico	Preditor	Sen (%)	Spe (%)	Acc (%)	MCC
Zn^{2+}	MetalDetector	38.26	99.83	98.22	0.565
	S-SITE	45.14	97.88	96.51	0.387
	TargetS	41.70	99.80	98.29	0.588
	IonSeq	70.04	92.56	91.97	0.347
	MIB	40.12	99.07	97.53	0.451
	COACH	32.39	99.37	97.62	0.422
	IonCom	76.72	95.59	95.10	0.474
	MionSite	70.65	99.68	98.92	0.771
	GASS-Metal (1)	52.14	99.35	98.75	0.515
	GASS-Metal (2)	65.19	99.49	98.96	0.647
	GASS-Metal (3)	75.96	99.68	99.32	0.756
Ca^{2+}	MetalDetector	0.62	99.86	98.03	0.016
	S-SITE	10.17	99.51	97.87	0.159
	TargetS	19.71	99.70	98.23	0.322
	IonSeq	0.00	100.0	98.16	N/A
	MIB	17.61	99.17	97.68	0.213
	COACH	16.18	97.82	96.32	0.122
	IonCom	28.01	99.47	98.16	0.365
	MionSite	42.53	99.71	98.66	0.552
	GASS-Metal (1)	9.38	98.84	97.60	0.081
	GASS-Metal (2)	22.93	99.00	97.96	0.219
	GASS-Metal (3)	29.05	99.09	98.15	0.281
Mg^{2+}	MetalDetector	1.57	99.85	98.31	0.043
	S-SITE	35.02	97.28	96.31	0.227
	TargetS	15.16	99.84	98.52	0.298
	IonSeq	0.00	100.0	98.44	N/A
	MIB	22.21	99.33	98.12	0.268
	COACH	21.12	97.60	96.41	0.144
	IonCom	24.12	99.25	98.07	0.276
	MionSite	24.35	99.69	98.51	0.362
	GASS-Metal (1)	13.05	99.34	98.75	0.124
	GASS-Metal (2)	33.88	99.54	99.05	0.333
	GASS-Metal (3)	55.74	99.69	99.36	0.550
Mn^{2+}	MetalDetector	21.05	99.72	98.69	0.319
	S-SITE	69.47	98.38	98.01	0.493
	TargetS	28.42	99.75	98.82	0.408
	IonSeq	2.11	99.94	98.67	0.081
	MIB	47.75	99.52	98.84	0.514
	COACH	27.01	99.82	98.87	0.420
	IonCom	54.06	99.39	98.80	0.534
	MionSite	57.27	99.40	98.84	0.558
	GASS-Metal (1)	19.45	99.45	98.70	0.188
	GASS-Metal (2)	43.75	99.60	99.10	0.433
	GASS-Metal (3)	55.20	99.60	99.35	0.559

Tabela 4.6: Resultados da comparação entre GASS-Metal com outros métodos *estado da arte* dos metais Fe^{3+}, Cu^{2+}, Fe^{2+} e Co^{2+}.

Íon Metálico	Preditor	Sen (%)	Spe (%)	Acc (%)	MCC
Fe^{3+}	MetalDetector	28.57	99.76	99.24	0.360
	S-SITE	90.48	98.78	98.71	0.560
	TargetS	28.57	99.55	99.03	0.296
	IonSeq	80.95	96.85	96.73	0.350
	MIB	52.63	97.83	97.49	0.277
	COACH	77.52	99.73	99.57	0.725
	IonCom	77.12	99.80	99.64	0.756
	MionSite	82.90	99.75	99.63	0.765
	GASS-Metal (1)	73.40	99.97	99.85	0.733
	GASS-Metal (2)	92.00	99.97	99.85	0.920
	GASS-Metal (3)	92.00	99.97	99.85	0.920
Cu^{2+}	MetalDetector	58.33	99.91	99.47	0.712
	S-SITE	25.00	97.41	96.64	0.138
	TargetS	-	-	-	-
	IonSeq	41.67	99.11	98.50	0.365
	MIB	83.33	98.03	97.88	0.503
	COACH	50.00	97.14	96.64	0.268
	IonCom	50.00	99.55	99.03	0.517
	MionSite	48.02	99.63	99.08	0.523
	GASS-Metal (1)	67.00	99.93	99.85	0.666
	GASS-Metal (2)	67.00	99.93	99.85	0.666
	GASS-Metal (3)	67.00	99.93	99.85	0.666
Fe^{2+}	MetalDetector	33.33	99.86	99.04	0.496
	S-SITE	66.67	95.54	95.18	0.309
	TargetS	-	-	-	-
	IonSeq	97.80	99.28	99.26	0.782
	MIB	100.00	99.44	99.45	0.830
	COACH	66.67	98.05	97.66	0.437
	IonCom	100.00	99.44	99.45	0.830
	MionSite	94.98	99.56	99.50	0.830
	GASS-Metal (1)	78.00	99.67	99.67	0.775
	GASS-Metal (2)	89.00	99.67	99.67	0.887
	GASS-Metal (3)	89.00	99.67	99.67	0.887
Co^{2+}	MetalDetector	16.67	99.58	98.69	0.217
	S-SITE	55.21	84.68	84.36	0.113
	TargetS	-	-	-	-
	IonSeq	-	-	-	-
	MIB	33.33	95.48	94.81	0.138
	COACH	53.21	91.56	91.14	0.162
	IonCom	-	-	-	-
	MionSite	58.77	92.58	92.22	0.195
	GASS-Metal (1)	75.00	99.95	99.90	0.749
	GASS-Metal (2)	75.00	99.95	99.90	0.749
	GASS-Metal (3)	75.00	99.95	99.90	0.749

Tabela 4.7: Resultados da comparação entre GASS-Metal com outros métodos *estado da arte* dos metais Na^+, K^+, Cd^{2+} e Ni^{2+}.

Íon Metálico	Preditor	Sen (%)	Spe (%)	Acc (%)	MCC
Na^+	MetalDetector	0.00	100.0	94.57	NA
	S-SITE	0.00	88.62	83.85	-0.083
	TargetS	-	-	-	-
	IonSeq	0.00	89.43	84.62	-0.080
	MIB	-	-	-	-
	COACH	0.00	94.31	89.23	-0.057
	IonCom	0.00	86.18	81.54	-0.093
	MionSite	14.29	94.31	90.00	0.081
	GASS-Metal (1)	40.00	97.58	95.35	0.376
	GASS-Metal (2)	40.00	97.58	95.35	0.376
	GASS-Metal (3)	40.00	97.58	95.35	0.376
K^+	MetalDetector	0.00	99.76	95.47	-0.010
	S-SITE	31.49	94.66	91.95	0.216
	TargetS	-	-	-	-
	IonSeq	31.15	96.00	93.21	0.249
	MIB	-	-	-	-
	COACH	35.15	96.48	93.84	0.298
	IonCom	35.14	97.94	95.24	0.366
	MionSite	36.57	97.82	95.18	0.371
	GASS-Metal (1)	0.00	98.18	96.42	-0.018
	GASS-Metal (2)	0.00	98.18	96.42	-0.018
	GASS-Metal (3)	20.00	98.54	97.13	0.185
Cd^{2+}	MetalDetector	0.00	99.75	95.47	-0.009
	S-SITE	14.29	91.92	89.27	0.041
	TargetS	-	-	-	-
	IonSeq	-	-	-	-
	MIB	35.71	92.68	90.73	0.187
	COACH	7.14	97.69	94.55	0.057
	IonCom	-	-	-	-
	MionSite	9.26	98.23	95.19	0.097
	GASS-Metal (1)	11.11	98.54	97.13	0.097
	GASS-Metal (2)	11.11	98.54	97.13	0.097
	GASS-Metal (3)	11.11	98.54	97.13	0.097
Ni^{2+}	MetalDetector	50.00	100.0	98.17	0.700
	S-SITE	0.00	94.29	90.83	-0.047
	TargetS	-	-	-	-
	IonSeq	-	-	-	-
	MIB	100.00	94.29	94.50	0.614
	COACH	0.00	95.24	91.74	-0.043
	IonCom	-	-	-	-
	MionSite	25.00	99.05	96.33	0.337
	GASS-Metal (1)	75.00	99.87	99.75	0.749
	GASS-Metal (2)	75.00	99.87	99.75	0.749
	GASS-Metal (3)	75.00	99.87	99.75	0.749

Falando mais especificamente do desempenho do GASS-Metal nos testes, as análises aqui realizadas levaram em consideração principalmente os valores de MCC, pois é uma métrica que consegue informar melhor a qualidade dos resultados. Os resultados da Tabela 4.5 mostram os íons metálicos Zn^{2+}, Ca^{2+}, Mg^{2+} e Mn^{2+} e é possível observar que, em relação ao MCC, o GASS-Metal obtém uma performance melhor do que todos os métodos nos testes dos íons Mg^{2+} e Mn^{2+}, a segunda colocação ao buscar *targets* de Zn^{2+} e a quarta colocação nos testes com íons de Ca^{2+}. Tudo isso levando em consideração os resultados de número (3), que observam todos os sítios encontrados na busca.

Já os resultados que analisam os 10 e 100 melhores indivíduos ranqueados de acordo com seu valor de Zn^{2+}, instâncias (1) e (2), o GASS-Metal ainda tem resultados satisfatórios, tendo valores próximos aos melhores métodos em praticamente todos os íons metálicos. Como informado anteriormente, não necessariamente um sítio bem ranqueado de acordo com seu valor de *fitness* corresponde a um indivíduo similar ao *target* do experimento. Neste caso demonstra-se este tipo de situação, para observar que o GASS-Metal ainda consegue entregar bons resultados em relação a outros métodos estado da arte, mesmo analisando uma parte de seus resultados.

A Tabela 4.6 contém os resultados dos testes realizados com os íons de Fe^{3+}, Cu^{2+}, Fe^{2+} e Co^{2+}. Nestes testes, o GASS-Metal obteve resultados ainda melhores. Nos testes com sítios metálicos de Fe^{3+}, o GASS-Metal obteve a melhor colocação entre os preditores ao analisar os 100 primeiros e todos os sítios candidatos (resultados (2) e (3)). E mesmo levando em consideração apenas os 10 primeiros indivíduos, obteve a terceira melhor colocação entre os preditores. Nos testes de Cu^{2+}, Fe^{2+} e Co^{2+} também foram observados bons resultados. O GASS-Metal conseguiu o segundo maior valor de MCC nos testes com Cu^{2+} em todas as instâncias de resultados e a primeira e segunda melhor performance nos testes com Fe^{2+} (nas instâncias (2), (3) e (1), respectivamente).

Os testes com sítios de Co^{2+} mostraram resultados bem superiores em relação a todos os outros métodos de predição com valor de MCC de 0.749. Os casos onde os valores das instâncias de resultados que ficaram iguais ocorrem porque foram encontrados os melhores indivíduos em posições mais baixas do *ranking*, pois sejam os resultados (1), (2) e (3) definidos como conjuntos, (1) pertence a (2) que pertence a (3). Os dados reportados nesta Tabela 4.6 mostram também que alguns métodos não realizam a predição de sítios metálicos de muitos tipos diferentes de íons. O GASS-Metal, por utilizar apenas informações de distâncias entre resíduos, consegue performar em sítios metálicos de todos os tipos, desde que haja *templates* para os íons metálicos.

Vale ressaltar que, como visto na Tabela 3.2 da Seção 3.3, a quantidade de resíduos pertencentes aos sítios metálicos aqui gerados, para todos os íons da Tabela 4.6, é igual ao conjunto utilizado pelos autores do MIonSite em seu teste. Isso não garante que são exatamente os mesmos resíduos, porém, por se tratar de poucas proteínas e poucos resíduos é bem provável que a grande maioria dos resíduos dos sítios sejam sim iguais.

A Tabela 4.7 traz os resultados obtidos nos testes com os íons metálicos de Na^{+},

K^+, Cd^{2+} e Ni^{2+}. Nas três instâncias de resultados, o GASS-Metal obteve os mesmos resultados nos testes com Na^+ e Ni^{2+} e em ambos obteve resultados superiores a todos os preditores aqui comparados. Já nos testes envolvendo sítios metálicos de K^+, o GASS-Metal obteve resultados não tão bons nas instâncias (1) e (2), porém, ao se analisar todos os indivíduos da busca, obteve-se um valor de MCC igual a 0.185 que se aproxima das demais ferramentas. Os valores obtidos com o íon Cd^{2+} tiveram resultado satisfatório, ficando na segunda posição entre os métodos avaliados.

Uma questão relevante a se destacar nos resultados são os valores altos tanto de especificidade como de acurácia mostrados em praticamente todos os testes com todas os preditores. Isso se deve à natureza do problema e à forma de que os testes foram feitos. Como também constatado pelos autores do MIonSite (Qiao e Xie, 2019), o motivo se dá pelo fato das classes (resíduos que são do sítio metálico e resíduos que não são do sítio metálico) deste problema serem muito desbalanceadas. Proteínas facilmente tem centenas ou até milhares de resíduos, e um sítio metálico muitas vezes não contém sequer uma dezena deles.

Um exemplo deste desbalanceamento ocorre na proteína 5ZM4 (*Fe(II)/(alpha)ketoglutarate-dependent dioxygenase AndA with preandiloid C* - EC: 1.14.11) que contém um total de 1140 resíduos sendo apenas 3 deles pertencentes a seu sítio metálico de Fe. Nota-se que ao reportar um conjunto de resíduos como pertencentes ao sítio metálico, independente da predição estar correta ou não, o número de valores do tipo "Verdadeiro Negativo" (TN) será sempre elevado. Isso faz com que as métricas de avaliação de especificidade e acurácia (Seção 3.4)(que dão uma importância maior ao valor de TN) acabam por ter praticamente sempre resultados superiores a 90%. Ainda assim, mesmo com esse problema, o GASS-Metal consegue encontrar bons resultados, tendo acertado 100% dos resíduos desta proteína 5ZM4.

Esta característica desbalanceada dos dados deste experimento foi também responsável pela opção de não utilizar uma curva ROC para a demonstração dos resultados, apesar da mesma estar descrita na Seção 3.4. Curvas ROC não são aconselhadas em problemas com desbalanceamento de dados pois uma pequena variação no número de predições corretas ou incorretas nas classes com menos representantes podem causar grandes mudanças na curva e no valor AUC, levando assim a análises incorretas.

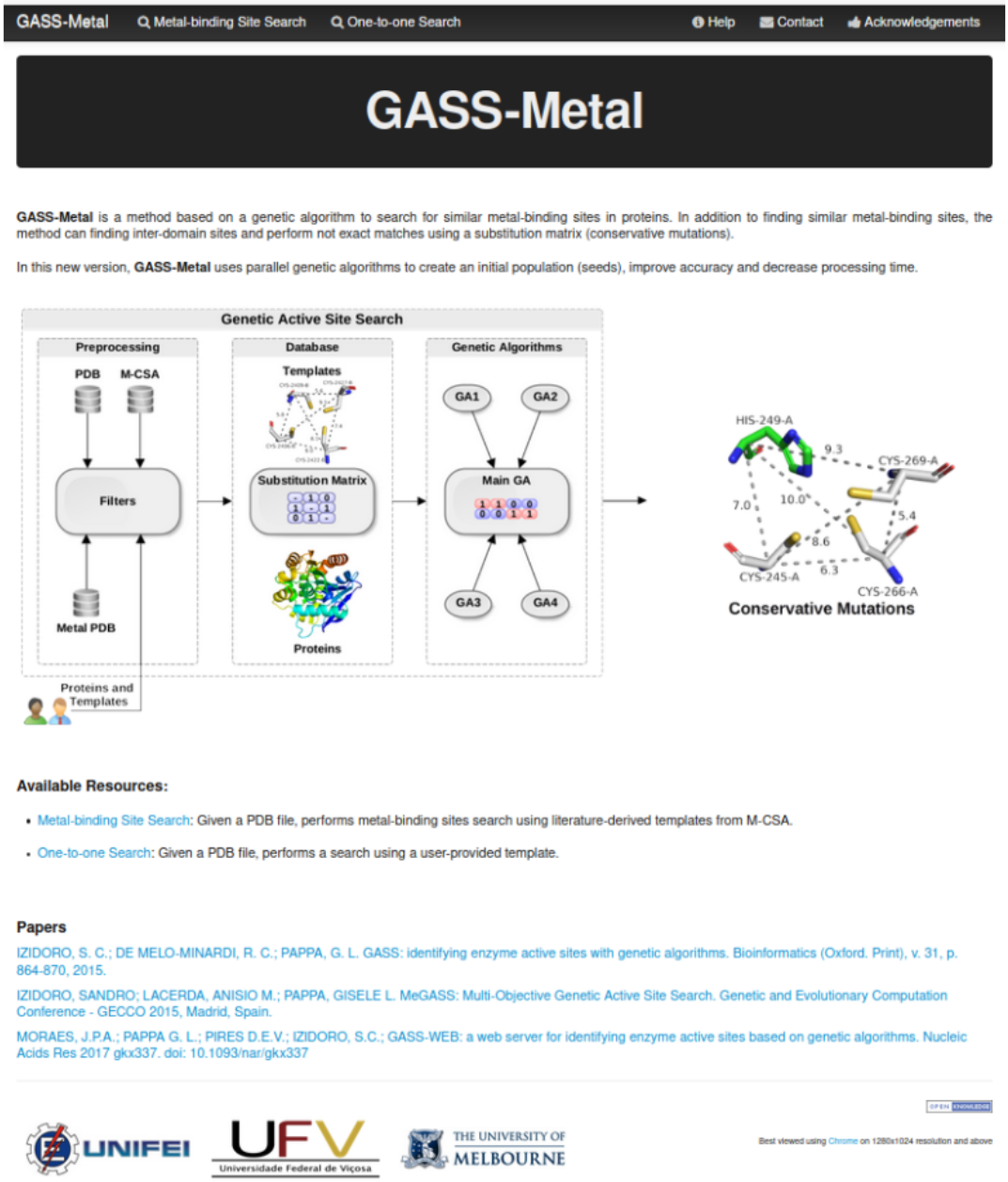
Vale mencionar novamente que pelo fato dos autores do trabalho de referência deste teste não disponibilizarem os sítios metálicos dos *targets* das proteínas, os testes não puderam ser reproduzidos fielmente. É importante destacar que, com *targets* e *templates* corretos, os resultados do GASS-Metal podem ser ainda melhores. Além disso, estes testes levaram em consideração apenas sítios dentro de uma mesma cadeia, devido principalmente à limitações de preditores em trabalhar apenas em um único domínio. O GASS-Metal é capaz de predizer sítios metálicos também em situações interdomínios.

Com os resultados aqui obtidos, foi possível observar que o método consegue trabalhar com ampla gama de íons metálicos e obtém bons resultados em praticamente todos eles.

4.4 Servidor web GASS-Metal

Com a intenção de disponibilizar o GASS-Metal para toda a comunidade científica, um servidor web foi implementado[1]. O servidor conta com duas maneiras distintas de realizar a identificação de sítios metálicos similares em proteínas: uma busca de sítios metálicos a partir de *templates* baseados no M-CSA (Seções 2.3.3 e 3.1.1) e MetalPDB (Seção 2.3.2) uma busca de sítios *um-para-um*. As Figuras 4.19 e 4.14 apresentam uma descrição do funcionamento geral do servidor GASS-Metal e sua página principal, respectivamente. As seções seguintes descrevem as duas formas de busca realizada e especificações técnicas de implementação do servidor.

Figura 4.14: Página inicial do servidor web GASS-Metal.



[1]Endereço do GASS-Metal: http://200.131.157.114

4.4.1 Busca de sítios similares utilizando *templates* LIT do M-CSA

A primeira forma de utilização do servidor trata-se da busca a partir de *templates* baseados no M-CSA. Esta busca consiste na forma padrão de utilização do GASS-Metal, onde tendo como base os diversos *templates* obtidos através do M-CSA, o servidor retorna uma lista de sítios ativos similares a esses templates, ordenados de acordo com seu valor *fitness*. A definição pelo uso de mutações conversativas (Seção 3.1.2) é de preferência do usuário.

A Figura 4.15 apresenta a busca utilizando *templates* baseados no M-CSA, ou busca tradicional do GASS-Metal (*Metal-binding Site Search*). Os passos apresentados nesse tipo de busca dizem respeito à proteína alvo onde será realizada a busca (Figura 4.15 - Step 1), bem como filtros utilizados para refinar a busca de acordo com os *templates* (Figura 4.15 - Step 2) e pelo uso ou não de mutações conservativas (Figura 4.15 - Step 3). A seguir, descreve-se cada um dos passos:

- Passo 1: O usuário deverá fornecer o nome da proteína, ou fornecer um arquivo PDB, que servirá como alvo da busca de sítios metálicos. Será nesta proteína que o GASS-Metal buscará por sítios similares utilizando seus *templates*.

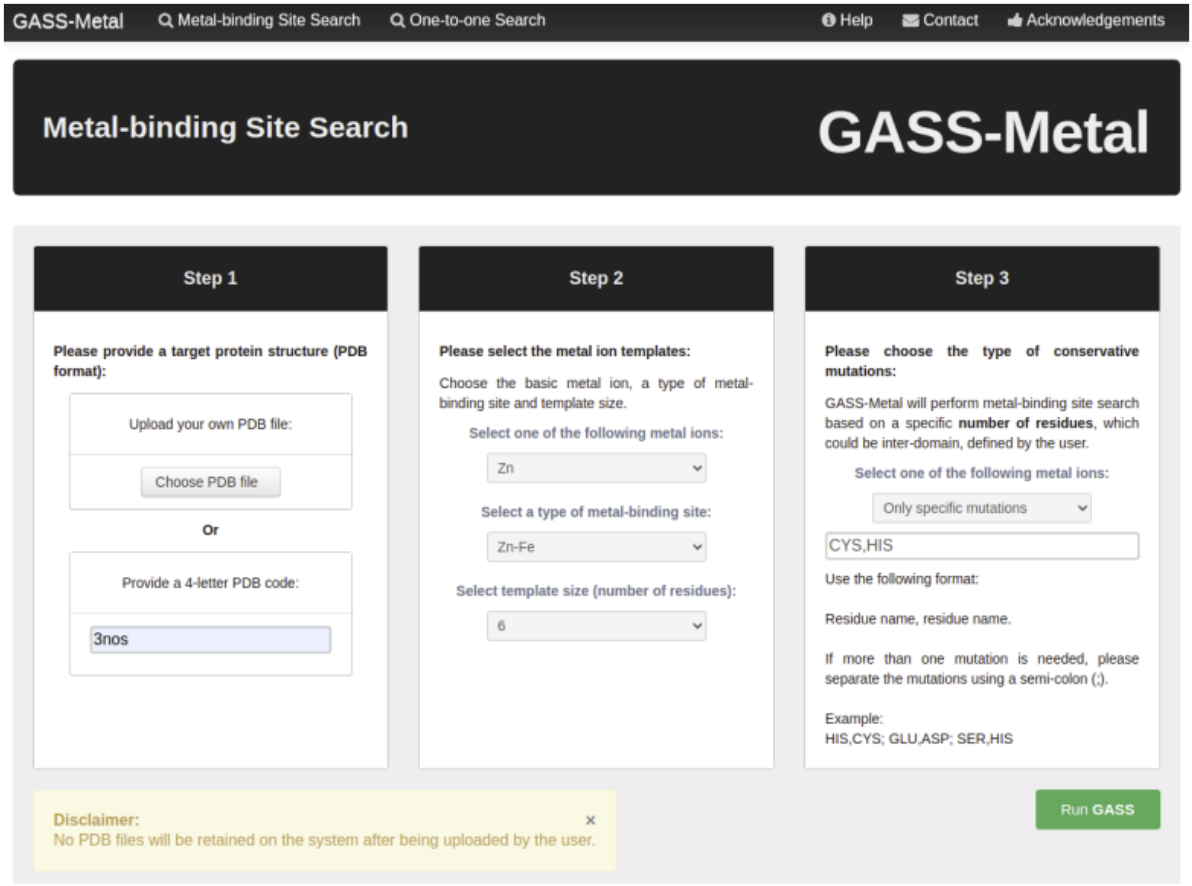
- Passo 2: Onde são filtrados os *templates* em que a busca do GASS-Metal vai se basear. O usuário deverá escolher, primeiramente, o íon metálico do sítio, o tipo deste sítio (se é simples, contendo apenas um íon, ou composto, tendo dois ou mais íons metálicos) e o número de resíduos que compõem os templates. O usuário poderá selecionar a opção *All*, fazendo com que o GASS-Metal procure em todos os *templates* daquele filtro específico.

- Passo 3: Por último, tem-se o passo referente às mutações conservativas. O usuário pode escolher uma dentre três opções: utilizar substituições providas pelo GASS-Metal, utilizar suas próprias substituições ou não utilizar substituição alguma.

Através do botão *Run GASS*, o GASS-Metal é executado, buscando na proteína alvo os sítios similares com base nos *templates* de acordo com os filtros aplicados na página e levando em consideração se as mutações conservativas foram aplicadas.

Os resultados são apresentados em forma de uma tabela (Figura 4.16), onde os sítios metálicos similares encontrados são ordenados pelo valor de *fitness*. O servidor informa o valor de *fitness* e os resíduos dos sítios metálicos encontrados na proteína alvo. Além disso, exibe também o nome da proteína e o *template* similar a cada sítio encontrado, bem como informações sobre a função, EC Number, Uniprot e resolução da estrutura. O usuário poderá escolher a quantidade de sítios similares visualizar, além de fazer o download dos resultados.

Figura 4.15: Página Metal-binding Site Search do servidor web GASS-Metal.



Para exibir a estrutura da proteína e o sítio similar encontrado, é necessário clicar no ícone de *um olho* ao lado dos resíduos encontrados, ou ao lado do PDB_ID do template. O GASS-Metal utiliza o LiteMol Viewer[2] para exibição das moléculas em 3D. A Figura 4.17 apresenta um resultado do GASS-Metal sendo exibido pelo LiteMol. Os resíduos do sítio similar encontrado aparecem em vermelho na proteína. Na parte inferior da janela do LiteMol é possível selecionar os outros sítios que o GASS-Metal encontrou.

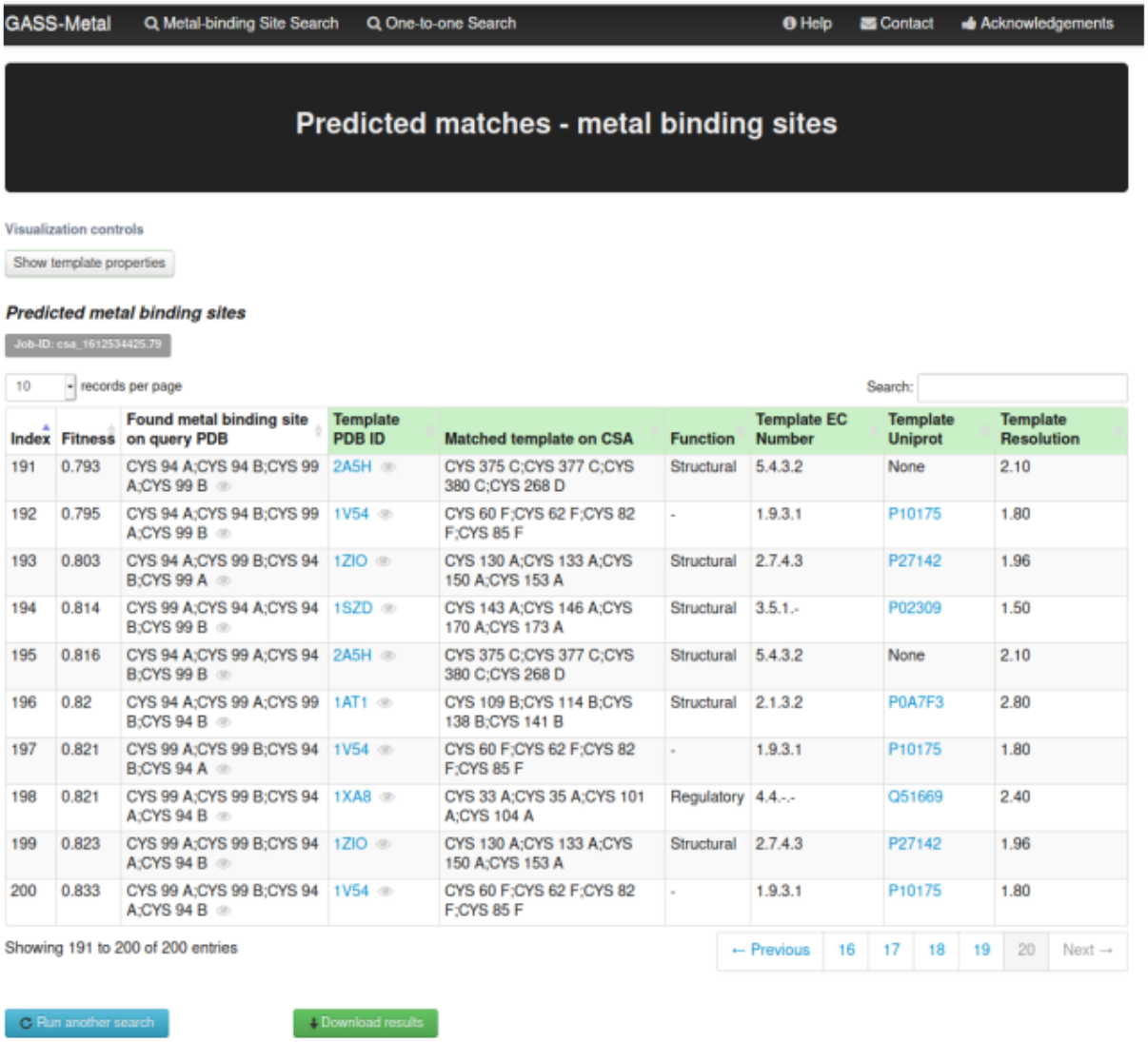
4.4.2 Busca de sítios similares *um-para-um*

A busca *um-para-um* é um método de busca que utiliza o método GASS-Metal para realizar a busca por sítios metálicos similares em uma proteína alvo baseando-se em um *template* de uma proteína referência. Neste tipo de busca, nenhuma mutação conservativa ou *template* do GASS-Metal é utilizado, todas as informações são fornecidas unicamente pelo usuário do servidor.

A Figura 4.18 apresenta a opção de busca *um-para-um* (*One-to-one Site Search*). Neste caso, o usuário deve seguir os quatro passos descritos na página para obter os resultados de busca. Os três primeiros passos se referem a informações sobre uma proteína de referência a qual a busca por sítios metálicos vai se basear. O último passo se refere à proteína alvo,

[2]https://www.litemol.org

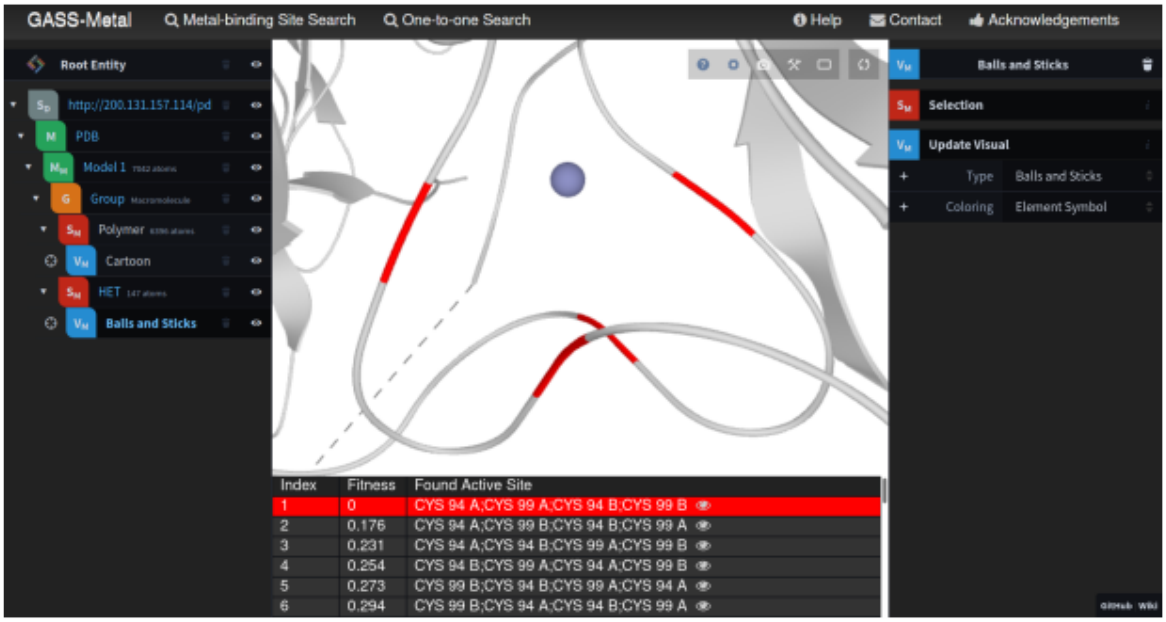
Figura 4.16: Página de Resultados - Metal-binding Site Search.



onde será feita a busca. A seguir uma descrição mais detalhada de cada um dos quatro passos da busca *um-para-um*:

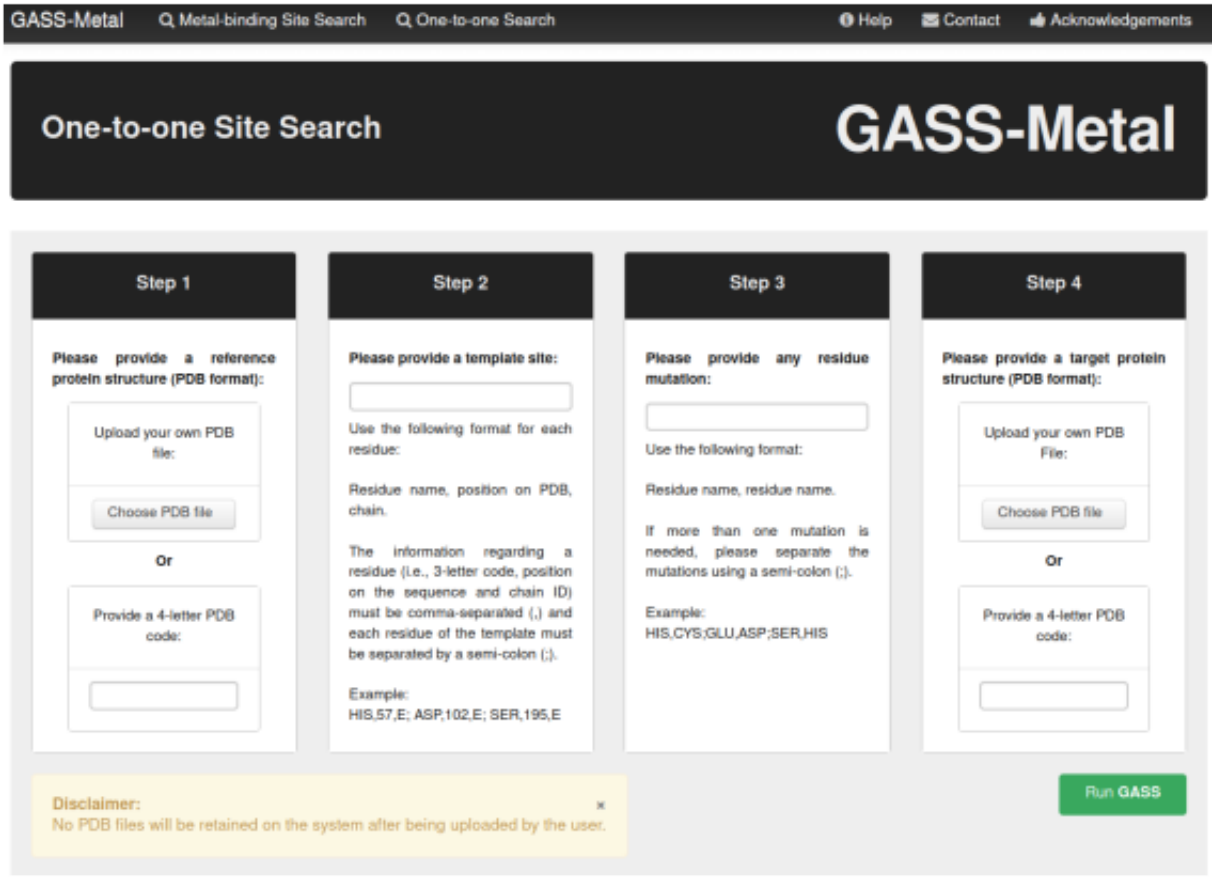
- Passo 1: O usuário deve indicar o nome da proteína, ou fornecer um arquivo PDB, que servirá como referência para a busca de sítios metálicos pelo GASS-Metal.
- Passo 2: Tendo sido definida a proteína, deve-se indicar qual o *template* de referência (que deve estar contido na proteína de referência) na qual a busca irá se basear.
- Passo 3: Neste passo o usuário pode indicar que não deseja usar substituições de resíduos na busca ou informar que deseja mutações. Caso opte pelo uso de mutações conservativas, o usuário deverá fornecer quais resíduos podem ser substituídos na proteína alvo.
- Passo 4: No último passo, o usuário deverá fornecer o nome da proteína, ou fornecer um arquivo PDB, que servirá como alvo da busca de sítios metálicos. Será nesta

Figura 4.17: LiteMol - Metal-binding Site Search.



proteína que o GASS-Metal buscará por sítios similares ao *template* informado no passo 2.

Figura 4.18: LiteMol - One-to-one Search.



Tendo sido seguidos os quatro passos anteriores, o GASS-Metal é executado, buscando sítios similares de uma proteína referência em uma única proteína alvo (daí o nome *um-para-um*). Ao final da execução, os resultados são apresentados da mesma maneira que a busca pelos *templates* baseados no M-CSA.

4.4.3 Especificações técnicas do servidor web

O servidor GASS-Metal utiliza o *framework* de desenvolvimento Flask, que usa a linguagem Python para o desenvolvimento de aplicações web. O Flask visa manter o núcleo de desenvolvimento de maneira simples, dando flexibilidade ao desenvolvedor em diversas decisões. Ele depende de duas bibliotecas externas: o motor de *template* Jinja2 e o conjunto de ferramentas WSGI Werkzeug. Suas vantagens incluem simplicidade no desenvolvimento, rapidez, soluções para projetos pequenos e aplicações mais robustas.

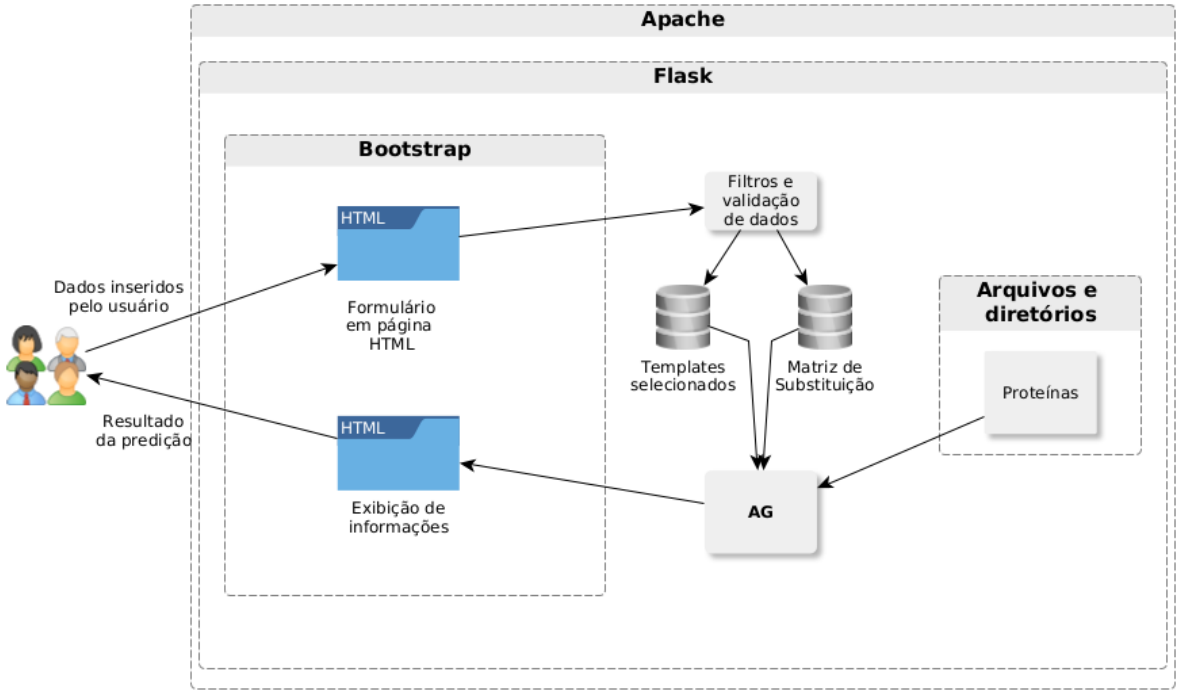
Para o design do *front-end* do servidor é utilizado o Bootstrap, um *framework* web com código-fonte aberto para desenvolvimento de componentes de interface e front-end para sites e aplicações web. O Bootstrap utiliza aspectos de HTML, CSS e JavaScript e se baseia em modelos de design para a tipografia, com foco na melhora da experiência do usuário.

O *back-end* do GASS-Metal é excetuado em um servidor Apache, responsável por disponibilizar páginas e recursos para a navegação web. Suas funcionalidades são mantidas através de uma estrutura de módulos, o que permite que desenvolvedores os implementem por meio da API do software. O servidor Apache é capaz de executar código em PHP, Perl, Shell Script e ASP e pode atuar como servidor FTP, HTTP, entre outros.

O GASS-Metal não trabalha com nenhum banco de dados. Em vez disso utiliza o sistema de arquivos e diretórios para o armazenamento de *templates* e outras informações pertinentes. A escolha pelo não uso de um banco de dados propriamente dito se deve pela não necessidade de tal abordagem. O GASS-Metal é uma aplicação simples, porém robusta, que consegue um desempenho melhor utilizando o sistema de arquivos e diretórios.

Para as requisições das buscas realizadas pelo GASS-Metal (Seções 4.4.1 e 4.4.2) o servidor trabalha praticamente da mesma forma: há primeiramente a obtenção dos dados informados pelo usuário através dos formulários definidos nas páginas HTML, fazendo o tratamento e verificação da integridade desses dados. Depois disso o servidor filtra os *templates* (em relação ao tipo do sítio metálico e seu tamanho) que serão utilizados na busca e define quais mutações conservativas irá utilizar naquela instância de busca. Por fim, o algoritmo GASS é executado com os *templates* definidos, gerando uma página de resultados depois do término da busca. Uma representação do funcionamento do servidor GASS-Metal pode ser vista na Figura 4.19.

Figura 4.19: Esquema do funcionamento do servidor web GASS-Metal



Capítulo 5

Conclusões e Trabalhos Futuros

Foi apresentado neste trabalho o GASS-Metal, um servidor web para identificação de sítios metálicos similares em proteínas baseado em algoritmos genéticos paralelos. O método é baseado no algoritmo GASS, que, inicialmente, proposto para a predição de sítios ativos, foi agora adaptado para o contexto de sítios metálicos em metaloproteínas. O GASS-Metal se destaca de outros métodos de predição ao trabalhar com vários tipos de íons metálicos diferentes e em buscar sítios metálicos interdomínios, além de obter resultados consistentes em diversos testes.

Diferentemente do que havia sido proposto por (Izidoro et al., 2014), o GASS-Metal trabalha com AGs paralelos. Isso se deve ao fato de que, apesar de obter bons resultados (superior a 90%) na predição de sítios catalíticos, o mesmo não acontecia com sítios metálicos. Observou-se, através do *teste de sanidade* (Seção 4.1) que apenas 74.45% dos *templates* de sítios metálicos eram encontrados, necessitando assim de uma nova abordagem para o problema. Com uma nova definição da população inicial do AG, agora com os resíduos da proteína sendo separados por quadrantes em relação ao centroide da estrutura, o GASS-Metal obteve um acerto médio de 92.86% no *teste de sanidade*.

Um outro teste feito para verificar a robustez do GASS-Metal foi com proteínas homólogas (Seção 4.2). Neste teste foram avaliados *templates* de 8 íons metálicos diferentes com o objetivo de analisar se o GASS-Metal seria capaz de encontrar sítios metálicos de proteínas homólogas tendo como base *templates* de proteínas curadas da literatura proveniente do M-CSA. Este teste, além de demonstrar mais uma vez que o GASS-Metal consegue uma alta porcentagem de acertos na predição de sítios metálicos, serviu também para validar a importância da matriz de substituição. Com o uso de substituições (mutações conservativas) houve um aumento na taxa de acerto em todos os *templates* avaliados, com valores superiores a 90% em todos os casos.

Um último teste tratou de comparar o GASS-Metal com diversos outros métodos *estado da arte* através da replicação do teste realizado por Qiao e Xie (2019). Apesar de não ser possível reproduzir fielmente o experimento (por conta da inviabilidade na obtenção dos *targets*) foi possível gerar novos dados com base na métrica de definição de sítios

metálicos do BioLiP bem similares ao experimento original. Os resultados obtidos pelo GASS-Metal superaram outros métodos em diversos tipos de sítios metálicos diferentes, tendo uma performance superior a todos os métodos em 5 dos 12 íons avaliados e tendo performance equivalente aos melhores em outros 6. Isto levando em consideração indivíduos ranqueados entre os 100 primeiros do *ranking* de resultados do GASS-Metal. E mesmo com bons resultados obtidos até aqui, o uso dos *targets* corretos poderia levar a resultados ainda melhores da predição de sítios metálicos.

Por fim, com o objetivo de prover à comunidade científica a ferramenta de busca de sítios metálicos, foi criado um servidor web que hospeda o GASS-Metal. O servidor contempla duas formas de busca de sítios metálicos: uma busca baseada nos *templates* LIT provenientes do M-CSA e MetalPDB e uma busca do tipo *um-para-um*, que permite ao usuário entrar com uma proteína referência, um *template* e uma proteína alvo e utilizar o GASS-Metal para realizar a busca do *template* na proteína alvo.

5.1 Direções de trabalhos futuros

Após a análise dos resultados, percebeu-se que algumas melhorias poderiam aprimorar ainda mais a busca de sítios metálicos do GASS-Metal. Pode-se citar as melhorias pertinentes ao método: o uso de novos *templates* baseados na base de dados BioLiP, o aprimoramento da geração da população do AG, a utilização de *templates* não redundantes e o uso de uma função *fitness* multiobjetiva para a busca de sítios similares.

5.1.1 Templates baseados no BioLiP

O GASS-Metal utiliza proteínas curadas pelo M-CSA e encontra, através do MetalPDB, os sítios metálicos de tais proteínas para definir seus *templates* que são utilizados para realizar a busca de sítios metálicos. Ao realizar os testes com *templates* da base de dados BioLiP e obter bons resultados, viu-se a possibilidade de utilizá-los também na ferramenta principal. Sendo assim, uma melhoria pertinente ao GASS-Metal seria disponibilizar na ferramenta a opção de escolha de diferentes tipos de *templates*, sendo um deles com base no MetalPDB e outro no BioLiP.

5.1.2 Aprimoramento da geração da população do AG

A abordagem de separar os resíduos de uma proteína em quadrantes no espaço tridimensional fez com que a taxa de acerto do GASS-Metal pudesse ser melhorada consideravelmente. Contudo, ainda existem casos onde a separação por quadrante não resolveu todos os problemas. É o caso da proteínas 1CT9, que apesar de ter uma melhora na taxa de acertos no *teste de sanidade* de 1.67% para 47.5%, ainda está longe de ter bons valores, como é o caso de outros íons metálicos com taxas de acerto superiores a 90%.

Um ponto a se trabalhar nesse sentido seria melhorar a geração da população dos AGs. Com a definição de populações iniciais mais uniformes o GASS-Metal poderia ter uma variabilidade genética maior e assim obter melhores resultados. A geração de *clusters* para separar os resíduos de uma proteína na população do GASS-Metal poderia ajudar nesse sentido. Ao invés de utilizar um centroide como ponto de separação e definir quadrantes, poderia-se calcular a densidade de resíduos na proteína e a partir disso separar os os resíduos em grupos de acordo com sua densidade.

Dessa forma, o GASS-Metal não teria sempre 4 grupos de resíduos (um para cada quadrante), mas sim uma forma dinâmica na separação dos resíduos. Com isso, o paralelismo dos AGs também poderia variar de acordo com cada instância do problema. Em vez de deixar definido um valor estático de 4 AGs paralelos, uma *thread* para cada *cluster* poderia ser criada, e o GASS-Metal teria um número entre 1 e N de AGs, sendo N a quantidade de *clusters* definidos. Isso faria com que além de melhorar os resultados de predição poderia otimizar a busca em relação ao tempo de execução.

Uma forma híbrida de atacar esse problema também poderia ser levada em consideração. Embora a busca restrita apenas a uma cadeia faça com que sítios interdomínio não sejam encontrados, uma busca híbrida que utilizasse uma busca em cadeia, quadrantes e *clusters* poderia otimizar ainda mais o método. Dependendo da proteína, abordagens diferentes e/ou conjuntas poderiam ser utilizadas.

5.1.3 Utilização de *templates* não redundantes

Um problema visto durante os testes foi que *templates* muito similares influenciam consideravelmente na posição do *ranking* que um sítio metálico candidato está. Se existe um número grande de *templates* similares, dado um sítio metálico candidato encontrado, as distâncias (ou valor de *fitness*) entre o candidato e todos os *templates* similares serão muito próximas. Isso faz com que uma parte considerável dos resultados tenha praticamente a mesma informação.

Exemplos de redundância observados nos *templates* do GASS-Metal ocorrem nos sítios de Zn. Dos 193 *templates* deste íon metálico, 33 deles tem tamanho igual a 3 e são formados por 3 HIS. Outros 35 *templates* de Zn de tamanho 4 contém as mesmas 4 CYS. É possível observar que em relação aos resíduos e tamanhos esses *templates* são exatamente iguais entre si, o que faz com que o valor de *fitness* de sítios candidatos em relação a esses *templates* redundantes sejam muito próximos.

Uma maneira de trabalhar nesse problema seria a definição de *templates* que representassem os casos redundantes. O *template* representante poderia ser obtido ao se calcular as coordenadas médias dos resíduos entre todos os *templates* redundantes (criando assim um novo *template* que não necessariamente seja um dos que já exista no conjunto). E uma outra forma seria definindo um dos *templates* existentes como padrão, removendo todos os outros da base do GASS-Metal.

5.1.4 Função *fitness* multiobjetiva

(Izidoro et al., 2015) propôs em seu trabalho uma função *fitness* multiobjetiva na busca de sítios ativos em proteínas. Além de utilizar as distâncias entre resíduos, analisou-se também a profundidade em que eles se encontram na estrutura da proteína. Uma abordagem parecida poderia ser utilizada também no GASS-Metal.

A profundidade de resíduos é uma informação relevante para a predição de sítios catalíticos pois, para que possa interagir com outras moléculas, os resíduos geralmente estão em regiões mais próximas da superfície da proteína. Porém, sítios metálicos desempenham diversos papeis em uma proteína, podendo assumir funções de estabilização estrutural, transferência de elétrons, transporte, além de catálises. Isso faz com que os resíduos de sítios metálicos nem sempre estejam próximos da superfície, fazendo com que a profundidade talvez não traga bons resultados.

Contudo, a ideia de trazer outras informações para ajudar a função *fitness* é válida. Utilizar informações físico-químicas do ambiente do sítio metálico como um segundo critério de avaliação da *fitness* pode aprimorar os resultados do GASS-Metal.

Referências Bibliográficas

(1999). Nomenclature committee of the international union of biochemistry and molecular biology (NC-IUBMB), Enzyme Supplement 5 (1999). *European Journal of Biochemistry*, 264(2):610–650.

A. J. Umbarkar, P. D. S. (2015). Crossover operators in genetic algorithms:a review. *ICTACT Journal on Soft Computing*, 6(1):1083–1092.

Ajitha, M.; Sundar, K.; Arul Mugilan, S. e Arumugam, S. (2018). Development of metal-active site and zinccluster tool to predict active site pockets. *Proteins: Structure, Function, and Bioinformatics*, 86(3):322–331.

Akcapinar, G. B. e Sezerman, O. U. (2017). Computational approaches for de novo design and redesign of metal-binding sites on proteins. *Bioscience Reports*, 37(2). BSR20160179.

Altman, R. B. e Dugan, J. M. (2009). *Structural Bioinformatics*. Wiley-Blackwell, 2 edição.

Andreini, C.; Bertini, I. e Cavallaro, G. (2011). Minimal functional sites allow a classification of zinc sites in proteins. *PLOS ONE*, 6(10):1–13.

Andreini, C.; Bertini, I. e Rosato, A. (2009). Metalloproteomes: A bioinformatic approach. *Accounts of Chemical Research*, 42(10):1471–1479. PMID: 19697929.

Banaszak, L. J. (2000). Chapter 11 - metal ions bound to proteins. In BANASZAK, L. J., editor, *Foundations of Structural Biology*, pp. 137 – 147. Academic Press, San Diego.

Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissing, H.; Shindyalov, I. N. e Bourne, P. E. (2000). The protein data bank. *Nucleic Acids Res*, 28:235–242.

Boer, J. L. e Hausinger, R. P. (2013). *Nickel-Binding Sites in Proteins*, pp. 1528–1534. Springer New York, New York, NY.

Brylinski, M. e Skolnick, J. (2011). Findsite-metal: Integrating evolutionary information and machine learning for structure-based metal-binding site prediction at the proteome level. *Proteins: Structure, Function, and Bioinformatics*, 79(3):735–751.

Burley, S. K.; Bhikadiya, C.; Bi, C.; Bittrich, S.; Chen, L.; Crichlow, G. V.; Christie, C. H.; Dalenberg e Zhuravleva, M. (2020). Rcsb protein data bank: powerful new tools for exploring 3d structures of biological macromolecules for basic and applied research and education in fundamental biology, biomedicine, biotechnology, bioengineering and energy sciences. *Nucleic Acids Research*, 49(D1):D437–D451.

Butenhof, D. R. (1997). *Programming with POSIX threads*. Addison-Wesley Professional.

Cai, P.; Cai, Y.; Chandrasekaran, I. e Zheng, J. (2016). Parallel genetic algorithm based automatic path planning for crane lifting in complex environments. *Automation in Construction*, 62:133 – 147.

Camargo, G. d. M. (2006). Controle da pressão seletiva em algoritmo genético aplicado a otimização de demanda em infra-estrutura aeronáutica. Master's thesis, Escola Politécnica, Universidade de São Paulo.

Cantu-Paz, E. (1998). A survey of parallel genetic algorithms. *CALCULATEURS PARALLELES*, 10.

Cao, X.; Hu, X.; Zhang, X.; Gao, S.; Ding, C.; Feng, Y. e Bao, W. (2017). Identification of metal ion binding sites based on amino acid sequences. *PLOS ONE*, 12(8):1–16.

Cervantes, J. e Stephens, C. R. (2006). "optimal"mutation rates for genetic search. GECCO 2006, pp. 1313–1320, New York, NY, USA. Association for Computing Machinery.

Cetin, U. e Gundogmus, Y. E. (2019). Feature selection with evolving, fast and slow using two parallel genetic algorithms. In *2019 4th International Conference on Computer Science and Engineering (UBMK)*, pp. 699–703.

Cooper, G. e Hausman, R. (2004). *The Cell: A Molecular Approach*. ASM Press.

Denesyuk, A. I.; Permyakov, S. E.; Johnson, M. S.; Denessiouk, K. e Permyakov, E. A. (2020). System approach for building of calcium-binding sites in proteins. *Biomolecules*, 10(4).

Diaz-Gomez, P. A. e Hougen, D. F. (2007). Initial population for genetic algorithms: A metric approach. In Arabnia, H. R.; Yang, J. Y. e Yang, M. Q., editores, *Proceedings of the 2007 International Conference on Genetic and Evolutionary Methods, GEM 2007, June 25-28, 2007, Las Vegas, Nevada, USA*, pp. 43–49. CSREA Press.

Doerr, B.; Doerr, C. e Ebel, F. (2015). From black-box complexity to designing new genetic algorithms. *Theoretical Computer Science*, 567:87–104.

Dréo, J.; Chatterjee, A.; Pétrowski, A.; Siarry, P. e Taillard, E. (2006). *Metaheuristics for Hard Optimization: Methods and Case Studies*. Springer Berlin Heidelberg.

Dudev, M. e Lim, C. (2007). Discovering structural motifs using a structural alphabet: Application to magnesium-binding sites. *BMC Bioinformatics*, 8(1):106.

Durdagi, S.; Roux, B. e Noskov, S. Y. (2013). *Potassium-Binding Site Types in Proteins*, pp. 1809–1815. Springer New York, New York, NY.

Eiben, A. e Smith, J. (2007). *Introduction to Evolutionary Computing*. Natural Computing Series. Springer Berlin Heidelberg.

El-Gebali, S.; Mistry, J.; Bateman, A.; Eddy, S. R.; Luciani, A.; Potter, S. C.; Qureshi, M.; Richardson, L. J.; Salazar, G. A.; Smart, A. et al. (2019). The pfam protein families database in 2019. *Nucleic acids research*, 47(D1):D427–D432.

Fauzi Mohd Johar; Farah Ayuni Azmin; Mohamad Kadim Suaidi; Shibghatullah, A. S.; Badrul Hisham Ahmad; Siti Nadzirah Salleh; Mohamad Zoinol Abidin Abd Aziz e Shukor, M. M. (2013). A review of genetic algorithms and parallel genetic algorithms on graphics processing unit (gpu). In *2013 IEEE International Conference on Control System, Computing and Engineering*, pp. 264–269.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874.

Finkelstein, J. (2009). Metalloproteins. *Nature*, 460(7257):813–813.

Furnham, N.; Holliday, G. L.; de Beer, T. A. P.; Jacobsen, J. O. B.; Pearson, W. R. e Thornton, J. M. (2013). The Catalytic Site Atlas 2.0: cataloging catalytic sites and residues identified in enzymes. *Nucleic Acids Research*, pp. 1–5.

Gabriel, E.; Fagg, G. E.; Bosilca, G.; Angskun, T.; Dongarra, J. J.; Squyres, J. M.; Sahay, V.; Kambadur, P.; Barrett, B.; Lumsdaine, A.; Castain, R. H.; Daniel, D. J.; Graham, R. L. e Woodall, T. S. (2004). Open mpi: Goals, concept, and design of a next generation mpi implementation. In Kranzlmüller, D.; Kacsuk, P. e Dongarra, J., editores, *Recent Advances in Parallel Virtual Machine and Message Passing Interface*, pp. 97–104, Berlin, Heidelberg. Springer Berlin Heidelberg.

Goldberg, D.; David Edward, G.; Goldberg, D. e Goldberg, V. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Artificial Intelligence. Addison-Wesley Publishing Company.

Griep, S. e Hobohm, U. (2009). Pdbselect 1992-2009 and pdbfilter-select. *Nucleic Acids Research*, 38(suppl1):D318–D319.

Hand, D. (2009). Measuring classifier performance: a coherent alternative to the area under the ROC curve. *Machine Learning*, 77(1):103–123.

Harding, M. M.; Nowicki, M. W. e Walkinshaw, M. D. (2010). Metals in protein structures: a review of their principal features. *Crystallography Reviews*, 16(4):247–302.

Haynes, W. M. (2014). *CRC handbook of chemistry and physics*. CRC press.

He, W.; Liang, Z.; Teng, M. e Niu, L. (2015). mFASD: a structure-based algorithm for discriminating different types of metal-binding sites. *Bioinformatics*, 31(12):1938–1944.

Henikoff, S. e Henikoff, J. G. (1992). Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919.

Hu, X.; Dong, Q.; Yang, J. e Zhang, Y. (2016). Recognizing metal and acid radical ion-binding sites by integrating ab initio modeling with template-based transferals. *Bioinformatics*, 32(21):3260–3269.

ISO (2012). *ISO/IEC 14882:2011 Information technology — Programming languages — C++*. International Organization for Standardization, Geneva, Switzerland.

Izidoro, S.; Lacerda, A. M. e Pappa, G. L. (2015). Megass: Multi-objective genetic active site search. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation*, GECCO Companion '15, pp. 905–910, New York, NY, USA. Association for Computing Machinery.

Izidoro, S. C. (2005). Determinação do número de agrupamentos em conjuntos de dados multidimensionais utilizando algoritmos genéticos. *INFOCOMP Journal of Computer Science*, 4(4):67–72.

Izidoro, S. C.; de Melo-Minardi, R. C. e Pappa, G. L. (2014). GASS: identifying enzyme active sites with genetic algorithms. *Bioinformatics*, 31(6):864–870.

Jonson, P. H. e Petersen, S. B. (2001). A critical view on conservative mutations. *Protein Engineering, Design and Selection*, 14(6):397–402.

Katoch, S.; Chauhan, S. S. e Kumar, V. (2020). A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*.

Keskin, O.; Gursoy, A.; Ma, B. e Nussinov, R. (2008). Principles of protein- protein interactions: what are the preferred ways for proteins to interact? *Chemical reviews*, 108(4):1225–1244.

Khrustalev, V. V.; Barkovsky, E. V. e Khrustaleva, T. A. (2016). Magnesium and manganese binding sites on proteins have the same predominant motif of secondary structure. *Journal of Theoretical Biology*, 395:174 – 185.

Kozlowski, L. P. (2016). Proteome-pI: proteome isoelectric point database. *Nucleic Acids Research*, 45(D1):D1112–D1116.

Kurdi, M. (2016). An effective new island model genetic algorithm for job shop scheduling problem. *Computers & Operations Research*, 67:132 – 142.

Laskowski, R. A.; Watson, J. D. e Thornton, J. M. (2005). Protein function prediction using local 3d templates. *Journal of molecular biology*, 351(3):614–626.

Lev, B.; Roux, B. e Noskov, S. Y. (2013). *Sodium-Binding Site Types in Proteins*, pp. 2112–2118. Springer New York, New York, NY.

Levy, R.; Edelman, M. e Sobolev, V. (2009). Prediction of 3d metal binding sites from translated gene sequences based on remote-homology templates. *Proteins: Structure, Function, and Bioinformatics*, 76(2):365–374.

Lin, Y.-F.; Cheng, C.-W.; Shih, C.-S.; Hwang, J.-K.; Yu, C.-S. e Lu, C.-H. (2016). Mib: Metal ion-binding site prediction and docking server. *Journal of Chemical Information and Modeling*, 56(12):2287–2291. PMID: 27976886.

Lippi, M.; Passerini, A.; Punta, M. e Frasconi, P. (2012). Metal binding in proteins: Machine learning complements x-ray absorption spectroscopy. In Flach, P. A.; De Bie, T. e Cristianini, N., editores, *Machine Learning and Knowledge Discovery in Databases*, pp. 854–857, Berlin, Heidelberg. Springer Berlin Heidelberg.

Liu, Y. e Tao, L. (2008). Protein structure prediction based on an improved genetic algorithm. In *2008 2nd International Conference on Bioinformatics and Biomedical Engineering*, pp. 577–580.

Lu, C.-H.; Lin, Y.-F.; Lin, J.-J. e Yu, C.-S. (2012). Prediction of metal ion–binding sites in proteins using the fragment transformation method. *PLOS ONE*, 7(6):1–12.

Luscombe, N. M.; Greenbaum, D. e Gerstein, M. (2001). What is bioinformatics? a proposed definition and overview of the field. *Methods of information in medicine*, 40(04):346–358.

Madhuri e Deep, K. (2009). A state-of-the-art review of population-based parallel meta-heuristics. In *2009 World Congress on Nature Biologically Inspired Computing (NaBIC)*, pp. 1604–1607.

Maiorov, V. N. e Crippen, G. M. (1994). Significance of root-mean-square deviation in comparing three-dimensional structures of globular proteins. *Journal of Molecular Biology*, 235(2):625 – 634.

Majd, A.; Lotfi, S. e Sahebi, G. (2013). Review on parallel evolutionary computing and introduce three general framework to parallelize all ec algorithms. *The 5th Conference on Information and Knowledge Technology*, pp. 61–66.

McPhalen, C. A.; Strynadka, N. C. e James, M. N. (1991). Calcium-binding sites in proteins: A structural perspective. In Anfinsen, C.; Edsall, J. T.; Richards, F. M. e Eisenberg, D. S., editores, *Metalloproteins: Structural Aspects*, volume 42 of *Advances in Protein Chemistry*, pp. 77 – 144. Academic Press.

Meadows, B.; Riddle, P.; Skinner, C. e Barley, M. M. (2013). Evaluating the seeding genetic algorithm. In Cranefield, S. e Nayak, A., editores, *AI 2013: Advances in Artificial Intelligence*, pp. 221–227, Cham. Springer International Publishing.

Medhavi Mallick, A. S. V. e Shankaracharya (2011). Tools for predicting metal binding sites in protein: A review. *Current Bioinformatics*, 6(4):444–449.

Mika, S. e Rost, B. (2003). Uniqueprot: Creating representative protein sequence sets. *Nucleic acids research*, 31(13):3789–3791.

Mitchell, M. (1998). *An Introduction to Genetic Algorithms*. A Bradford book. Bradford Books.

Moraes, J. P. A.; Pappa, G. L.; Pires, D. E. V. e Izidoro, S. C. (2017). GASS-WEB: a web server for identifying enzyme active sites based on genetic algorithms. *Nucleic Acids Research*, 45(W1):W315–W319.

Morris, G. M.; Goodsell, D. S.; Halliday, R. S.; Huey, R.; Hart, W. E.; Belew, R. K. e Olson, A. J. (1998). Automated docking using a lamarckian genetic algorithm and an empirical binding free energy function. *Journal of Computational Chemistry*, 19(14):1639–1662.

Nilmeier, J. P.; Kirshner, D. A.; Wong, S. E. e Lightstone, F. C. (2013). Rapid catalytic template searching as an enzyme function prediction procedure. *PLOS ONE*, 8(5):1–17.

Nowostawski, M. e Poli, R. (1999). Parallel genetic algorithm taxonomy. In *1999 Third International Conference on Knowledge-Based Intelligent Information Engineering Systems. Proceedings (Cat. No.99TH8410)*, pp. 88–92.

Otovic, E.; Njirjak, M.; Zuzic, I.; Kalafatovic, D. e Mausa, G. (2020). Genetic algorithm parametrization for informed exploration of short peptides chemical space. In *2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pp. 1–3.

Passerini, A.; Lippi, M. e Frasconi, P. (2011). MetalDetector v2.0: predicting the geometry of metal binding sites from protein sequence. *Nucleic Acids Research*, 39(suppl2):W288–W292.

Passerini, A.; Lippi, M. e Frasconi, P. (2012). Predicting metal-binding sites from protein sequence. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 9(1):203–213.

Putignano, V.; Rosato, A.; Banci, L. e Andreini, C. (2017). Metalpdb in 2018: a database of metal sites in biological macromolecular structures. *Nucleic Acids Research*, 46(D1):D459–D464.

Qiao, L. e Xie, D. (2019). Mionsite: Ligand-specific prediction of metal ion-binding sites via enhanced adaboost algorithm with protein sequence information. *Analytical Biochemistry*, 566:75 – 88.

Ribeiro, A. J. M.; Holliday, G. L.; Furnham, N.; Tyzack, J. D.; Ferris, K. e Thornton, J. M. (2017). Mechanism and catalytic site atlas (m-csa): a database of enzyme reaction mechanisms and active sites. *Nucleic Acids Research*, 46(D1):D618–D623.

Roberts, R. J. (2004). Identifying Protein Function - A Call for Community Action. *PLoS Biology*, 2:293–294.

Santana, C. A.; Silveira, S. d. A.; Moraes, J. P.; Izidoro, S. C.; de Melo-Minardi, R. C.; Ribeiro, A. J.; Tyzack, J. D.; Borkakoti, N. e Thornton, J. M. (2020). Grasp: a graph-based residue neighborhood strategy to predict binding sites. *Bioinformatics*, 36(Supplement_2):i726–i734.

Schapire, R. E. (1999). A brief introduction to boosting. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'99, pp. 1401–1406, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Sciortino, G.; Garribba, E.; Rodríguez-Guerra Pedregal, J. e Maréchal, J.-D. (2019). Simple coordination geometry descriptors allow to accurately predict metal-binding sites in proteins. *ACS Omega*, 4(2):3726–3731.

Shukla, D. e Singh, U. (2021). A generic framework for evolution of deep neural networks using genetic algorithms. In Gupta, D.; Khanna, A.; Bhattacharyya, S.; Hassanien, A. E.; Anand, S. e Jaiswal, A., editores, *International Conference on Innovative Computing and Communications*, pp. 117–127, Singapore. Springer Singapore.

Sobolev, V. e Edelman, M. (2013). Web tools for predicting metal binding sites in proteins. *Israel Journal of Chemistry*, 53(3-4):166–172.

Song, J.; Li, C.; Zheng, C.; Revote, J.; Zhang*, Z. e Webb*, G. I. (2017). Metalexplorer, a bioinformatics tool for the improved prediction of eight types of metal-binding sites using a random forest algorithm with two- step feature selection. *Current Bioinformatics*, 12(6):480–489.

Soni, N. e Kumar, T. (2014). Study of various mutation operators in genetic algorithms. volume 5, pp. 4519–4521.

Tainer, J. A.; Roberts, V. A. e Getzoff, E. D. (1992). Protein metal-binding sites. *Current Opinion in Biotechnology*, 3(4):378 – 387.

Umbarkar, A. J. e Joshi, M. S. (2013). Review of parallel genetic algorithm based on computing paradigm and diversity in search space. *ICTACT Journal on Soft Computing*, 3(4):615–622.

Yamada, K. e Tomii, K. (2013). Revisiting amino acid substitution matrices for identifying distantly related proteins. *Bioinformatics*, 30(3):317–325.

Yamashita, M. M.; Wesson, L.; Eisenman, G. e Eisenberg, D. (1990). Where metal ions bind in proteins. *Proceedings of the National Academy of Sciences*, 87(15):5648–5652.

Yang, J.; Roy, A. e Zhang, Y. (2012). BioLiP: a semi-manually curated database for biologically relevant ligand-protein interactions. *Nucleic Acids Research*, 41(D1):D1096–D1103.

Yang, J.; Roy, A. e Zhang, Y. (2013). Protein–ligand binding site recognition using complementary binding-specific substructure comparison and sequence profile alignment. *Bioinformatics*, 29(20):2588–2595.

Yu, D.-J.; Hu, J.; Yang, J.; Shen, H.-B.; Tang, J. e Yang, J.-Y. (2013). Designing template-free predictor for targeting protein-ligand binding sites with classifier ensemble and spatial clustering. *IEEE/ACM transactions on computational biology and bioinformatics*, 10(4):994–1008.

Zhang, Y. e Zheng, J. (2020). Bioinformatics of metalloproteins and metalloproteomes. *Molecules (Basel, Switzerland)*, 25(15):3366.

Zhao, W.; Xu, M.; Liang, Z.; Ding, B.; Niu, L.; Liu, H. e Teng, M. (2011). Structure-based de novo prediction of zinc-binding sites in proteins of unknown function. *Bioinformatics*, 27(9):1262–1268.

Zheng, H.; Chruszcz, M.; Lasota, P.; Lebioda, L. e Minor, W. (2008). Data mining of metal ion environments present in protein structures. *Journal of Inorganic Biochemistry*, 102(9):1765 – 1776.

Zheng, H.; Cooper, D. R.; Porebski, P. J.; Shabalin, I. G.; Handing, K. B. e Minor, W. (2017). Checkmymetal: a macromolecular metal-binding validation tool. *Acta Crystallographica Section D*, 73(3):223–233.

Zvelebil, M. e Baum, J. O. (2008). *Understanding Bioinformatics*. Garland Science.