Uma Abordagem para Análise de Padrões em Banco de Dados de Doadores de Órgãos

Marcio N. P. Silva[1], **Karla Figueiredo**[1], **Luís C. M. S. Porto**[2], **Alexandre C. Sena**[1]

[1]Instituto de Matemática e Estatística – Universidade do Estado do Rio de Janeiro
Rio de Janeiro – RJ – Brasil

[2]Laboratório de Histocompatibilidade e Criopreservação
Universidade do Estado do Rio de Janeiro – RJ – Brasil

Abstract. *The search for a compatible donor is carried out through a search algorithm, whose objective is to find potential donors in a volunteer register. In this context, based on the implementation of an algorithm to search for compatible donors, the objective of this article is to propose and implement a tool to analyze patterns in organ donor databases, which will help physicians and researchers obtain information to improve the process of organ donation among unrelated donors. Initial results showed that the tool is capable of performing the proposed analyses, as well as significantly reducing the time required according to the available computational architecture.*

Resumo. *A procura por um doador compatível é realizada através de um algoritmo de busca, cujo objetivo é buscar em um cadastro de voluntários os potenciais doadores. Nesse contexto, a partir da implementação de um algoritmo de busca por doadores compatíveis, o objetivo deste artigo é propor e implementar uma ferramenta para analisar padrões em banco de dados de doadores de órgãos, que ajude a médicos e pesquisadores obterem informações para aprimorar o processo de doação de órgãos entre doadores não relacionados. Resultados iniciais mostraram que a ferramenta é capaz de realizar as análises propostas, assim como reduzir significativamente o tempo necessário de acordo com a arquitetura computacional disponível.*

1. Introdução

O transplante de células-tronco hematopoéticas pode ser a única forma de tratamento para pacientes com certos tipos de câncer (por exemplo, linfoma), *bone marrow failure disorders* ou neoplasias hematológicas [Singh and McGuirk 2016]. Dentre os muitos fatores para se conseguir um transplante bem sucedido, a compatibilidade *Human Leukocyte Antigens* (HLA) entre paciente e doador é a mais importante [Eapen et al. 2014, Tiercy 2016, Dehn et al. 2019].

Enquanto alguns pacientes conseguem encontrar doadores compatíveis dentro da própria família, muitos pacientes tem que recorrer a doadores voluntários (doadores não relacionados) cadastrados em registros nacionais ou internacionais. De acordo com estudos baseados nos registros do programa nacional de doadores de medula óssea dos Estados Unidos (NMDP), 70% dos pacientes não tem um parente HLA-compatível, ou seja, dependem dos doadores não relacionados [Gragert et al. 2014]. Nestes casos, a rápida e correta identificação de possíveis doadores é uma tarefa crucial.

Esta busca por um doador compatível é realizada através de um algoritmo de compatibilidade HLA (*HLA matching algorithm* – HMA) [Bochtler et al. 2016], cujo objetivo é buscar em um cadastro de voluntários os potenciais doadores [Steiner 2012]. A compatibilidade entre paciente-doador é definida pelo número de alelos compartilhados pelos genes HLA-A, -B, -C, -DRB1 e -DQB1 [Geffard et al. 2019]. A entrada principal do algoritmo é a tipagem HLA do paciente, onde no mínimo devem ser informados os genes HLA-A, -B e -DRB1, e a saída do algoritmo é uma lista com os prováveis candidatos a doadores.

Nesse contexto, a partir da implementação de um algoritmo de compatibilidade HLA, o objetivo deste artigo é propor e implementar uma ferramenta para apoiar médicos e pesquisadores, por meios de análise de padrões em banco de doadores de órgãos, e, com isso, ajudar a aprimorar o processo de doação de órgãos entre doadores não relacionados. Mais especificamente, a ferramenta tem o objetivo de comparar a tipagem HLA de doadores e pacientes e, a partir dessas consultas realizar comparações antropológicas/populacionais, inferir o tempo de espera para pacientes que necessitam de transplante de medula óssea, comparação da prevalência de determinados alelos, entre outras análises. Uma vez que essas análises podem ser demoradas, outro objetivo deste trabalho é paralelizar a busca, e consequentemente, reduzir o tempo de resposta.

Avaliações preliminares mostram a viabilidade da abordagem proposta. A ferramenta é capaz de realizar as buscas e gerar as análises para os pesquisadores. Além disso, o tempo de busca pode ser significativamente reduzido de acordo com a máquina disponível em função do escalonador de busca estar paralelizado.

O restante deste trabalho está organizado da seguinte forma: na Seção 2 são apresentados os trabalhos relacionados; a Seção 3 descreve os algoritmos de busca; em seguida, na Seção 4, é apresentada a abordagem para a análise de registros de doadores de órgãos; a implementação inicial da ferramenta é descrita na Seção 5 seguida da sua avaliação experimental na Seção 6; por fim, as conclusões são apresentados na Seção 7.

2. Trabalhos Relacionados

Esta seção apresenta o que de mais importante foi encontrado na literatura sobre algoritmos de busca HLA. O artigo em [Bochtler et al. 2016] compara os principais aspectos do comportamento dos algoritmos de busca, especialmente quando eles têm que lidar com a grande variedade de dados de genótipos HLA nos registros de doadores atuais. De maneira geral, os algoritmos comparados produziram praticamente os mesmos resultados, com muito pouca discrepância entre eles. Por sua vez, o trabalho em [Steiner 2012] analisa o projeto dos algoritmos de busca e destaca armadilhas típicas na implementação do algoritmo e da estrutura de dados subjacente. Além disso, o artigo analisa as várias etapas e características necessárias na construção de tais algoritmos.

O trabalho apresentado em [Dehn et al. 2016] descreve o algoritmo de busca HapLogic, que é capaz de prever informações sobre os alelos de doadores, seja por falta da informação ou porque a informação na base está em baixa resolução. Essa mesma abordagem já tinha sido implementada em [Bochtler et al. 2008]. Os dois usam métodos estatísticos para prever as informações ausentes ou em baixa resolução. A aplicação *web* Easy-HLA [Geffard et al. 2019] usa estimativas do método estatístico baseado em máxima verossimilhança para inferir haplótipos HLA e, posteriormente, prever

informações de HLA indisponíveis. Ou seja, a ferramenta pode ser usada para prever informações sobre alelos não existentes ou que estejam em resoluções baixa ou média.

Diferente de todos os trabalhos apresentados nesta seção, a abordagem proposta (e implementada) para análise de padrões em banco de dados de doadores de medula óssea não só é capaz de apresentar uma lista com possíveis doadores a partir da tipagem HLA do paciente mas, principalmente, ela é capaz de realizar contagem de alelos, calcular a frequência alélica, inferir o tempo de espera na fila de transplante, entre outras análises.

3. Algoritmos de Busca HLA e suas Especificidades

A busca por um doador HLA não relacionado é um processo complexo, influenciado por fatores, como a semelhança HLA do paciente, idade, raça/etnia, entre outros. Outra dificuldade é que, em função da evolução dos métodos de tipagem (i.e. exame para identificar os alelos do paciente), os dados dos doadores disponíveis nos registros podem variar bastante, contendo informações mais simples, provenientes de testes sorológicos, ou até mesmo informações mais complexas, resultados de tipagem baseados em sequências genotípicas de alta resolução [Dehn et al. 2016].

Esse processo de busca complexo e desafiador é realizado rotineiramente nos registros de doadores por um programa de computador especializado, no qual o algoritmo de busca HLA pode ser considerado o elemento central [Bochtler et al. 2016]. Uma vez que a compatibilidade HLA, dentre os muitos fatores para se conseguir um transplante bem sucedido, é a mais importante, ela é o principal critério adotado pelos algoritmos de buscas. Porém outros critérios de preferências secundárias, como sexo, idade, raça/etnia, entre outros, devem estar disponíveis pois também podem ter impacto no sucesso ou fracasso do transplante [Tiercy 2016, Dehn et al. 2019]. O resultado desta tarefa é uma lista ordenada limitada de potenciais doadores. Assim, é de suma importância a identificação rápida (dado caráter de urgência, visando salvar pacientes, que essas buscas possuem) e confiável de potenciais doadores voluntários adultos para pacientes individuais (ou vários pacientes).

Para auxiliar na seleção de doadores, a maioria dos registros em todo o mundo desenvolveu seu próprio algoritmo de busca com base nas suas experiências e capacidades. De maneira geral, é esperado que todo algoritmo de busca apresente algumas características importantes, que são descritas a seguir [Steiner 2012]: (1) Determinístico: a mesma entrada deve sempre levar aos mesmos resultados; (2) Ordenação: os resultados devem ser ordenados de acordo com critérios pré-definidos; (3) Exaustivo: todas as entradas do banco de dados devem ser utilizadas na pesquisa. (4) Escalável: os tamanhos dos bancos de dados podem variar significativamente em tamanho e o algoritmo deve ser capaz de lidar com isso; (5) Rápido: não só a exatidão é importante, mas também a velocidade de busca; (6) Configurável: deve ser possível definir critérios de correspondência HLA paciente-doador e critérios de preferência secundários (sexo, idade, raça/etnia, atualização/acessibilidade ao doador, entre outros).

A compatibilidade doador-receptor é definida pelo número de alelos compartilhados nos *loci* HLA-A, -B, -C, -DRB1 e -DQB1, onde cada *locus* é formado por dois alelos. Portanto, os critérios de correspondência indicam a quantidade (e, opcionalmente, a localização) de incompatibilidades (*mismatch*) permitidas para cada paciente pesquisado [Zachary and Leffell 2016]. As buscas devem considerar uma combinação de alelos

relevantes para transplante. Normalmente, as seguintes opções são utilizados: (i) 6/6 ou 5/6: HLA-A, -B e -DRB1; (ii) 8/8 ou 7/8: HLA-A, -B, -C e -DRB1; (iii) 10/10 ou 9/10: HLA-A, -B, -C, -DRB1 e -DQB1; (iv) 12/12 ou 11/12: HLA-A, -B, -C, -DRB1, -DQB1 e -DPB1. As opções 5/6, 7/8, 9/10 e 11/12 permitem que um dos alelos do doador seja diferente do receptor. Como cada *locus* possui dois alelos, a compatibilidade é $2 \times quantidade\_loci$.

Um outro fator de grande influência nas buscas é o que é chamado de resolução da tipagem. A obtenção dos códigos genéticos está sujeita a fatores econômicos, visto que depende de exames laboratoriais custosos e a técnica utilizada tem grande impacto no detalhamento do código genético obtido, gerando cadastros em resoluções diferentes: baixa, média e alta. A resolução baixa determina apenas o grupo alélico. Por sua vez, a resolução média produz uma lista com os possíveis códigos alélicos. Por fim, a resolução alta determina o código alélico definitivo do doador. O ideal é encontrar um doador compatível utilizando alta resolução para evitar problemas pós-transplante [Lee et al. 2007]. Porém, por motivos econômicos, a tipagem HLA de uma grande parte dos registros dos bancos de dados de doadores de órgãos está, em geral, apenas em baixa ou média resolução. Além do nível de resolução, alguns alelos podem ser compatíveis para o transplante baseados nas características apenas em uma determinada região da molécula sendo agrupados com a letra P ao final (e.g. HLA-A*02:01P inclui -A*02:01/A*02:09/A*02:66) ou diferirem na sequencia intrônica (3º ou 4º campos) sendo anotados com uma letra G ao final após o 3º Campo (ex: A*02:01:01G inclui todos os alelos A*02:01).

Para exemplificar a diferença que a resolução provoca nas buscas, alguns exemplos são apresentados a seguir. A Tabela 1 apresenta o resultado da busca em baixa resolução pelo alelo **HLA-A*02**, permitindo uma incompatibilidade em um dos alelo para 5 doadores diferentes que foram digitados em resoluções diferentes. O doador 1 foi tipificado em baixa resolução, os doadores 2, 3 e 4 em média resolução e doador 5 em alta resolução. Repare que para os alelos dos doadores em média resolução existe ambiguidade, ou seja, o exame utilizado não permitiu definir com precisão o alelo do doador. Como pode ser visto, a pesquisa considera apenas os dois primeiros dígitos da tipagem HLA do doador (baixa resolução). Logo, essa busca retornará uma quantidade considerável de acertos (*matches*), pois incluirá registros que foram tipificados em baixa resolução.

Tabela 1. Exemplo de busca por HLA-A*02 (Baixa Resolução).

id	dna_a_1n	dna_a_2n	MATCHED?
1	/02/	/31/	True
2	/02:01/02:01L/02:01Q/...	/33:01/33:03/33:03Q/...	True
3	/29:02:01G/29:02P/29:02/	/30:02:01G/30:02P/30:02/	False
4	/01:01:01G/01:01P/01:01/	/02:01:01G/02:01P/02:01/	True
5	/01:01/	/02:01/	True

Por sua vez, a Tabela 2 apresenta um exemplo de busca por **HLA-A*02:01** considerando resolução média/intermediária, permitindo uma incompatibilidade em um alelo para os mesmos 5 doadores apresentados anteriormente. Como pode ser visto, a busca agora considera não apenas os dois primeiros dígitos da tipificação do HLA do doador, mas também os que seguem o símbolo "**:**" e os campos com ambiguidades de alelos.

É importante enfatizar que esta é a busca computacionalmente mais exigente, pois os campos do registro digitados na nomenclatura definida pelo NMDP (*National Marrow Donor Program*) precisam ser expandidos antes da busca, o que adiciona uma sobrecarga considerável. Doadores tipificados em baixa resolução não são considerados.

Tabela 2. Exemplo de busca por HLA-A*02:01 (Resolução Média/Intermediária).

id	dna_a_1n	dna_a_2n	MATCHED?
1	/02/	/31/	False
2	/02:01/02:01L/02:01Q/...	/33:01/33:03/33:03Q/...	True
3	/29:02:01G/29:02P/29:02/	/30:02:01G/30:02P/30:02/	False
4	/01:01:01G/01:01P/01:01/	/02:01:01G/02:01P/02:01/	True
5	/01:01/	/02:01/	True

Por fim, a Tabela 3 apresenta um exemplo de busca pelo alelo **HLA-A*02:01** em alta resolução considerando os mesmos parâmetros das buscas anteriores. Como pode ser visto, a pesquisa agora considera uma correspondência exata com a *string* "02:01", descartando doadores digitados com ambiguidades (i.e. **id** 2, 3 e 4) e em baixa resolução (i.e. **id** 1).

Tabela 3. Exemplo de busca por HLA-A*02:01 (Alta Resolução).

id	dna_a_1n	dna_a_2n	MATCHED?
1	/02/	/31/	False
2	/02:01/02:01L/02:01Q/...	/33:01/33:03/33:03Q/...	False
3	/29:02:01G/29:02P/29:02/	/30:02:01G/30:02P/30:02/	False
4	/01:01:01G/01:01P/01:01/	/02:01:01G/02:01P/02:01/	False
5	/01:01/	/02:01/	True

4. Uma Abordagem para Análise de Padrões em Banco de Dados de Doadores de Órgãos

Conforme descrito detalhadamente no capítulo anterior, os algoritmos de busca basicamente procuram um doador compatível, avaliando se todos as informações de entrada (pacientes) coincidem com as que estão no banco de doadores. Portanto, os algoritmos de busca não são capazes de realizar análises em um banco de dados de doadores de órgãos, ou seja, realizar tarefas importantes como: comparações antropológicas/populacionais; inferir o tempo de espera para pacientes que necessitam de transplante de medula óssea; Comparação da prevalência de determinados alelos; Contagem de alelos comuns (C), intermediários (O), bem documentados (WD) e não comuns e bem documentados (NCWD); número de campos do nível de resolução alélica, entre outras análises.

Assim, esta seção descreve a abordagem proposta neste trabalho para análise de registros de doadores de órgãos. Como pode ser visto na Figura 1, a solução proposta é composta de vários módulos, cada um tratando de uma função específica para solucionar o problema em questão. A seguir, são descritas as tarefas de cada um dos módulos.

A interface gráfica (do inglês, *Graphical User Interface* - GUI), Figura 1.(1), permite aos usuários inserirem todas as informações necessárias para realizar a análise no

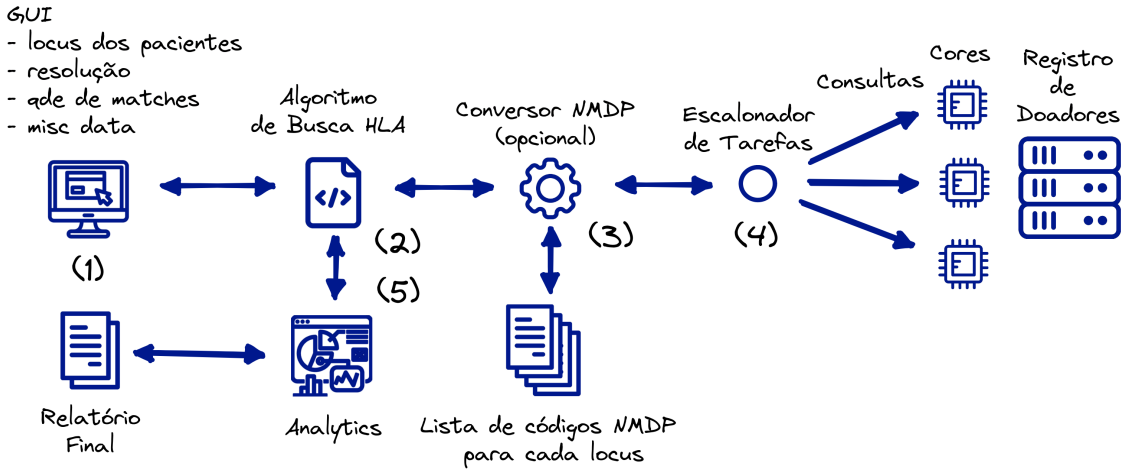

Figura 1. Abordagem Proposta para Análise de Padrões em Banco de Dados de Doadores de Órgãos

registro de doadores. Os principais dados a serem informados são os alelos HLA e a resolução a ser utilizada, podendo ser essa informação sobre um único paciente ou um arquivo contendo múltiplos pacientes. O usuário poderá escolher diversos parâmetros como os *loci* a serem comparados, raça/etnia, localidade do doador, entre outros. Se nenhum parâmetro for escolhido, a pesquisa padrão considerará todo o registro, os três *loci* mais presentes no registro de doadores de órgãos do Brasil (HLA-A, -B, -DRB1) e a mesma resolução da entrada. Outro recurso importante da GUI é fornecer *feedback* sobre o andamento da busca que está sendo executada (i.e. uma caixa de diálogo informando o tempo estimado para terminar a análise).

Uma segunda informação a ser definida pelo usuário para consultas mais avançadas é o tipo de análise a ser realizada. Enquanto para alguns tipos de análise mais simples (e.g a contagem de doadores compatíveis), apenas os parâmetros citados no parágrafo anterior são suficientes, para consultas mais complexas (e.g. inferência do tempo de espera na fila de transplante) o tipo de análise deve ser informado.

Conforme apresentado na Seção 3, o algoritmo de busca tem um papel crucial, sendo uma ferramenta essencial responsável por encontrar potenciais doadores. Na ferramenta proposta neste trabalho, o Algoritmo de Busca HLA (Figura 1.(2)), após receber os parâmetros de entrada escolhidos na GUI, é responsável por gerar as consultas apropriadas para buscar as informações necessárias. Assim que as consultas são geradas, elas são disponibilizadas para o módulo seguinte, o Conversor NMDP.

O Conversor NMDP, Figura 1.(3), recebe como entrada a lista de alelos do NMDP e tem duas funções principais. Se os dados do paciente forem fornecidos pela GUI usando códigos NMDP (ou seja, contêm ambiguidades), este módulo será responsável por expandir o código NMDP nas consultas geradas pelo Algoritmo de Busca HLA. Este módulo também poderá ser usado para expandir os códigos NMPD, caso existam campos com essa nomenclatura nos registros dos banco de dados de doadores de órgãos.

Em seguida, o escalonador de buscas (Figura 1.(4)) é responsável por orquestrar as tarefas (ou seja, consultas) na arquitetura disponível. A principal função deste módulo

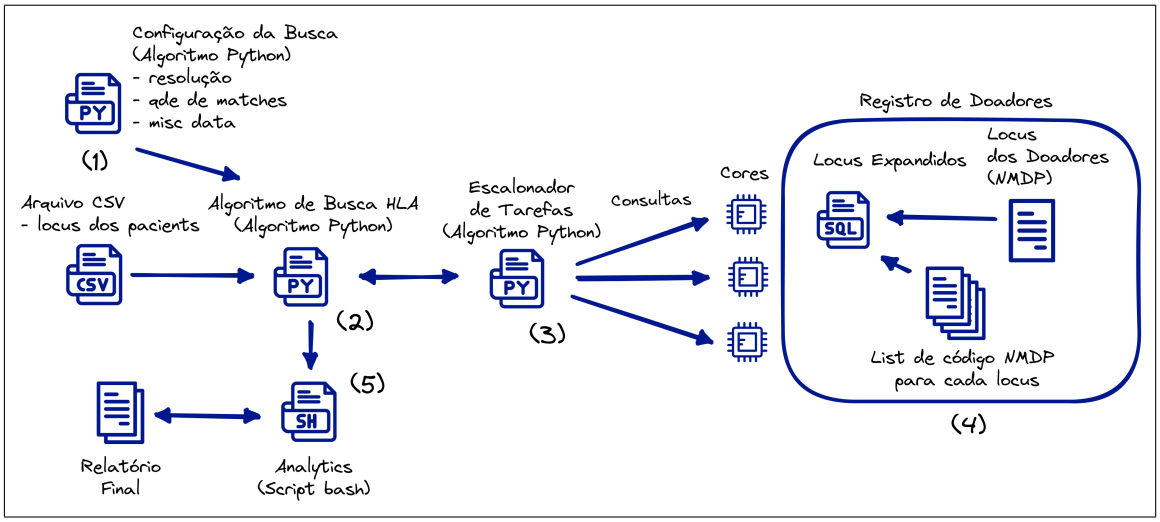

Figura 2. Implementação Inicial da Abordagem Proposta

é escalonar as tarefas explorando todo o potencial da máquina, minimizando o tempo de execução. Este módulo deve ser capaz, não só de agrupar as consultas em tarefas, mas também de escalonar as tarefas na arquitetura disponível.

Por fim, o módulo de análise, Figura 1.(5), a partir dos resultados das consultas, tem a função de gerar os resultados através de relatórios e tabelas. Uma das complexidades de desenvolver esse módulo é a necessidade de unificar (ou separar) os resultados das consultas. Por exemplo, o usuário da ferramenta poderá selecionar uma análise que envolva várias consultas distintas, como por exemplo, em diferentes resoluções. Nesse caso, o módulo deverá apresentar os resultados para cada tipo de busca e, quando possível, unificar os resultados.

5. Implementação Inicial da Ferramenta

A partir da abordagem proposta na Seção 4, esta seção descreve a implementação inicial do *framework* para realizar análise de padrões em banco de dados de doadores de órgãos. O objetivo desta implementação inicial não é apenas mostrar a viabilidade da proposta apresentada, mas servir de base para a criação da ferramenta de busca completa. Uma visão geral do *framework* implementado é apresentado na Figura 2.

A ferramenta de busca foi implementada usando uma série de programas em Python que são capazes de ler os dados de entrada do paciente, assim como os parâmetros de busca fornecidos pelo usuário. Embora a interface gráfica ainda não esteja disponível, a ferramenta já está preparada para receber a saída da interface gráfica, ou seja, um *script* em Python contendo a configuração geral da busca (resolução, número de incompatibilidades, entre outros atributos), Figura 2.(1). Além disso, também são passados os dados dos *loci* do paciente que são carregados a partir de um arquivo CSV.

O programa de busca HLA implementado em Python, Figura 2.(2), utiliza as informações fornecidas pela GUI (Script Python + arquivo CSV) para gerar uma lista de consultas SQL que irão realizar as buscas necessárias. As consultas são definidas por uma série de combinações de *strings*, que levam em consideração o *locus* a ser utilizado (quantidade de incompatibilidades permitidas), resolução e outras informações como et-

nia, estado, sexo, entre outras. É importante ressaltar que nesta implementação inicial não é utilizada uma função para pontuação e ranqueamento dos resultados da busca, uma vez que esses resultados são utilizados para contagens e análises e, não para definir doadores. Assim, a ordem dos registros retornados pelas consultas SQL não interfere nas análises realizadas pela ferramenta proposta.

A lista de consultas é então enviada para o Escalonador de Tarefas implementado em Python, Figura 2.(3), que executa as consultas em paralelo de acordo com os recursos da máquina disponível. A implementação atual permite que o usuário defina a quantidade de *threads* a ser utilizada, sendo a opção padrão a quantidade de núcleos (*cores*) disponíveis.

Com relação ao Registro do Doadores, Figura 2.(4), este trabalho utiliza um subconjunto dos dados do Registro Brasileiro de Doadores Voluntários de Medula Óssea (REDOME), contendo apenas dados genéricos do doador (não há dados pessoais disponíveis) e seu genótipo HLA. O REDOME foi criado em 1993 e, ao longo das últimas décadas, experimentou um crescimento significativo, sendo o terceiro maior registro do mundo com mais de 5,5 de cadastros de doadores voluntários de todo o país.

O banco de dados utilizado foi o MariaDB e é composto por 7 tabelas diferentes e uma visão que combina essas tabelas (Figura 3). O módulo Conversor NMDP apresentado na Figura 1.(3) não foi necessário, pois esta expansão está configurada diretamente no banco de dados. A tabela principal contém informações do doador como local e data de nascimento, sexo, etnia e, especialmente, a tipagem para HLA -A, -B, -C, -DRB1 e -DQB1 com várias resoluções (armazenadas com códigos NMDP). Existem também 5 tabelas separadas com os códigos NMPD, uma para cada *locus*. Essas tabelas são combinadas para produzir uma visão SQL que apresenta todas as informações do doador, considerando o formato dos campos dos *loci* expandidos, segundo os códigos NMDP para nomenclatura HLA da OMS.

6. Análise Experimental

Esta seção avalia os resultados produzidos através da implementação inicial da ferramenta proposta neste trabalho. A ferramenta pode ser executada em computadores pessoais (i.e. *desktop* e *notebooks*), servidores ou até mesmo em uma nuvem computacional. As plata-

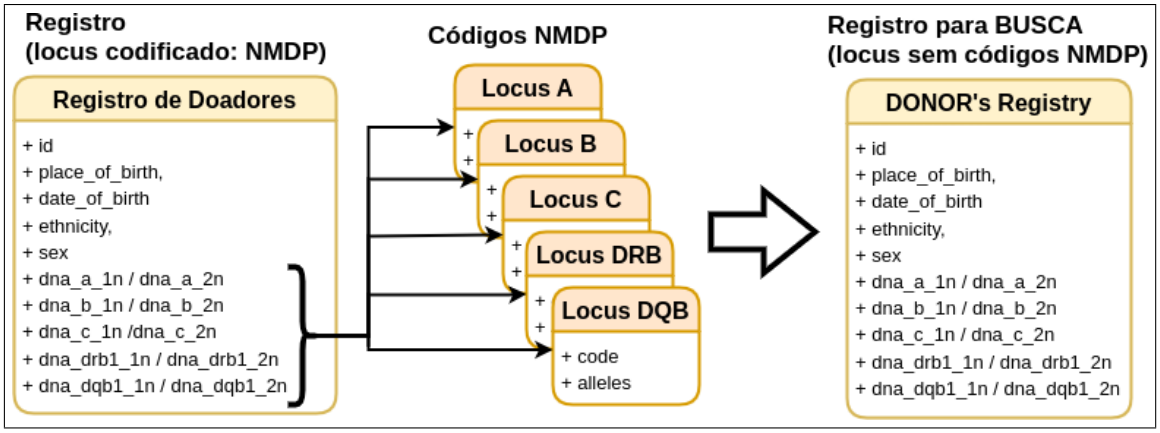

Figura 3. Estrutura do Banco de Dados da Ferramenta Implementada

formas em nuvem são uma opção atraente para executar aplicações paralelas a um baixo custo, possuindo as vantagens conhecidas de disponibilidade e escalabilidade, pagando-se apenas pelo que for utilizado [Bateman and Wood 2009]. Dado que um dos objetivos da ferramenta proposta é acelerar o resultado das análises, esse tipo de ambiente computacional se encaixa perfeitamente. Dessa forma, uma vez que a ferramenta esteja disponível em uma plataforma de nuvem, o usuário poderá escolher uma análise mais rápida a um custo mais alto ou esperar um pouco mais de tempo pelas análises a um custo mais baixo. O importante é aproveitar ao máximo o potencial da máquina (instância) escolhida.

Dessa forma, os experimentos foram executados na *Amazon Elastic Compute Cloud (EC2)*, que oferece diversos tipos de instâncias para todo tipo de computação. Neste trabalho, foram utilizadas as instâncias que podem ser vista na Tabela 4.

Tabela 4. Instâncias utilizadas da Amazon EC2

Instância	CPU	RAM (GiB)	Núcleos	vCPUs	custo/hora
c6i.2xlarge	Intel Xeon 8375C 2.90GHz	16	4	8	0.34
c6i.24xlarge	Intel Xeon 8375C 2.90GHz	192	48	96	4.08

As próximas duas subseções apresentam duas análises distintas executadas com a ferramenta de busca: contagem de doadores compatíveis e frequência alélica. Além disso, para cada uma dessas análises dois aspectos são avaliados: o resultado da análise e o tempo de execução para realizar toda a análise.

6.1. Contagem de Doadores Compatíveis

O primeiro experimento avalia o resultado final de uma contagem de doadores compatíveis. Para isso, a ferramenta recebeu como entrada um arquivo contendo o id e a tipagem HLA para os *loci* HLA-A, B e DRB1 de 100 pacientes. Foram realizadas buscas em baixa (**LOW**) e média (**MEDIUM**) resolução. Conforme explicado na Seção 3, a busca em baixa resolução considera apenas os dois primeiros dígitos de cada alelo (i.e. para o alelo $32:01$ apenas os dois dígitos iniciais, 32, são considerados na busca). Por sua vez, a busca em média resolução considera toda a *string* e, possíveis, ambiguidades. A Tabela 5 apresenta o **id** do paciente, os alelos para cada um dos *loci* escolhido na busca e, por fim, a contagem de doadores compatíveis, tanto para a compatibilidade completa (6/6), como também permitindo um único alelo não compatível (5/6), tanto para baixa, quanto para média resolução. Para facilitar a visualização, apenas os resultados para os 10 primeiros pacientes são apresentados.

Os resultados em baixa resolução mostram que enquanto alguns pacientes possuem vários doadores totalmente compatíveis (e.g o paciente de **id** 7 possui 330 doadores compatíveis), dois pacientes obtiveram apenas 1 doador totalmente compatível (6/6). Por outro lado, para todos os 10 pacientes foram encontrados centenas de doadores compatíveis, quando permitido apenas um alelo diferente (5/6). Entretanto, quando se analisa os resultados da contagem em resolução média, a quantidade de doadores compatíveis diminui consideravelmente, especialmente para doadores totalmente compatíveis (6/6). Neste caso, três pacientes não conseguiram nenhum doador compatível. Por outro lado, foram encontrados centenas de doadores compatíveis nas buscas 5/6. Esses resultados devem ser conjugados com o que foi mencionado na Seção 3: quanto maior a resolução e a quantidade de alelos compatíveis, maiores são as chances para o sucesso do transplante.

Tabela 5. Resultado da contagem de *matches* gerado pelo módulo de análise da ferramenta de busca para as resoluções baixa (LOW) e média (MEDIUM)

id	LOCUS						Total of Matches			
	HLA-A		HLA-B		HLA-DRB1		LOW		MEDIUM	
							5/6	6/6	5/6	6/6
1	32:01	01:01	49:01	08:01	03:01	08:01	884	1	407	0
2	02:01	02:01	14:02	51:01	11:01	15:01	2789	40	704	2
3	29:02	31:01	44:03	57:01	07:01	07:01	2228	55	1593	40
4	03:01	03:01	39:01	44:03	07:01	11:01	334	1	108	0
5	01:01	11:01	08:01	55:01	03:01	14:01	1810	204	1176	129
6	02:01	23:01	15:04	44:03	07:01	14:02	4008	110	516	35
7	01:01	29:02	08:01	44:03	03:01	13:40	9126	330	3146	7
8	29:02	68:02	15:16	44:03	07:01	11:01	4727	130	194	3
9	03:01	32:01	35:01	51:01	01:01	15:02	1346	21	196	0
10	23:01	24:02	35:01	44:03	07:01	01:03	4378	73	470	3

Outra análise importante da ferramenta é o tempo (em segundos) e o custo (em dólar) para realizar a análise, que podem ser vistos na Tabela 6. Eles foram obtidos utilizando a instância $c6i.2xlarge$ (Tabela 4). Conforme descrito na Seção 5, a ferramenta está preparada para dividir as buscas em *threads* distintas, o que permite que computadores com mais de um núcleo possam executar essas *threads* em paralelo.

Tabela 6. Tempo (s) e Custo ($) para cada tipo de busca

№Threads	5/6 LOW		6/6 LOW		5/6 MED		6/6 MED	
	T (s)	C ($)	T (s)	C ($)	T (s)	C ($)	T (s)	C ($)
1	1012.95	0.096	121.12	0.011	7424.05	0.701	145.98	0.014
2	511.08	0.048	62.77	0.006	3753.78	0.355	78.03	0.007
4	258.60	0.024	33.49	0.003	1901.51	0.180	40.56	0.004
8	201.14	0.019	28.95	0.003	1821.66	0.172	36.00	0.003

Como a instância utilizada para executar esse experimento tem 8 vCPUs (4 núcleos) foram executados experimentos com 1, 2, 4 e 8 *threads*. Como esperado, a medida que se aumentou a quantidade de *threads*, o desempenho melhorou proporcionalmente. A exceção foi a execução com 8 *threads*, uma vez que a máquina tem apenas 4 núcleos, onde cada núcleo possui a tecnologia *hyperthreading*, que permite que mais de uma *thread* trabalhe em cada núcleo, mas o ganho de desempenho não é proporcional.

Os resultados mostram que o uso do paralelismo da ferramenta proporcionou uma redução considerável no tempo de execução que, naturalmente, se reflete no custo para executar na nuvem, uma vez que o preço é cobrado pelo tempo de uso das instâncias. Por exemplo, ao utilizar apenas uma *thread* o tempo da busca 5/6 em média resolução foi de ≈ 7424 segundos e custou 70 centavos de dólar. Por sua vez, a busca com 8 *threads* reduziu o tempo de execução para ≈ 1821 segundos e custou apenas 17 centavos de dólar. Ou seja, a execução paralela permitiu economizar dinheiro e reduzir o tempo de execução de ≈ 2 horas para ≈ 30 minutos.

Como o tempo de execução da análise 5/6 em resolução média foi razoavelmente longo, decidiu-se repetir essa mesma análise em uma instância maior, $c6i.24xlarge$. Os tempos de execução utilizando 24, 48 e 96 *threads* foram reduzidos para apenas 358.93, 215.02 e 205.64 segundos, respectivamente. Além disso, o custo para a análise com 96

threads foi de apenas 23 centavos de dólar. Ou seja, comparando com a execução na instância mais lenta ($c6i.2xlarge$), o tempo foi reduzido de $\approx$ 30 minutos para apenas 205.64 segundos e o custo subiu de 17 centavos para apenas 23 centavos de dólar.

6.2. Frequência Alélica

A segunda análise calcula a Frequência Alélica dos alelos dos *loci* HLA-A, B e DRB1 no REDOME em baixa resolução. A entrada foi um arquivo contendo a lista de todos os alelos cadastrados no banco IPD-IMGT/HLA [EMBL-EBI 2022] referentes aos *loci* sendo estudados. O IPD-IMGT/HLA é um banco de dados mundial gerenciado pelo comitê responsável pela nomenclatura de alelos (*Nomenclature for Factors of the HLA System*), entidade mundial responsável pelo cadastro e curadoria de alelos HLA.

A Frequência Alélica é calculada pela razão da contagem direta da quantidade de cada alelo na lista de entrada e a quantidade total de registros no cadastro. O estudo das frequências alélicas de uma população é importante pois apresenta um retrato da diversidade genética. Mudanças nessas frequências podem indicar que uma certa variação genética está ocorrendo ou que novas mutações foram introduzidas na população. A Tabela 7 apresenta o resultado da frequência alélica para os cinco primeiros alelos dos *loci* HLA-A, B e DRB1 em baixa resolução. Nos resultados apresentados o alelo $A*02$ é que possui a maior frequência, 0.449 e o alelo $B*13$ o de menor frequência, 0.031.

Tabela 7. Frequência Alélica (FA) para os alelos dos *loci* HLA-A, B and DRB1 no REDOME em baixa resolução

HLA-A	AF	HLA-B	AF	HLA-DRB1	AF
A*01	0.173	B*07	0.133	DRB1*01	0.188
A*02	0.449	B*08	0.098	DRB1*03	0.186
A*03	0.175	B*13	0.031	DRB1*04	0.234
A*11	0.103	B*14	0.103	DRB1*07	0.242
A*23	0.100	B*15	0.175	DRB1*08	0.120

Como o tempo de execução para realizar esse tipo de análise é muito grande ele foi executado diretamente na maior instância, $c6i.24xlarge$. Os tempos para calcular as frequências alélicas de todos os alelos dos *loci* HLA-A, B and DRB1 foram, respectivamente, 4180.7, 2048.9 e 2326.2 segundos. Por sua vez, os custos em dólar para executar essas análises foram, respectivamente, 4.74, 2.32 e 2.64.

7. Conclusões

Este trabalho propôs e implementou uma ferramenta para realizar análise de padrões em banco de dados de doadores de órgãos. Diferentemente dos algoritmos de buscas, a ferramenta proposta é capaz de realizar análises como contagem de alelos, contagem de doadores compatíveis, frequência alélica, entre outras.

Resultados obtidos com a implementação inicial mostraram não somente a capacidade da ferramenta de realizar as análises, mas também de ser capaz de fazer as consultas em paralelo, o que diminui bastante o tempo de execução e reduz, consideravelmente, o custo em ambientes de nuvens computacionais.

Trabalhos futuros irão implementar novas análises como, por exemplo, inferir o tempo de espera na fila de transplante através do aumento/diminuição da quantidade de

doadores compatíveis ao longo dos últimos anos. Além disso, todos os módulos serão finalizados, permitindo que pesquisadores possam realizar suas pesquisas e, com isso, ajudar a aprimorar o processo de doação de medula óssea entre doadores não relacionados.

Agradecimentos

Os autores agradecem o apoio da CAPES, FAPERJ através do edital APQ1 26/2021 e do projeto CNPq/AWS 440014/2020-4.

Referências

Bateman, A. and Wood, M. (2009). Cloud computing. *Bioinformatics*, 25(12):1475–1475.

Bochtler, W., Beth, M., Eberhard, H., and Mueller, C. (2008). Optimatch (r) - a universally configurable hla matching framework. *Tissue Antigens*, 71(4):321.

Bochtler, W. et al. (2016). A Comparative Reference Study for the Validation of HLA-Matching Algorithms in the Search for Allogeneic Hematopoietic Stem Cell donors and cord blood units. *HLA*, 87(6).

Dehn, J. et al. (2016). HapLogic: A predictive human leukocyte antigen–matching algorithm to enhance rapid identification of the optimal unrelated hematopoietic stem cell sources for transplantation. *Biology of Blood and Marrow Transplantation*, 22(11):2038–2046.

Dehn, J. et al. (2019). Selection of unrelated donors and cord blood units for hematopoietic cell transplantation: guidelines from the NMDP/CIBMTR. *Blood*, 134(12):924–934.

Eapen, M. et al. (2014). Impact of allele-level HLA matching on outcomes after myeloablative single unit umbilical cord blood transplantation for hematologic malignancy. *Blood*, 123(1):133–140.

EMBL-EBI (2022). Ipd-imgt/hla - database. (Acessado em 13/02/2022).

Geffard, E. et al. (2019). Easy-HLA: a validated web application suite to reveal the full details of HLA typing. *Bioinformatics*, 36(7):2157–2164.

Gragert, L. et al. (2014). Hla match likelihoods for hematopoietic stem-cell grafts in the u.s. registry. *New England Journal of Medicine*, 371(4):339–348. PMID: 25054717.

Lee, S. et al. (2007). High-resolution donor-recipient hla matching contributes to the success of unrelated donor marrow transplantation. *Blood*, 13(110):4576–4583.

Singh, A. K. and McGuirk, J. P. (2016). Allogeneic stem cell transplantation: A historical and scientific overview. *Cancer Research*, 76(22):6445–6451.

Steiner, D. (2012). Computer algorithms in the search for unrelated stem cell donors. *Bone Marrow Res.*, 1(1):1–7.

Tiercy, J.-M. (2016). How to select the best available related or unrelated donor of hematopoietic stem cells? *Haematologica*, 101(6):680–687.

Zachary, A. A. and Leffell, M. S. (2016). HLA mismatching strategies for solid organ transplantation - a balancing act. *Frontiers in Immunology*, 7(DEC):1–14.