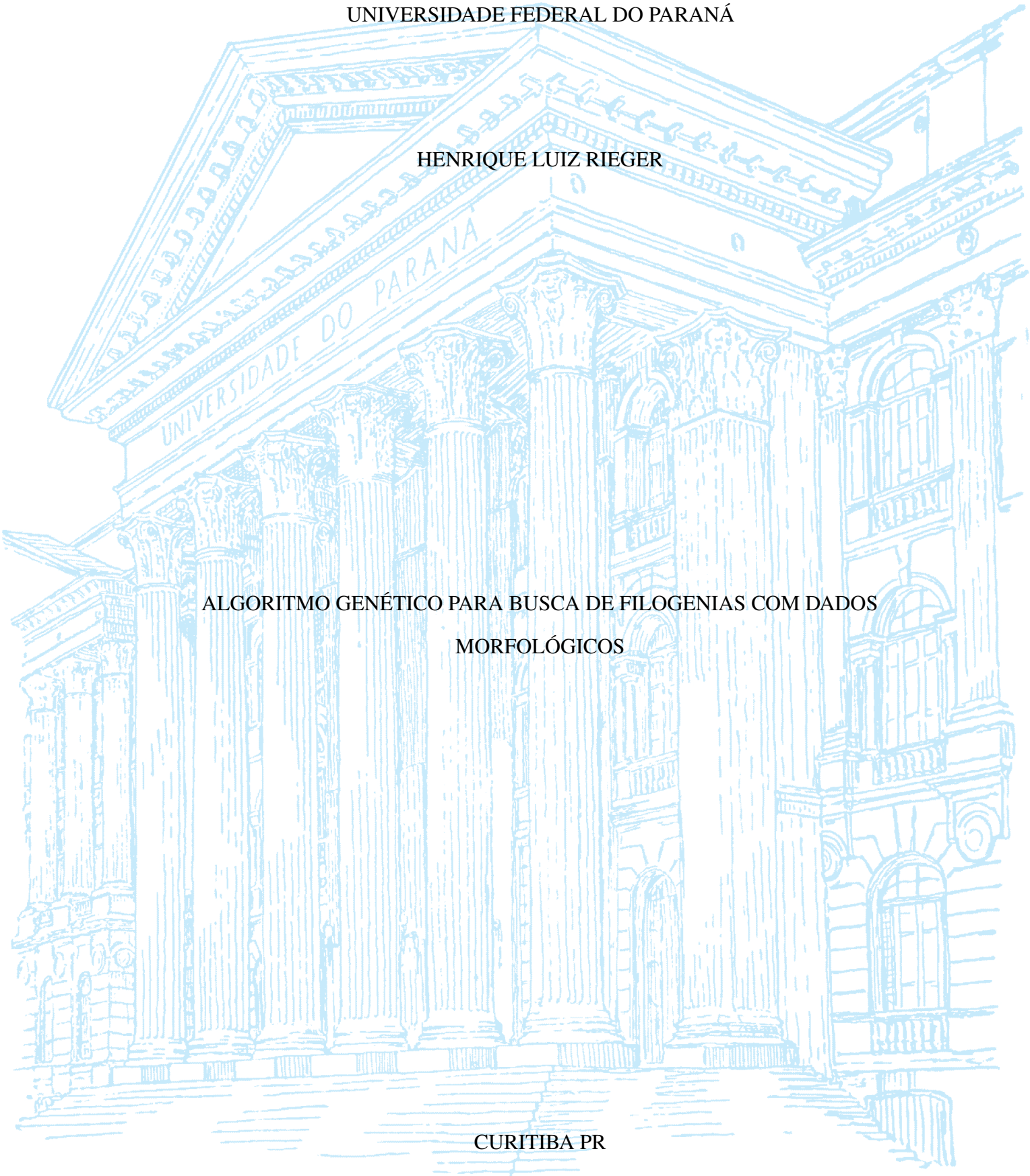
UNIVERSIDADE FEDERAL DO PARANÁ

HENRIQUE LUIZ RIEGER



ALGORITMO GENÉTICO PARA BUSCA DE FILOGENIAS COM DADOS MORFOLÓGICOS

CURITIBA PR

2024

HENRIQUE LUIZ RIEGER

ALGORITMO GENÉTICO PARA BUSCA DE FILOGENIAS COM DADOS MORFOLÓGICOS

Trabalho apresentado como requisito parcial à conclusão do Curso de Bacharelado em Ciência da Computação, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Computação*.

Orientador: Eduardo Jaques Spinosa.

CURITIBA PR

2024

À minha irmã Luiza, minha companhia de todos os dias e luz da minha vida

AGRADECIMENTOS

Gostaria de agradecer primeiramente ao meu orientador, prof. Eduardo Spinosa, tanto por aceitar a minha ideia maluca de misturar Ciência da Computação e Paleontologia no meu TCC, quanto por me acompanhar e me apoiar durante toda essa jornada. Quero deixar também meus agradecimentos ao prof. Fabricius Domingos, do Departamento de Zoologia da UFPR, com quem conversei múltiplas vezes e que me ajudou com ideias e sugestões para este trabalho.

Agradeço também aos membros do Laboratório de Paleontologia da UFPR, em especial ao prof. Robson Tadeu Bolzon, à profª. Cristina Silveira Vega e ao doutorando Malton Carvalho Fraga, que me acolheram no laboratório nesses últimos 18 meses e me permitiram vivenciar um pouco do que é ser paleontólogo. Graças a vocês, realizei meu maior sonho de infância.

Por fim, mas não menos importante, preciso agradecer a meus amigos e familiares, que tiveram que me ouvir falar entusiasticamente sobre os tópicos do meu trabalho várias vezes, ao mesmo tempo que compreenderam minha ausência em outros momentos. Em especial, devo agradecer a minha mãe Katia e meu pai Regis, que não só me apoiaram de coração em todos os momentos, como arranjaram tempo para dar sugestões e me auxiliar na escrita do texto final. Vocês são minha maior inspiração.

RESUMO

A busca por árvores filogenéticas é um dos problemas mais importantes no campo da Biologia, sendo computacionalmente complexa e normalmente só podendo ser feita por heurísticas. Algoritmos genéticos se mostraram uma forma eficiente de percorrer um amplo espaço de busca e obter boas filogenias. Este trabalho apresenta o GASPAR, um algoritmo genético projetado para realizar inferência filogenética com base em dados morfológicos, usados principalmente em contextos paleontológicos, pelo critério de máxima parcimônia. O desempenho desse algoritmo foi testado em relação a abordagens mais tradicionais (i.e. *branch and bound* e *hill climbing*), implementadas no mesmo código, utilizando bases de dados sintéticas e empíricas. Ainda que a performance do novo método para as instâncias testadas seja similar à da heurística *hill climbing*, ainda é ligeiramente inferior, mas provou ser uma alternativa promissora.

Palavras-chave: Algoritmos genéticos. Inferência filogenética. Morfologia.

ABSTRACT

Searching phylogenetic trees is one of the most important problems in the field of Biology, being computationaly complex and usually achievable only by heuristics. Genetic algorithms have been shown to be an efficient way of traversing the search space and obtain good phylogenies. This work introduces GASPAR, a new genetic algorithm designed for phylogenetic inference based on morphological data, mainly used in paleontological contexts, by the maximum parsimony criterion. The efficiency of this algorithm was tested against more traditional approaches (i.e. branch and bound and hill climbing) implemented in the same code, using synthetic and empirical databases. Although the performance of the new method for the tested instances is similar to that of hill climbing, it is still slightly below it, but has proven to be a promissing alternative.

Keywords: Genetic algorithms. Phylogenetic inference. Morphology.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ACRÔNIMOS

AG	Algoritmo Genético
CFI	*Consensus fork index*
GASPAR	*Genetic Algorithm for Searches under Parsimony*
HC	*Hill climbing*
IB	Inferência Bayesiana
MP	Máxima Parcimônia
MV	Máxima Verossimilhança
NNI	*Nearest Neighbor Interchange*
NJ	*Neighbor Joining*
PDG	*Prune-Delete-Graft*
SPR	*Subtree Pruning and Regrafting*
RFn	distância de Robinson-Foulds normalizada
TBR	*Tree Bissection and Reconnection*
UPGMA	*Unweighted Pair Group Method with Arithmetic Mean*

LISTA DE SÍMBOLOS

$\Delta(I, \Pi)$	Diferença entre o valor de parcimônia do indivíduo I e do melhor indivíduo na população Π
Π	População do algoritmo genético
$Fitness(I, \Pi)$	Aptidão do indivíduo I na população Π, a partir da parcimônia
I	Indivíduo da população do algoritmo genético
$n!!$	Fatorial duplo, equivalente a $n!$ excluindo-se fatores pares
$P(I, \Pi)$	Probabilidade de selecionar o indivíduo I na população Π para a próxima geração
$Pars(I)$	Valor de parcimônia do indivíduo I, conforme o algoritmo de Fitch
s	Força de seleção do algoritmo genético

SUMÁRIO

1 INTRODUÇÃO

Árvores evolutivas apresentam visualmente hipóteses sobre a origem dos seres vivos e intrigam biólogos desde que Darwin concebeu a teoria da evolução. Sua inferência passou a ser metodológica e ganhou maior relevância a partir do desenvolvimento da Sistemática Filogenética por Hennig (1965).

Filogenias não são úteis apenas para contar a história da vida na Terra, mas sua aplicação se estende por diversas áreas da Biologia. Todavia, reconstruir a evolução de um grupo não é simples: as topolgias possíveis crescem de maneira supra-exponencial conforme o número de organismos analisados (Cavalli-Sforza e Edwards, 1967). Independentemente do critério de avaliação usado para a análise, encontrar a melhor filogenia é um problema NP-Completo (Day et al., 1986; Chor e Tuller, 2005), necessitando de heurísticas na maior parte dos casos para ser computada de maneira viável.

Entre algumas das heurísiticas menos exploradas para buscar árvores filogenéticas estão os algoritmos genéticos. Essa classe de algoritmos bioinspirados já se provou bastante útil para inferir filogenias, obtendo bom desempenho quando comparada a métodos tradicionais (e.g. Zwickl, 2006). Além disso, algumas implementações permitem percorrer o espaço de possibilidades otimizando múltiplas métricas simultaneamente (e.g. Zambrano-Vega et al., 2016).

Ainda que a pesquisa em filogenia computacional tenha avançado consideravelmente nos últimos anos, pouco estudo foi feito sobre análises baseadas em características morfológicas (Lee e Palci, 2015; Giribet, 2015). A informação presente nesses dados tem sinal filogenético fraco quando comparado a sequências genômicas (Berger e Stamatakis, 2010), mas compõem o único meio de incluir táxons fósseis em filogenias. Organismos extintos representam mais de 99% dos seres vivos que já passaram pelo planeta (Barbosa et al., 2024) e são de extrema importância para entender as relações evolutivas em um nível mais profundo. No entanto, DNA e outras moléculas orgânicas não são normalmente fossilizados, de forma que resta apenas a morfologia para compreender a evolução desses grupos.

Este trabalho compreende o primeiro passo no desenvolvimento de um novo *software* para análises filogenéticas, utilizando um algoritmo genético para encontrar filogenias de máxima parcimônia a partir de caracteres morfológicos. No mesmo código, foram implementados também métodos de busca mais tradicionais (*hill climbing* e *branch and bound*), permitindo uma visão detalhada do desempenho do algoritmo genético com suas alternativas. As comparações foram feitas sobre dados sintéticos, cuja relação evolutiva "verdadeira" é conhecida, e em um conjunto de dados empíricos (reais), mais representativos do uso prático, mas cuja comparação precisa ser feita sobre uma árvore que também é uma hipótese.

O restante desta monografia está organizado da seguinte forma: O Capítulo 2 trata da fundamentação teórica sobre filogenias e o uso de algoritmos genéticos para sua inferência. Já o Capítulo 3 apresenta o GASPAR, o novo algoritmo genético desenvolvido neste trabalho, e seus detalhes de implementação. No Capítulo 4 é descrita a metodologia usada para os testes de desempenho do GASPAR em comparação algoritmos tradicionais, cujos resultados são apresentados e discutidos no Capítulo 5. Por fim, o Capítulo 6 conclui o texto com comentários finais sobre o desempenho do algoritmo e algumas perspectivas para pesquisas futuras.

2 FUNDAMENTOS

Este capítulo irá introduzir alguns conceitos que serão utilizados durante a monografia, familiarizando o leitor com os temas abordados. Primeiramente, será feita uma apresentação sobre filogenias e seus principais conceitos e desafios. Também serão tratadas as diversas formas de se realizar uma análise filogenética, desde funções de avaliação e métodos de busca até a junção e análise estatística dos resultados. Por fim, será também introduzido o funcionamento geral de um algoritmo genético, tema principal deste trabalho, e como o mesmo pode ser usado no contexto de análises filogenéticas.

2.1 FILOGENIAS

Uma filogenia, ou árvore filogenética, é uma hipótese sobre a história evolutiva de um grupo de seres vivos (Felsenstein, 2004). Ela é representada por um grafo conexo acíclico em que os vértices de grau 1 (folhas) representam os táxons de interesse da análise, os seres cuja relação deseja-se descobrir. Os vértices internos, de grau maior que 1, representam táxons hipotéticos ancestrais de outros nós da árvore. Os nós internos são por vezes chamados de HTUs (em inglês, *hypothetical taxonomical units*) e os nós-folha são chamados de OTUs (em inglês, *operational taxonomical units*, Sneath e Sokal, 1962). As figuras 2.1 e 2.2 apresentam exemplos de árvores filogenéticas.

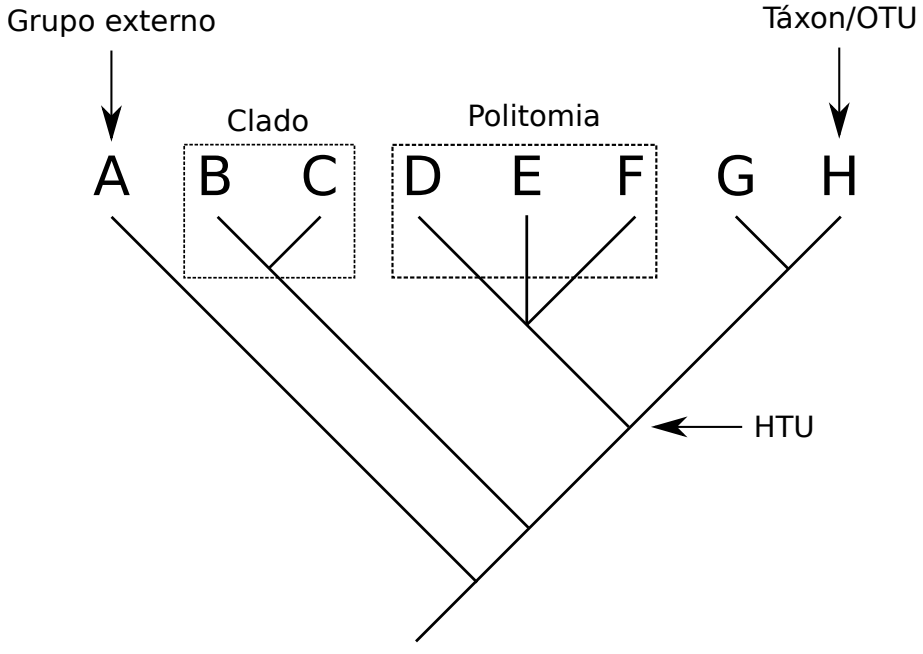


Figura 2.1: Exemplo de árvore filogenética enraizada. As setas e caixas pontilhadas indicam exemplos de algumas das definições abordadas.

Normalmente, uma árvore filogenética é representada enraizada, de forma que sabe-se qual o táxon mais ancestral e quais são seus descendentes. A relação de ancestralidade pode ser definida utilizando informação externa (com a inclusão de grupos externos à análise), ou com o emprego de técnicas como relógios moleculares (Felsenstein, 2004). No entanto, uma filogenia pode ser representada de maneira não-enraizada, isto é, sem que se saiba qual a "direção" da evolução no diagrama, apenas quais seres vivos estão relacionados entre si (Kidd e

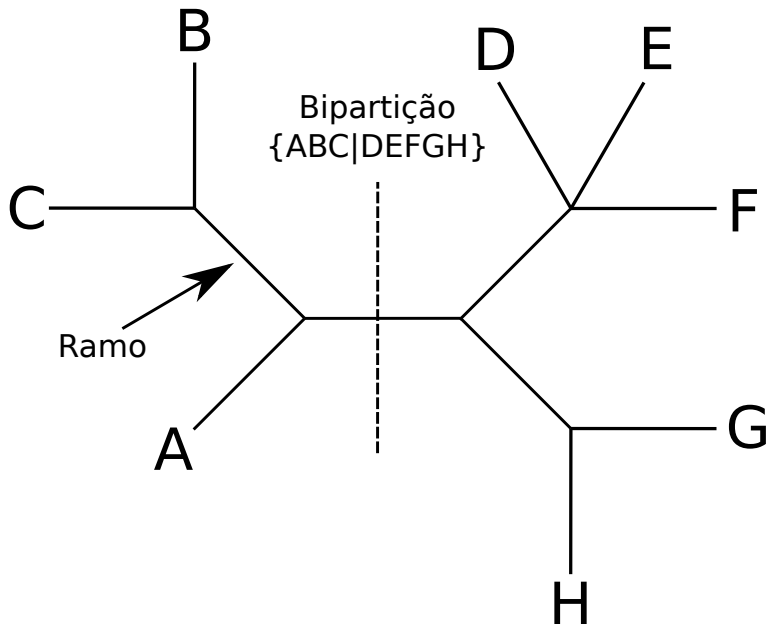


Figura 2.2: Exemplo de árvore filogenética não-enraizada, usando a mesma topologia da figura 2.1. As setas e linhas pontilhadas indicam exemplos de algumas das definições abordadas.

Sgaramella-Zonta, 1971). Em uma árvore enraizada, o conjunto de nós que contém um táxon ancestral e todos os seus descendentes é chamado de clado. Já em uma árvore não-enraizada, é comum se referir às bipartições que um ramo (aresta) implica na árvore, uma vez que a relação de ancestralidade é desconhecida.

Em geral, clados ou bipartições partem de um vértice de grau 3, ou seja, filogenias compreendem árvores binárias. Este modelo de evolução em que um táxon se diversifica em dois (bifurcação) é considerado mais simples e preferível a irradiações em mais táxons (multifurcações ou politomias). É comum, no entanto, que filogenias sejam apresentadas com diversas multifurcações, seja por falta de informação para distinguir uma bifurcação (politomia branda) ou por ramificações em múltiplas linhagens (politomia rígida; Maddison, 1989).

Para este trabalho, uma filogenia de n táxons será uma árvore não-enraizada com n vértices de grau 1 e $n - 2$ vértices de grau 3, (perfeitamente binária). Multifurcações serão utilizadas apenas quando for produzido um consenso entre múltiplas árvores.

2.1.1 Importância

Filogenias compõem a base de toda uma sistemática baseada apenas nas relações evolutivas entre os seres vivos e não apenas em sua semelhança morfológica. Hennig (1965) discorre:

> "In morphological systems, the 'beginner' which belongs to each group is a formal idealistic standard ('Archetype') whose connections with the other members of the group are likewise purely formal and idealistic. But, in a phylogenetic system, the 'beginner' to which each group formation relates is a real reproductive community which has at some time in the past really existed as the ancestral species of the group in question, independently of the mind which conceives it, and which is linked by genealogical connections with the other members of the group and only with these."

Dessa forma, análises filogenéticas são uma das pedras fundamentais da Biologia moderna. Todavia, sua utilidade se estende além da sistemática, permeando aplicações em

biogeografia, epidemiologia, alinhamento múltiplo de sequências e estudos de conservação ambiental (Hennig, 1965; Cotta e Moscato, 2002; Zwickl, 2006).

2.1.2 Entrada e codificação

Uma análise filogenética recebe como entrada uma matriz $n \times m$, com n táxons e m caracteres. Cada táxon (linha) na matriz corresponde a uma unidade taxonômica da análise – como um gênero, uma espécie ou mesmo um espécime de determinado ser vivo – enquanto cada caracter (coluna) representa uma unidade de informação hereditária sobre o ser vivo em questão, que pode partir de diversas fontes. A análise consiste em inferir, da melhor maneira possível, a hipótese de qual topologia de árvore melhor explica os caracteres observados.

Tradicionalmente, matrizes de caracteres são construídas a partir de uma série de traços morfológicos presentes nos táxons de interesse, o que requer um extenso trabalho de coleta e análise por parte dos pesquisadores envolvidos (Giribet, 2015). Essas características são geralmente agrupadas em valores discretos, muitas vezes binários, mas podendo ser multiestado ou até mesmo contínuos (Felsenstein, 2004). Os caracteres podem também ser ordenados (aditivos) ou não-ordenados (não-aditivos), caso seja conhecido qual o estado ancestral dos mesmos.

Nas últimas décadas, dados moleculares tornaram-se as fontes de dados mais comuns para análises filogenéticas, tanto pela disponibilidade e custo de obtenção quanto pela quantidade de informação que são capazes de fornecer, ao ponto que hoje são considerados a norma no campo (Giribet, 2015). Esses dados podem consistir tanto de sequências genômicas (DNA/RNA) quanto proteômicas (proteínas/aminoácidos), agrupadas em um alinhamento de sequências.

Embora em desuso para grande parte das pesquisas envolvendo filogenia, dados morfológicos ainda são de extrema importância, uma vez que representam a única fonte de informação disponível sobre táxons fósseis (extintos), e portanto, para pesquisas paleontológicas. A inclusão de fósseis em análises filogenéticas com táxons viventes também pode melhorar os resultados finais (Berger e Stamatakis, 2010; Mongiardino Koch et al., 2021). Além disso, o uso de caracteres morfológicos serve como teste para os resultados de uma filogenia gerada usando dados de sequências, bem como as informações de fósseis podem ser integradas para melhorar as estimativas de datação da árvore (Lee e Palci, 2015; Giribet, 2015). Dessa forma, a pesquisa de métodos adequados para análises com dados morfológicos ainda se faz bastante necessária.

2.2 ANÁLISES FILOGENÉTICAS

Para poder inferir uma filogenia a partir de matrizes de caracteres, é preciso desenvolver algum método de busca e avaliação com base nos dados de entrada. Um dos primeiros critérios utilizados para aferir a qualidade de uma árvore é o da Máxima Parcimônia (MP), que consiste em buscar pela hipótese que se justifica pelo menor número de passos evolutivos, isto é, pela árvore que requer o menor número de mudanças de estado entre ancestrais e descendentes (Edwards e Cavalli-Sforza, 1963). O cálculo da parcimônia de uma árvore, bem como a reconstrução dos possíveis estados ancestrais, podem ser realizados pelo algoritmo de Fitch (1971).

Além da MP, técnicas baseadas em distância também podem ser empregadas, em que é calculada uma matriz adicional de distâncias entre os táxons de entrada, a partir de suas sequências. Essa nova matriz é então usada como base para calcular a qualidade das filogenia obtidas. Tais métodos por vezes são criticados por não aproveitarem completamente a informação presente nos dados de entrada (Cotta e Moscato, 2002). Entre eles, enquadram-se a técnica de Mínimos Quadrados (Cavalli-Sforza e Edwards, 1967; Fitch e Margoliash, 1967) e de Mínima Evolução

ou Caminho Mínimo (Kidd e Sgaramella-Zonta, 1971). Os algoritmos de *clustering* UPGMA e *Neighbor-Joining* (Sneath e Sokal, 1962; Saitou e Nei, 1987) servem como aproximações rápidamente calculáveis, assumindo-se certas restrições sobre os dados.

A necessidade de justificativas estatísticas para os resultados obtidos nas análises alavancou o uso e desenvolvimento de critérios baseados em modelos, como a Máxima Verossimilhança (MV; Felsenstein, 1973, 1981) e Inferência Bayesiana (IB; Rannala e Yang, 1996). Ambos se utilizam de modelos probabilísticos de evolução, sendo que na MV é encontrada a árvore que melhor explica os dados (P(dados|árvore)) enquanto na IB é obtida uma amostra da distribuição de probabilidade *a posteriori* (P(árvore|dados)) das árvores filogenéticas sob o modelo e os dados.

Devido à sua maior consistência estatística, métodos baseados em modelo têm sido preferidos nos últimos anos (Zwickl, 2006). No entanto, existe certa dificuldade em definir bons modelos evolutivos para dados morfológicos, de modo que, seja por tradição ou pela falta de modelos estatisticamente robustos, somado à facilidade de compreensão e implementação de algoritmos como o de Fitch, a parcimônia continua sendo usada no contexto morfológico, com a recomendação (ou pelo menos constatação de equivalência) por diversos trabalhos (Goloboff et al., 2018; Schrago et al., 2018; Keating et al., 2020; Barbosa et al., 2024).

2.2.1 Métodos de busca

Embora os métodos detalhados acima descrevam como avaliar a melhor árvore filogenética, falta descrever como deve ser feita a busca entre as possibilidades. O problema de encontrar a melhor filogenia sob o critério de máxima parcimônia é comprovadamente NP-Completo (Day et al., 1986), e sob máxima verossimilhança, NP-Difícil (Chor e Tuller, 2005).

O espaço de busca cresce substancialmente a cada táxon adicionado à análise: Cavalli-Sforza e Edwards (1967) demonstraram que, para um conjunto de n táxons, há $(2n-5)!!$ topologias não enraizadas possíveis, e $(2n-3)!!$ topologias enraizadas. Dessa forma, métodos de busca exaustiva não são viáveis para mais do que alguns poucos táxons, e a busca por árvores não-enraizadas requer menos trabalho do que a busca por topologias enraizadas.

Como possível solução, a técnica de *branch and bound* permite encontrar várias topologias ótimas sem a necessidade de percorrer todo o espaço de busca. Hendy e Penny (1982) descreveram dois algoritmos que permitem incluir recursivamente táxons individuais ou em pares a filogenias avaliandas por máxima parcimônia. Ainda que mais eficiente que uma busca ingênua por todas as possibilidades, seu uso ainda é bastante limitado pelo tamanho da análise e pela ordem de adição dos táxons, de forma que matrizes com poucas dezenas de sequências tornam-se inviáveis de serem analisadas. Logo, o emprego de métodos heurísticos é necessário para entradas maiores.

A técnica de *hill climbing* compreende uma das mais simples e utilizadas heurísticas de busca. Esse algoritmo consiste em analisar, para um determinado ponto do espaço de busca, todos os vizinhos imediatos do mesmo. Caso haja algum vizinho melhor (com menor valor de parcimônia ou maior valor de verossimilhança, por exemplo), ou um conjunto de vizinhos melhores, toma-se o com valor mais alto (ou baixo, em caso de minimização) como novo ponto de partida, "subindo o morro". O processo é repetido até que não haja mais vizinhos com melhor valor, e o ponto com tais características é retornado como resultado. Esse, no entanto, é um ótimo local do problema, e não há garantias de que seja a solução com melhor pontuação possível.

No caso de filogenias, cada topologia de árvore é considerada uma possível solução, podendo ser avaliada por praticamente qualquer um dos métodos descritos anteriormente. Percorrer o espaço de busca ao redor de uma árvore também é desafiador, já que não é trivial definir um "vizinho" para uma filogenia. Para isso, uma série de operadores, usualmente

chamados de operadores de troca de ramos, pode ser definida. Alguns dos mais comuns compreendem o *Nearest Neighbor Interchange*, (NNI), *Subtree Pruning and Regrafting* (SPR) e *Tree Bissection and Reconnection* (TBR), que serão descritos em mais detalhes no próximo capítulo.

2.2.2 Consenso de múltiplos resultados

É bastante comum que critérios como máxima parcimônia e inferência bayesiana recomendem diversas topologias como igualmente válidas para os dados de entrada, bem como a aplicação de alguns métodos de cálculo de suporte, como o *bootstrap* (detalhado mais a seguir), produzam uma série de árvores. Para esses casos, a elaboração de uma hipótese única depende de um consenso entre todas as alternativas. Um dos métodos mais antigos para criação de árvores de consenso é o de Adams (1972), que consiste em criar uma árvore a partir de todas as declarações de três táxons compatíveis entre todas as árvores no resultado. No entanto, sua aplicação está limitada à árvores enraizadas, que não serão utilizadas neste trabalho.

Mais simples e facilmente aplicável que o consenso de Adams é o consenso estrito, que determina que apenas clados ou bipartições encontradas em todas as árvores de resposta devem estar presentes no resultado final. Esse método é comumente utilizado na elaboração de análises por parcimônia (Keating et al., 2020), mas é bastante conservador, colapsando muitos clados em politomias e descartando boa parte da informação presente no conjunto de topologias (Felsenstein, 2004).

Uma possível solução é o consenso por regra da maioria (Margush e McMorris, 1981). Nesse, são tomados todos os grupos presentes em 50% ou mais das análises. Tal modo de consenso permite agregar mais informação à resposta final de maneira menos conservadora, uma vez que considera a frequência das bipartições nos resultados como fonte de informação. Podem ser usadas variações da regra da maioria com cortes superiores a 50%, de modo a serem progressivamente mais conservadoras. Uma árvore com corte em 100% corresponde ao consenso estrito.

2.2.3 Testes estatísticos e suporte

Por se tratarem de hipóteses baseadas em dados e modelos estatísticos, análises filogenéticas são suscetíveis a testes para aferir a qualidade da árvore final. Diversos métodos paramétricos podem ser aplicados, mas como dependem da crença no modelo de evolução, correm o risco de subestimar o cálculo da incerteza (Felsenstein, 2004).

Um teste comumente utilizado para filogenias baseadas em parcimônia é o suporte de Bremer (1988), que consiste em comparar o resultado com árvores sub-ótimas, de forma que o suporte para cada clado corresponde a quantos passos evolutivos adicionais precisam ser considerados no consenso estrito para que o clado seja colapsado. Um dos principais problemas em usar essa técnica é que não há como saber quantos passos extras podem ser considerados como “bom suporte” (Felsenstein, 2004).

Entre testes não-paramétricos, que não dependem da aceitação do modelo evolutivo da análise, encontra-se principalmente o *bootstrap* (Felsenstein, 1985). Nesse, são feitas diversas réplicas da análise original, reamostrando os caracteres da matriz com reposição, obtendo a mesma quantidade de caracteres da matriz. Isso pode ser interpretado como uma mudança dos pesos de cada caracter, de forma que a soma dos pesos ainda se iguala a m. Dessa forma, é possível aferir a quantidade de vezes que cada clado/bipartição aparece num universo de análises mais amplo que o anterior, com grupos mais frequentes obtendo maior suporte estatístico.

Utilizando um consenso por regra da maioria, é possível recuperar apenas clados que tenham mais de 50% de suporte pelo teste.

Como alternativa ao *bootstrap*, a técnica de *jackknife* pode ser aplicada. Essa também consiste em repetir a análise diversas vezes, excluindo aleatoriamente um caracter por vez. Para uso em análises baseadas em parcimônia, Farris et al. (1996) propuseram o *Parsimony Jackknife*, em que cada réplica da análise desconsidera $1/e$ (aprox. 37%) dos caracteres.

2.3 ALGORITMOS GENÉTICOS E FILOGENIAS

Enquanto estratégias de busca como *hill climbing* e *branch and bound* são classicamente utilizadas para inferir a melhor filogenia sob determinado critério, não são os únicos métodos possíveis. Na verdade, essencialmente qualquer método de otimização pode ser aplicado à análise filogenética. Em especial, algoritmos genéticos (AGs; Holland, 1975) são heurísticas que se provaram bastante úteis para essa finalidade.

Esses algoritmos compreendem um método bioinspirado em que a otimização se dá por um processo de evolução simulada, composto por uma população de indivíduos que representam cada um uma possível solução do problema (genoma). Essa passa por diversas iterações de seleção com base em uma função de aptidão. Indivíduos selecionados passam por uma etapa de cruzamento/recombinação, incorporando "pedaços" da resposta desenvolvida por cada um a um novo indivíduo, que sofre pequenas mutações aleatórias que alteram levemente seu genoma. Cada uma dessas iterações é conhecida como geração.

Ao final de uma série de gerações, as pressões seletivas impostas pela função de aptidão devem direcionar a população final a uma solução ótima para o problema imposto. Assim como no *hill climbing*, AGs não garantem que o resultado encontrado será o ótimo global do problema (na verdade, sequer garantem um ótimo local), compreendendo uma busca heurísitica. No entanto, podem escapar de ótimos locais, uma vez que os operadores introduzem certa aleatoriedade ao método, permitindo que o mesmo siga em direções do espaço de busca que aprimoram levemente o valor de avaliação, explorando-o mais completamente (Congdon e Greenfest, 2001).

Alguns potenciais problemas de algoritmos genéticos estão na falta de reprodutibilidade dos resultados em relação ao *hill climbing*, cujo desenrolar é determinístico uma vez escolhido um ponto de partida. Devido à natureza estocástica do AG, execuções partindo de uma mesma população inicial podem levar a resultados completamente diferentes (Zwickl, 2006) [1]. Além disso, o número de parâmetros envolvido na aplicação do algoritmo genético é consideravelmente maior que o caso para o *hill climbing*, envolvendo tamanho e composição da população inicial, função de aptidão, operadores e probabilidades de seleção e mutação e critérios de parada.

2.3.1 Representação da população

Para problemas gerais de otimização, algoritmos genéticos são aplicados sobre genomas representados por uma *string* binária de tamanho fixo, ou um vetor de números reais, caso apropriado. Tal representação se torna difícil para a otimização de filogenias, uma vez que a resposta precisa ser uma estrutura em árvore com restrições bem definidas (Congdon e Greenfest, 2001).

Cotta e Moscato (2002) apresentam um método de codificação e decodificação de árvores filogenéticas em vetores de números inteiros, permitindo a aplicação de operadores mais convencionais sobre os dados. Embora os resultados tenham sido parcialmente promissores, os

[1] Os resultados podem ser repetidos usando-se uma mesma semente para o gerador de números pseudoaleatórios em todas as etapas do processo.

melhores ainda foram obtidos por uma representação direta, isto é, com cada indivíduo sendo composto pela topologia da árvore.

2.3.2 Operadores

A partir de uma representação direta, diversos operadores podem ser aplicados sobre os indivíduos. Para as mutações, uma movimentação aleatória de NNI, SPR ou TBR pode ser aplicada. Zwickl (2006) utiliza os operadores de NNI e SPR no seu algoritmo GARLI, além de uma movimentação SPR especial, podendo controlar a reinserção em ramos progressivamente mais próximos do ponto onde foram retirados, a fim de diminuir o impacto de mutações ao longo da execução. Congdon e Greenfest (2001) e Cotta e Moscato (2002) definem também um operador SWAP, em que dois nós-folha têm suas posições trocadas.

Para o operador de recombinação, Moilanen (1999) introduz o algoritmo *Prune-Delete-Graft* (PDG). O funcionamento desse é bastante semelhante à uma operação SPR, porém sendo realizado de uma árvore para outra, gerando um novo indivíduo. Os terminais presentes no ramo podado são deletados da árvore de destino antes da reinserção. O mesmo operador foi desenvolvido de maneira independente por Congdon e Greenfest (2001).

Quanto ao critério de seleção, o mesmo pode ser tomado de maneira proporcional ao critério utilizado para averiguar a qualidade da filogenia. Zwickl (2006) utiliza uma função exponencial normalizada baseada na diferença entre os valores de MV da população para o melhor indivíduo como função de aptidão.

2.3.3 Aplicações de algoritmos genéticos em inferência filogenética

Moilanen (1999) e Congdon e Greenfest (2001) descrevem algoritmos genéticos (PARSIGAL e Gaphyl, respectivamente) para análises sob o critério de máxima parcimônia. Enquanto o último utiliza apenas o AG na busca, o primeiro se aproveita de pequenas buscas locais usando *hill climbing*.

Zwickl (2006) mostra o funcionamento do algoritmo GARLI, baseado em máxima verossimilhança e utilizando apenas operadores de mutação (sem recombinação), apresentando uma versão sequencial e paralelizada. Brauer et al. (2002) também utilizam paralelismo para melhorar o desempenho do algoritmo GAML. Lemmon e Milinkovitch (2002) demonstram um AG (MetaPIGA) com múltiplas populações interagindo entre si ao longo das gerações.

Em uma abordagem híbrida, Zambrano-Vega et al. (2016) apresentam o *software* MO-Phylogenetics, que utiliza o algoritmo NSGA-II para otimização multiobjetivo dos critérios de MP e MV.

2.4 CONCLUSÃO

Neste capítulo foram abordados alguns conceitos fundamentais sobre filogenias e como é realizada uma análise filogenética, desde métodos de avaliação, busca, consenso e suporte. Foram abordados também alguns dos principais desafios e limitações de cada método. Por fim, foi apresentando o algoritmo genético como heurística de busca alternativa, seus problemas e vantagens, e como o mesmo pode ser usado na inferência filogenética. O próximo capítulo será destinado a descrever em mais detalhes a implementação do algoritmo genético, operadores de mutação e função de aptidão usados nos experimentos deste trabalho.

3 O ALGORITMO GASPAR

Neste trabalho, é proposto um novo *software* para inferência de árvores filogenéticas usando dados morfológicos. GASPAR (em inglês, *Genetic Algorithm for Searches under Parsimony*) é um algoritmo genético simples, porém efetivo, implementado para cumprir essa tarefa. Seu funcionamento é inspirado principalmente no algoritmo GARLI sequencial (Zwickl, 2006). O programa foi escrito em linguagem C (compilado com GCC 9.4.0 usando a *flag* `-O3`), com leitura de entrada em flex 2.6.4 e bison 3.5.1. O código-fonte, assim como os dados e scripts usados nos testes deste trabalho estão disponíveis em Rieger (2024).

GASPAR recebe como entrada um arquivo de configuração contendo a matriz de caracteres e uma série de comandos que permitem múltiplas análises sobre os mesmos dados. Os caracteres são representados por algarismos de 0 a 7 e os símbolos '?' para dados faltantes, '-' para lacunas e algarismos entre colchetes ('[]') para caracteres multiestado (cujo estado real é desconhecido, mas sabe-se quais os possíveis valores). Apesar da representação diferente, o programa trata lacunas e dados faltantes da mesma forma. Os comandos permitem programar análises pelos algoritmos *branch and bound*, *hill climbing* e pelo AG, bem como alterar seus parâmetros, definir um número máximo de árvores na resposta final e a quantidade de réplicas por *bootstrap*.

Ao início de uma busca pelo algoritmo genético, os indivíduos da população são gerados como filogenias aleatórias contendo todos os táxons de entrada. A cada geração, os indivíduos são escolhidos conforme sua aptidão (parcimônia) e passam por um processo de mutação, gerando uma nova população. Indivíduos com os melhores valores de aptidão até o momento são adicionados à resposta final. O algoritmo chega ao fim quando for atingido o número máximo de gerações ou na ocorrência de uma certa quantidade de gerações sem alteração no valor de máxima parcimônia; ambos os parâmetros são definidos pelo usuário. O funcionamento do código está detalhado nos algoritmos 1 e 2. A figura 3.1 apresenta o fluxograma do algoritmo.

Algoritmo 1 GASPAR:

Require: tam_população, max_gerações_sem_mudança
Ensure: resposta
1: resposta ← []
2: população ← []
3: aptidões ← []
4: **for all** $i \in$ [0..tam_população] **do**
5: população[i] ← topologia aleatória
6: **end for**
7: gerações_sem_mudança ← 0
8: **for all** $i \in$ [0..gerações] **do**
9: **if** gerações_sem_mudança > max_gerações_sem_mudança **then**
10: **break**
11: **end if**
12: nova_população ← **Geração**(população, aptidões, resposta, gerações_sem_mudança)
13: swap(população, nova_população)
14: clear(nova_população)
15: **end for**
16: atualizaResposta(população, aptidões) // Atualiza resposta uma última vez
17: **return** resposta

Algoritmo 2 Geração do algoritmo GASPAR:

Require: população, aptidões, resposta, gerações_sem_mudança
Ensure: nova_população
1: nova_população ← []
2: **for all** indivíduo ∈ população **do**
3: aptidões.append(F(indivíduo))
4: **end for**
5: melhor_aptidão ← min aptidões
6: **if** melhor_aptidão < min $\{F(I)|I \in$ resposta$\}$ **then**
7: gerações_sem_mudança ← 0
8: **else**
9: gerações_sem_mudança ← gerações_sem_mudança + 1
10: **end if**
11: atualizaResposta(população, melhor_aptidão) // Adiciona à resposta os membros da população com aptidão igual à da mesma, e a limpa antes caso a aptidão seja menor
12: nova_população.append(arg min (população)) // critério elitista preserva o melhor indivíduo
13: **for all** $n \in$ [1..tam(população)] **do**
14: $I \leftarrow$ amostra(população, aptidão)
15: mutação(I)
16: nova_população.append(I)
17: **end for** // amostra o restante conforme a aptidão e realiza mutações
18: **return** nova_população

3.1 ALGORITMOS ALTERNATIVOS: *BRANCH AND BOUND* E *HILL CLIMBING*

O programa GASPAR permite fazer também análises por *branch and bound* e *hill climbing*, caso o usuário deseje. Essa configuração foi adicionada para permitir a comparação entre o algoritmo genético e as alternativas, todas utilizando um mesmo ambiente de código.

A busca por *branch and bound* foi implementada conforme o algoritmo de adição de táxons individuais descrito por Hendy e Penny (1982). Os táxons são adicionados à topologia conforme a ordem que aparecem na entrada, o que pode afetar significativamente o tempo de computação.

Para o *hill climbing*, o algoritmo implementado foi bastante simplificado. Cada execução começa a partir de uma topologia aleatória, e permite um número de réplicas definido pelo usuário a fim de analisar múltiplos pontos de partida. A cada iteração, são analisados todos os vizinhos de acordo com o critério NNI ou SPR. Em caso de empate entre os melhores vizinhos, apenas um é incluído como ponto de partida para a próxima iteração. O algoritmo é interrompido caso não haja vizinhos de melhor valor de parcimônia, ou seja, ele não avança por "platôs" no espaço de busca.

3.2 APTIDÃO E SELEÇÃO

O processo de seleção do algoritmo genético é feito de maneira elitista, isto é, o indivíduo com melhor aptidão é sempre mantido para a próxima geração intacto, preservando a melhor resposta já obtida. O restante da população é selecionado com probabilidade proporcional à sua aptidão (parcimônia), calculada a partir algoritmo de Fitch (1971). As equações 3.1 e 3.2 expressam a função de aptidão de cada indivíduo I ($Fitness(I)$), correspondente à função exponencial da diferença entre sua parcimônia e a do melhor indivíduo ($\Delta(I, \Pi)$) na população

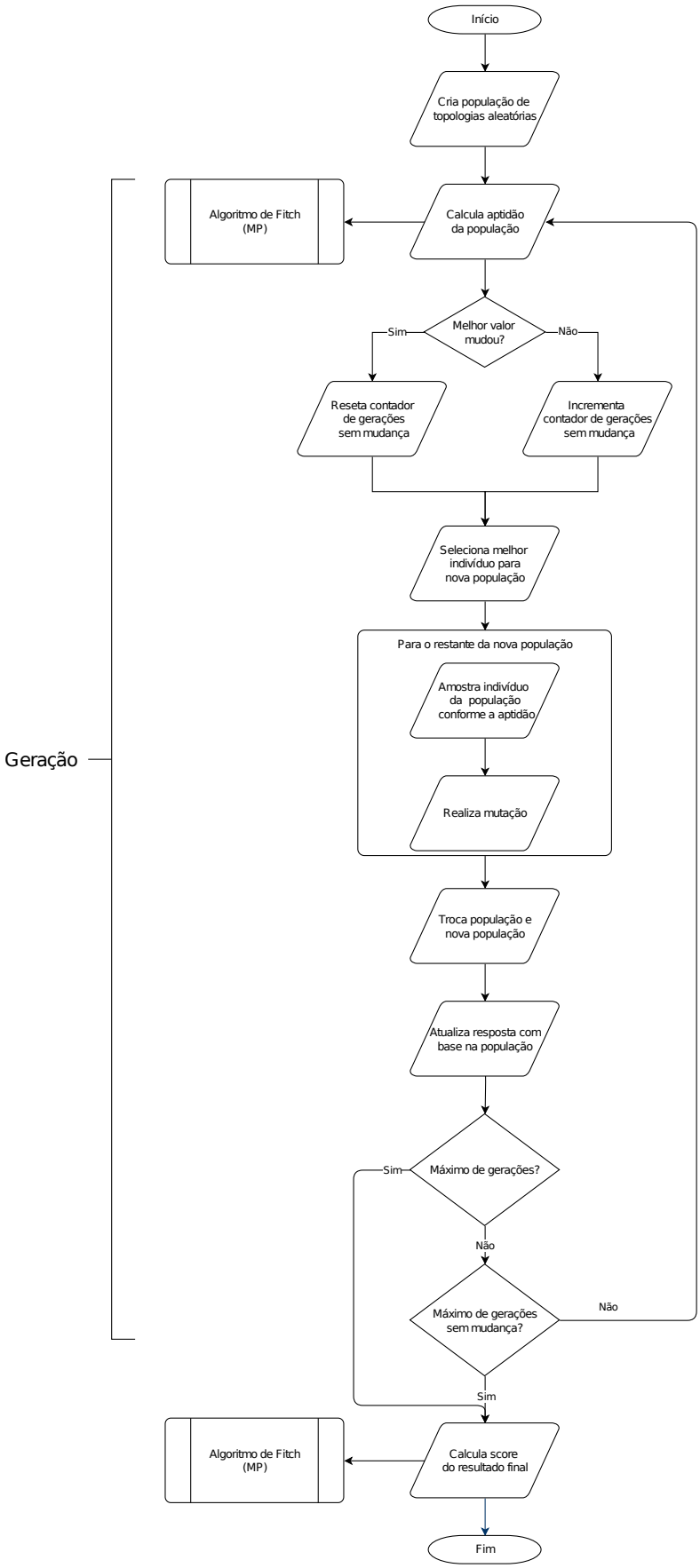


Figura 3.1: Fluxograma do funcionamento do algoritmo GASPAR.

Π, com valor sempre entre 0 e 1 (exatamente 1, no caso do melhor indivíduo, e progressivamente menor conforme a diferença aumenta). A função $Pars$ indica a parcimônia (número de passos evolutivos) do indivíduo. O termo s corresponde ao parâmetro de força de seleção, com valores menores aumentando a chance de seleção de indivíduos menos aptos e vice-versa.

$$\Delta(I, \Pi) = Pars(I) - \min\{Pars(p) | p \in \Pi\} \tag{3.1}$$

$$Fitness(I, \Pi) = e^{-s \cdot \Delta(I, \Pi)} \tag{3.2}$$

A chance de escolha para a próxima geração é dada pela equação 3.3, correspondente à aptidão do indivíduo dividida pela soma do valor de toda a população.

$$P(I, \Pi) = \frac{Fitness(I, \Pi)}{\sum\limits_{p \in \Pi} Fitness(p, \Pi)} \tag{3.3}$$

3.3 REPRESENTAÇÃO DA SEQUÊNCIA EM MEMÓRIA E PARALELISMO

Para diminuir os tempos de computação, o cálculo da parcimônia foi paralelizado conforme o algoritmo detalhado em Moilanen (1999). Assim, cada sequência é representada como um conjunto de 8 vetores binários (0 a 7, uma máscara por estado possível) de mesmo tamanho da sequência. Cada bit na máscara indica se o caracter em determinada posição permite aquele estado. A tabela 3.1 mostra a representação em memória da sequência de 5 estados `02?[1234]7`, com um dado faltante (caracter 3) e um caracter multiestado (caracter 4).

Sequência	0	2	?	[1234]	7
Estado 0	1	0	1	0	0
Estado 1	0	0	1	1	0
Estado 2	0	1	1	1	0
Estado 3	0	0	1	1	0
Estado 4	0	0	1	1	0
Estado 5	0	0	1	0	0
Estado 6	0	0	1	0	0
Estado 7	0	0	1	0	1

Tabela 3.1: Exemplo de representação de sequência em memória.

É importante lembrar que o procedimento descrito por Moilanen foi desenvolvido para caracteres binários, e teve que ser estendido para permitir 8 estados diferentes. Originalmente, as máscaras também eram armazenadas em vetores de `short`, computando 16 posições por vez. Para o GASPAR, as máscaras são armazenadas em vetores AVX2, realizando o cálculo para 256 caracteres simultaneamente.

A soma do número de passos evolutivos também foi otimizada por meio de um algoritmo de programação dinâmica. Ao início da busca, é armazenado um vetor com a soma dos pesos para cada possível configuração de mudança nos caracteres. Diferentemente das sequências, esses valores são definidos para cada conjunto de 8 posições (`char`).

3.4 OPERADORES

Como operadores de mutação, foram implementados dois algoritmos de "rearranjo" (em inglês, "*rearrangement*"; Felsenstein, 2004): *Nearest Neighbor Interchange* (NNI) e *Subtree*

Pruning and Regrafting (SPR). Ambos causam perturbações na topologia da árvore, trocando a configuração de certos ramos internos.

O algoritmo NNI consiste em deletar um ramo interno da árvore, reorganizar os vizinhos em cada ponta da aresta e conectá-los novamente. Suponha um ramo interno que conecte duas sub-árvores (S, T) a duas outras sub-árvores (U, V). Há dois possíveis movimentos NNI nesse caso: um que resulta em uma aresta conectando (S, U) a (T, V) e outro que resulta em uma conexão entre (S, V) e (U, T). A figura 3.2 demonstra essas possibilidades. Para o AG, uma mutação corresponde a qualquer uma das possibilidades de movimento em um ramo interno amostrado uniformemente.

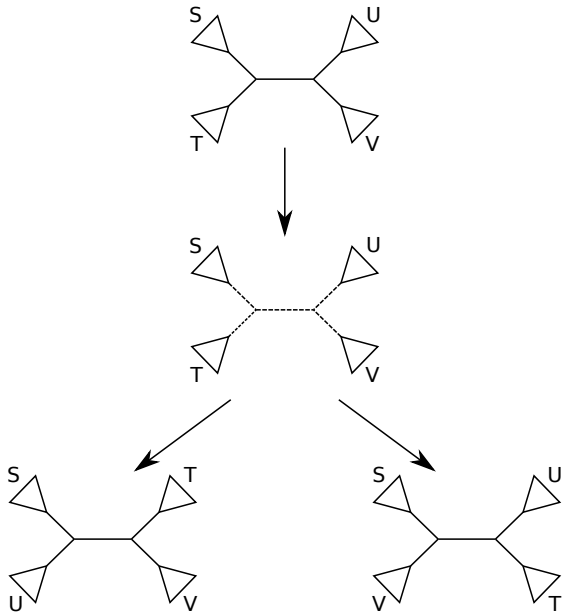


Figura 3.2: Exemplo de funcionamento do operador NNI. Baseado em Felsenstein (2004).

Um movimento pelo espaço do SPR é um pouco mais elaborado. Esse consiste em desconectar uma sub-árvore da topologia principal (poda), deletando os dois outros ramos aos quais a mesma está conectada e criando um novo ramo entre os nós desconectados; e reinserí-la em no meio de outro ramo qualquer da árvore. Tal comportamente está exemplificado na figura 3.3.

No caso do operador para o GASPAR, a implementação da mutação se baseia em um algoritmo de busca em profundidade na árvore partindo da aresta onde houve a poda. A cada ramo percorrido, há uma chance definida pelo usuário no arquivo de configuração de que a sub-árvore seja reinserida naquele ramo. Esse comportamento foi adotado para tornar mutações com grande impacto na topologia final da árvore menos frequentes, uma vez que a maior parte dos movimentos SPR leva a pontos de reinserção distantes da origem (Zwickl, 2006).

Ambos os algoritmos descritos acima também são usadas para buscas por *hill climbing*. Nesse caso, todas as possibilidades de movimentos são analisadas antes de decidir qual topologia será usada na iteração seguinte do processo.

Ainda, foi implementado um operador adicional para o algoritmo genético, denominado **híbrido**. Este escolhe aleatoriamente uma mutação NNI ou SPR usando uma distribuição de Bernoulli com valor p (probabilidade de realizar NNI) definido pelo usuário. Como o NNI representa um sub-caso do SPR, não há necessidade de implementar o operador híbrido para buscas por *hill climbing*.

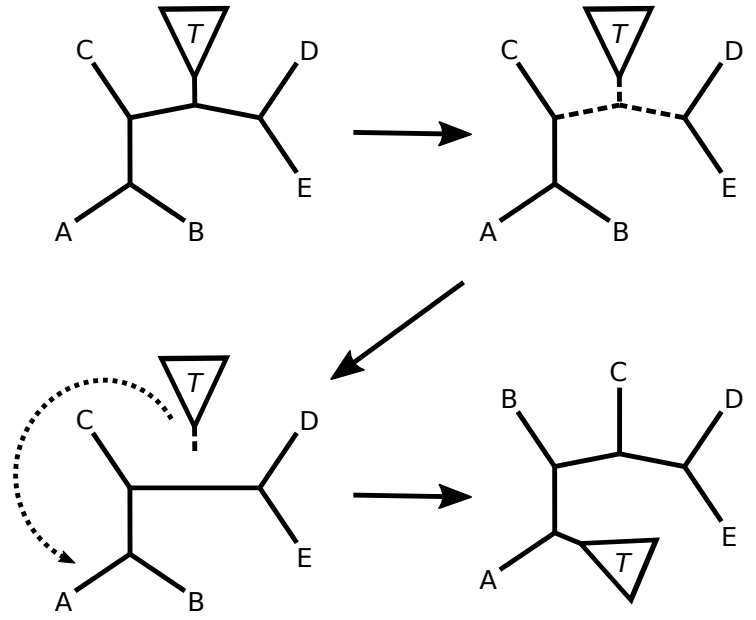


Figura 3.3: Exemplo de funcionamento do operador SPR. A poda é feita na sub-árvore T e a reinserção é feita no ramo que liga A a seu nó-pai.

3.4.1 Ausência de operadores de recombinação

O algoritmo GASPAR não utiliza operadores de recombinação, seguindo o exemplo do *software* GARLI (Zwickl, 2006). Embora tais operadores sejam definidos para árvores filogenéticas (ver *Prune-Delete-Graft* de Moilanen, 1999; também Congdon e Greenfest, 2001 e Cotta e Moscato, 2002), Brauer et al. (2002) argumentam que a recombinação teve pouco impacto em sua implementação paralela do algoritmo GAML, de forma que o mesmo não foi adotado subsequentemente pelo GARLI. Este trabalho, portanto, irá investigar o comportamento de um algoritmo genético baseado apenas no operador de mutação.

3.5 CONCLUSÃO

Neste capítulo, foi apresentado o funcionamento do algoritmo genético GASPAR, bem como de outros métodos de busca, compilados em um *software* de mesmo nome. Foram mostrados detalhes de implementação como operadores de mutação, função de aptidão e a paralelização dos cálculos de parcimônia. Também foi apresentada uma breve justificativa da ausência de um operador de recombinação.

O próximo capítulo trata dos métodos experimentais para medir o desempenho do GASPAR enquanto programa de inferência filogenética.

4 MATERIAIS E MÉTODOS

A fim de testar o desempenho do GASPAR para inferências filogenéticas, múltiplos testes foram realizados. Esses têm como objetivo averiguar a capacidade dos algoritmos em recuperar corretamente filogenias de referência, com base em matrizes de dados morfológicos discretos. A eficácia de cada algoritmo também foi estimada a partir dos valores obtidos dentro do critério de máxima parcimônia, bem como uma aproximação do tempo necessário para obter essas respostas.

4.1 DADOS

O desempenho foi avaliado utilizando uma série de matrizes, tanto sintéticas (geradas por *software*) quanto empíricas (baseadas em dados reais). Dados sintéticos foram utilizados pois permitem averiguar a relação "verdadeira" entre os organismos, porém podem introduzir diversos viéses (Huelsenbeck, 1995). Matrizes reais, por outro lado, apresentam um exemplo mais concreto dos desafios encontrados por programas de inferência filogenética, mas tornam difícil garantir a qualidade dos resultados, uma vez que as comparações precisam ser feitas com outras hipóteses.

4.1.1 Dados sintéticos

Para os experimentos, foram geradas matrizes de diferentes tamanhos, utilizando o programa TREvoSim (Keating et al., 2020). Esse utiliza um algoritmo de simulação de evolução para gerar organismos artificiais e suas relações filogenéticas, de modo bastante similar a um algoritmo genético. Como não necessariamente produz árvores de máxima parcimônia, mesmo métodos exatos (e.g. *branch and bound*) podem não obter a resposta correta.

Os conjuntos de matrizes foram criados com 32 táxons cada, variando a quantidade de caracteres entre 256, 512 e 1024. Cada combinação dessas dimensões forma um conjunto de dados com 30 matrizes cada. As simulações de evolução foram podadas após atingir cinco táxons idênticos, e as diferenças entre espécies foram definidas para 25 no conjunto de 256 caracters, 50 no conjunto de 512 e 100 no conjunto de 1024. Esses valores foram retirados dos testes originais feitos por Keating et al. (2020), com a diferença de que as matrizes de 128 caracteres foram substituídas por matrizes de 256 para poder preencher completamente os registradores AVX2 utilizados no cálculo da parcimônia, conforme descrito no capítulo 3. Caracteres não-informativos foram mantidos pois sua retirada ocasionava problemas na saída do programa, gerando caracteres extras de modo inconsistente.

Com o objetivo de poder utilizar o algoritmo *branch and bound* nas comparações, um conjunto adicional de 30 matrizes de 16 táxons e 256 caracteres também foi obtido. Neste caso, as simulações foram podadas ao atingir 3 táxons idênticos, e a diferença entre espécies foi mantida em 25, também com caracteres não-informativos.

A tabela 4.1 apresenta de maneira concisa as especificações de cada um dos conjuntos de dados.

4.1.2 Dados empíricos

A matriz escolhida como exemplo de dados reais foi o proposta por Barbosa et al. (2024), por incluir também uma árvore de referência concordante com análises realizadas por

Tabela 4.1: Especificações dos conjuntos de dados. Na parte de cima estão os parâmetros das matrizes artificiais geradas com o TREvoSim e abaixo as informações do conjunto real de Hexapoda.

Conj. de dados	Táxons	Caracteres	Limiar idênticos	Dif. entre espécies
Sintéticos	16	256	3	25
	32	256	5	25
	32	512	5	50
	32	1024	5	100
Hexapoda	32	115	-	-

Tabela 4.2: Lista dos parâmetros utilizados nas análises filogenéticas.

Algoritmo	Parâmetro	Valor
Algoritmo genético	Tamanho da população	8
	Nº de gerações sem alteração	20 mil
	Nº máximo de gerações	1 milhão
	Força de seleção *s*	0,5
	Prob. de reinserção SPR	5%
	Prob. NNI no operador híbrido	80%
Hill climbing	Nº de réplicas	8
Bootstrap	Nº de réplicas	100 (1 original + 99 reamostradas)

diversos métodos, e portanto considerada neste trabalho como a árvore "real" (figura 5.5). A matriz contém 115 caracteres morfológicos de 32 táxons do clado Hexapoda, que compreende, entre outros grupos, todas as espécies de insetos. Suas características também se encontram na tabela 4.1.

Para criar um conjunto de dados equivalente aos gerados pelo TREvoSim, a mesma matriz foi analisada 30 vezes de maneira independente. A vantagem dessa repetição é poder averiguar os resultados dos métodos de busca de modo mais independente das condições iniciais, uma vez que ambos apresentam algum fator de aleatoriedade. Essa diferença se mostrou mais significativa para o algoritmo genético, como será apresentado no capítulo 5.

4.2 ANÁLISES FILOGENÉTICAS

Todos os conjuntos de dados foram analisados pelo algoritmo genético descrito no capítulo 3, utilizando os operadores de mutação SPR, NNI e o híbrido entre os dois. Em todas as execuções foi utilizada uma população inicial de 8 indivíduos aleatórios, mutacionada até atingir 20 mil gerações sem alteração na aptidão do melhor indivíduo, por no máximo 1 milhão de gerações. A força de seleção *s* foi fixada em 0,5. Esses parâmetros foram baseados no trabalho de Zwickl (2006), à exceção do tamanho da população, que foi dobrado. A lista de todos os parâmetros usados encontra-se na tabela 4.2.

A chance de reinserção do operador SPR foi fixada em 5%, e o operador híbrido foi configurado para ter 80% de chance de preferir um movimento NNI aleatório a um SPR. Por serem operadores novos, não há valores de referência para esses parâmetros, mas testes preliminares (não descritos aqui) indicaram que esses seriam os mais adequados.

Como método de comparação direta, a implementação do algoritmo de *hill climbing* do GASPAR foi utilizada. A escolha de uma implementação autoral do algoritmo foi tomada para

que ambos utilizassem exatamente as mesmas funções para avaliar a parcimônia e realizar as trocas de ramos, representando uma comparação mais direta entre o desempenho de algoritmos do que de implementações específicas. As buscas partiram de 8 réplicas independentes, de forma a corresponder com o tamanho da população do AG. Igualmente, tanto NNI como SPR foram utilizados como estratégias de troca de ramos. Como todos os vizinhos gerados por movimentos de NNI são também obtidos usando SPR, não há necessidade de uma versão do operador híbrido para buscas por *hill climbing*.

Para ambos os algoritmos, foram realizadas 100 réplicas de *bootstrap* em cada matriz (1 com caracteres originais + 99 com caracteres reamostrados). Embora esse valor possa ser considerado baixo, em contraste com uma escolha mais usual de 1000 réplicas, o estudo realizado por Pattengale et al. (2010) na busca por um método dinâmico de determinar a quantidade necessária de réplicas aferiu que a maioria dos conjuntos de dados analisados atinge convergência dos valores de suporte entre 100 e 500 réplicas, mesmo em matrizes significativamente maiores (ainda que isso seja altamente variável entre diferentes matrizes, mesmo de tamanhos semelhantes). O elevado volume de dados analisados neste estudo também obrigou o emprego de um número menor de réplicas para ser completado em tempo viável.

O desempenho geral de cada algoritmo foi avaliado utilizando tanto a árvore de consenso estrito das filogenias obtidas usando apenas os caracteres originais quanto pelo consenso por regra da maioria de todas as árvores obtidas por *bootstrap*, colapsando ramos com suporte inferior a 50%. Em ambos os casos, o consenso foi calculado utilizando o software `consense` do pacote PHYLIP (Felsenstein, 2005) versão 3.697.

Adicionalmente, as matrizes de 16 táxons foram analisadas utilizando o algoritmo de *branch and bound* com adição sequencial de táxons descrito por Hendy e Penny (1982), também implementado no GASPAR. Devido a questões de tempo de execução (com cada matriz individual podendo tomar vários minutos), não foi possível testar esse método para conjuntos maiores. O suporte por *bootstrap* também não pôde ser aferido, de modo que apenas o consenso estrito dos resultados com caracteres originais foi obtido.

4.3 MÉTRICAS EXTRAÍDAS

Os resultados obtidos pelas análises filogenéticas foram comparados utilizando quatro métricas: acurácia, precisão, valor de parcimônia e número de avaliações. A acurácia e a precisão foram calculados de acordo com as definições costumeiramente aplicadas ao aprendizado de máquina, isto é, tal que a acurácia corresponde a

$$\text{acurácia} = \frac{\text{verdadeiro positivo} + \text{verdadeiro negativo}}{\text{total}} \tag{4.1}$$

enquanto a precisão corresponde a

$$\text{precisão} = \frac{\text{verdadeiro positivo}}{\text{verdadeiro positivo} + \text{falso positivo}}. \tag{4.2}$$

Essa definição exige que esses conceitos sejam definidos para filogenias. Um resultado **positivo** será definido como uma bipartição inferida sobre os dados de entrada, e será representado visualmente como um ramo ligando as partes na árvore final. Um resultado **negativo** será uma bipartição que não consta na árvore, implicando a ausência de um ramo ligando as partes na representação gráfica. Uma **hipótese** será uma das árvores de consenso obtidas pelos algoritmos de inferência, na qual serão avaliados os exemplos positivos e negativos. Uma árvore de **referência**

corresponde à filogenia "correta", e determina se os resultados da hipótese são verdadeiros ou falsos.

4.3.1 Acurácia

A acurácia de uma árvore filogenética pode ser calculada tomando-se a distância simétrica de Robinson-Foulds normalizada (RFn, Robinson e Foulds, 1981) entre a hipótese e a árvore de referência. Essa métrica compreende à menor quantidade de ramos que precisam ser sequencialmente removidos e adicionados para transformar uma árvore em outra, dividido pelo máximo de tais operações que podem ser feitas ($2n - 6$, para n táxons, sequencialmente deletando-se e adicionando-se $n - 3$ ramos internos).

Cada ramo deletado e adicionado no cálculo da distância, pelas definições acima, corresponde a um resultado falso positivo e falso negativo, respectivamente. Dessa forma, o complemento da métrica RFn corresponde aos resultados verdadeiros, isto é, podemos definir a acurácia de uma árvore filogenética como

$$\text{acurácia} = 1 - \text{RFn}. \tag{4.3}$$

Assim, uma árvore com acurácia 1 (100%, acerto total) é igual à árvore de referência, enquanto uma acurácia 0 (erro total) corresponde a uma árvore completemente diferente da referência, sem inferir nenhuma de suas bipartições.

A métrica RFn, sem tomar o complemento, mas considerando 0 o resultado ótimo e vice-versa, foi utilizada como medida de acurácia no trabalho de Barbosa et al. (2024). Keating et al. (2020) alegam que o uso da distância simétrica de Robinson-Foulds "corre o risco de confundir acurácia e precisão", mas utilizando tanto a definição de acurácia quanto as definições de positivos e negativos apresentadas acima, fica claro que o emprego da mesma corresponde de maneira exata ao esperado.

O cálculo da métrica RFn entre o resultado das análises filogenéticas e a filogenia de referência foi feito utilizando o valor `norm_rf` retornado pelo método `compare` implementado na biblioteca Python ETE Toolkit v3 (Huerta-Cepas et al., 2016).

4.3.2 Precisão

Barbosa et al. (2024) definiram sua métrica de precisão a partir do *consensus fork index* (CFI) de Colless (1980), no que chamaram de "precisão *via* resolução". O CFI corresponde ao quociente entre a quantidade de ramos internos de uma árvore e o máximo possível ($n - 3$, para n táxons), de forma a indicar o grau de "resolução" da árvore, com uma árvore perfeitamente binária apresentando valor 1, e um grafo estrela (arbusto) apresentando valor 0. No entanto, o CFI de uma hipótese conta todos os ramos obtidos pela mesma, isto é, todos os exemplos positivos, sejam eles verdadeiros ou falsos. Pela definição da equação 4.2, isso é proporcional apenas ao denominador do cálculo da precisão.

Uma alternativa é avaliar o CFI do consenso estrito entre a hipótese e a árvore de referência. Nesse caso, apenas os verdadeiros positivos são carregados para o consenso, cujo CFI agora é proporcional ao numerador da relação de precisão. Para filogenias de referência perfeitamente resolvidas, essa métrica bastaria para a precisão, mas é possível generalizá-la definindo a precisão como o quociente entre o CFI do consenso com o CFI da hipótese, de forma que:

$$\text{precisão} = \frac{\text{CFI(consenso)}}{\text{CFI(hipótese)}} = \frac{\frac{\text{VP}}{n-3}}{\frac{VP+FP}{n-3}} = \frac{VP}{VP+FP}. \tag{4.4}$$

Neste trabalho, a precisão foi calculada por um script Python utilizando o ETE Toolkit, fazendo a contagem dos nós internos do consenso estrito entre a hipótese e a referência (computado usando o `consense`) e a da hipótese, de forma a não precisar fazer o cálculo direto dos CFIs.

4.3.3 Outras métricas

Além da acurácia e da precisão, também foram avaliados o valor de máxima parcimônia e o número de avaliações realizado por cada busca. O valor de máxima parcimônia corresponde ao número de mudanças evolutivas obtido na árvore mais parcimoniosa obtida, e serve como medida da otimalidade da busca. Já o número de avaliações corresponde à quantidade de chamadas da função `fitchParsimony` dentro do código do GASPAR em cada análise, usada como uma estimativa da computação necessária para obter essa resposta. Para as buscas feitas com o algoritmo genético, o menor valor de parcimônia de cada geração também foi registrado, de modo a poder observar o progresso da busca com cada operador de mutação.

Todas as métricas foram agrupadas para cada conjunto de dados e cada algoritmo de busca. O melhor resultado de cada métrica do algoritmo genético também teve sua significância testada em relação ao melhor resultado do *hill climbing* em cada conjunto de dados utilizando um teste *t* pareado, implementado na biblioteca Python SciPy (Virtanen et al., 2020), versão 1.14.0.

4.4 CONCLUSÃO

Neste capítulo foram detalhados os conjuntos de dados utilizados nos testes e os parâmetros para gerar os dados sintéticos. Também foram apresentados os métodos usados para as análises filogenéticas e os parâmetros definidos para a execução do GASPAR. Por fim, foram definidas as métricas coletadas e como as mesmas podem ser definidas e calculadas.

O próximo capítulo irá focar em apresentar os resultados obtidos pelos experimentos.

5 RESULTADOS

Para todos os experimentos, os resultados foram bastante similares. Em sua maioria, apontam a prevalência do algoritmo *hill climbing* com o operador SPR em todos os aspectos, seguido de perto do algoritmo genético com o mesmo operador para mutação. Por outro lado, o operador NNI obteve piores médias, obtendo melhor desempenho no AG do que no HC. É também notável que as médias de acurácia e precisão são normalmente maiores nos experimentos que utilizaram *bootstrap* e consenso por regra da maioria do que nos que foram feitos apenas com consenso estrito.

A tabela 5.1 mostra os valores médios para o experimento realizado com 16 táxons e 256 caracteres. Nesse em específico, o algoritmo genético obteve maior acurácia e precisão que o *hill climbing*, superando inclusive os valores para a busca exata por *branch and bound*. No entanto, essa diferença não é significativa ($p > 0{,}1$), e deve ser atribuída à natureza estocástica do algoritmo. A quantidade de avaliações realizada pelo HC + SPR, porém, é apenas um décimo das realizadas pelo AG, com p-valor bastante significativo ($\ll 0,01$).

Tabela 5.1: Valores médios para matrizes de 16 táxons e 256 caracteres, com o melhor valor de cada coluna em negrito. Valores para *branch and bound* aparecem abaixo para referência. Siglas – **BnB**: *branch and bound*. **HC**: *hill climbing*. **AG**: algoritmo genético. **acur.**: acurácia. **prec.**: precisão. **parc.**: valor de parcimônia. **aval.**: número de avaliações de aptidão.

		NNI				SPR			
		acur.	prec.	parc.	aval.	acur.	prec.	parc.	aval.
HC	estrito	85,1%	85,8%	306,0	$\mathbf{5,96 \times 10^6}$	91,6%	92,8%	**296,4**	$\mathbf{5,97 \times 10^4}$
	bootstrap	90,9%	**94,1%**			91,6%	92,8%		
AG	estrito	85,7%	90,1%	**300,0**	$1,61 \times 10^7$	91,6%	92,8%	**296,4**	$5,73 \times 10^5$
	bootstrap	**92,1%**	92,9%			**92.1%**	**92,9%**		

	acur.	prec.	parc.	aval.
BnB	91,6%	92,8%	296,4	$4,47 \times 10^7$

Os resultados para os experimentos com 32 táxons estão presentes nas tabelas 5.2, 5.3 e 5.4, para os casos com 256, 512 e 1024 táxons, respectivamente. Essas apresentam um padrão, no qual os valores médios para os casos utilizando o operador NNI são superiores para o algoritmo genético, enquanto para os casos utilizando SPR so *hill climbing* se sobresai, com alguns poucos *outliers* ou valores em que há empate.

Essas diferenças são significativas para as métricas de acurácia e precisão com 256 caracteres ($p < 0{,}05$, com a maioria $< 0{,}01$), à exceção da acurácia para o *bootstrap* ($p \approx 0,06$). Já para o caso de 512 caracteres, a diferença é majoritariamente insignificante ($p \approx 0,08$). Para 1024 caracteres, as diferenças são significativas entre as métricas do consenso estrito ($p \approx 0,02$) mas não são entre as do *bootstrap* ($p \gg 0,05$). A diferença no valor de parcimônia não foi significativa apenas para 512 caracteres ($p \approx 0,08$). Em todos os casos, a diferença de chamadas da função de avaliação é significativa.

É possível observar também que a acurácia e precisão crescem conforme mais caracteres são adicionados, com pouco impacto no número de avaliações feitas pelos algoritmos. Isso é esperado, uma vez que a maior quantidade de informações tende a ser benéfica para a análise, auxiliando a resolução de casos em que pode haver mais homoplasia (caracteres "não-

parcimoniosos"). Isso é condizente com as recomendações de Tschopp e Upchurch (2018) de incluir o máximo de caracteres possível na análise.

Tabela 5.2: Valores médios para matrizes de 32 táxons e 256 caracteres, com o melhor valor de cada coluna em negrito. As siglas são as mesmas da tabela 5.1.

		NNI				SPR			
		acur.	prec.	parc.	aval.	acur.	prec.	parc.	aval.
HC	estrito	39,1%	39,1%	732,5	$7,17 \times 10^7$	85,8%	**87,1%**	**591,8**	$\mathbf{7,16 \times 10^5}$
	bootstrap	14,8%	73,3%			85,8%	86,5%		
AG	estrito	**55,4%**	65,9%	**660,8**	$\mathbf{1,73 \times 10^7}$	74,4%	79,3%	604,4	$2,76 \times 10^6$
	bootstrap	53,3%	**84,3%**			**86,0%**	**87,1%**		

Tabela 5.3: Valores médios para matrizes de 32 táxons e 512 caracteres, com o melhor valor de cada coluna em negrito. As siglas são as mesmas da tabela 5.1.

		NNI				SPR			
		acur.	prec.	parc.	aval.	acur.	prec.	parc.	aval.
HC	estrito	53,4%	53,4%	1416,3	$7,14 \times 10^7$	**89,1%**	**89,4%**	**1237,8**	$\mathbf{7,09 \times 10^5}$
	bootstrap	**55,4%**	86,5%			89,0%	89,2%		
AG	estrito	54,0%	67,6%	**1361,8**	$\mathbf{1,72 \times 10^7}$	84,2%	84,6%	1251,5	$3,30 \times 10^6$
	bootstrap	49,9%	**87,6%**			**89,1%**	**89,4%**		

Tabela 5.4: Valores médios para matrizes de 32 táxons e 1024 caracteres, com o melhor valor de cada coluna em negrito. As siglas são as mesmas da tabela 5.1.

		NNI				SPR			
		acur.	prec.	parc.	aval.	acur.	prec.	parc.	aval.
HC	estrito	57,1%	57,1%	**2862,7**	$7,12 \times 10^7$	90,7%	91,0%	**2495,7**	$\mathbf{7,16 \times 10^5}$
	bootstrap	**84,7%**	**92,3%**			**90,8%**	91,0%		
AG	estrito	47,6%	54,6%	2950,3	$\mathbf{1,70 \times 10^7}$	80,6%	82,2%	2537,3	$3,52 \times 10^6$
	bootstrap	55,0%	90,7%			**90,8%**	**91,1%**		

Os dados médios referentes à matriz empírica de Hexapoda encontram-se na tabela 5.5. Essa apresenta o mesmo padrão dos casos anteriores, em que a combinação HC + SPR obteve os melhores resultados, mas com o AG utilizando melhor o NNI (num caso ainda mais gritante, pois o HC com *bootstrap* não foi capaz de resolver nenhuma bipartição). No caso do operador SPR, todas as métricas obtiveram p-valor no teste t abaixo de 0,05, exceto a precisão do consenso estrito ($p \approx 0,21$), com o valor para o *bootstrap* da acurácia em $\approx 0,048$.

Vale ressaltar que embora as diferenças entre os valores de parcimônia não sejam significativas ($p \approx 0,06$), o algoritmo *hill climbing* obteve o resultado "ótimo" (225) em todas as buscas, enquanto o AG "errou" em oito das 30 instâncias (seis com valor 226, uma com 227 e uma com 236). Ainda que a árvore de referência não seja a mais parcimoniosa (valor 248), a busca pelo algoritmo genético não atingiu otimalidade em todos os casos.

O uso do operador híbrido com os parâmetros fornecidos não se mostrou vantajoso, obtendo resultados próximos aos do operador SPR, com número de avaliações similares aos do NNI. Isso mostra que, ainda que o NNI corresponda a 80% das operações de mutação realizadas, o maior impacto no resultado ainda parte das mutações SPR. Os valores médios obtidos em cada conjunto de dados com o operador híbrido estão presentes na tabela 5.6.

Tabela 5.5: Valores para a matriz de Hexapoda, com o melhor valor de cada coluna em negrito. As siglas são as mesmas da tabela 5.1.

		NNI				SPR			
		acur.	prec.	parc.	aval.	acur.	prec.	parc.	aval.
HC	estrito	6,88%	7,78%	362,4	$6,60 \times 10^7$	59,2%	72,0%	**225,0**	$\mathbf{6,67 \times 10^5}$
	bootstrap	0%	0%			**60,4%**	71,5%		
AG	estrito	41,4%	61,4%	**236,0**	$\mathbf{2,05 \times 10^7}$	54,9%	70,8%	226,6	$8,84 \times 10^5$
	bootstrap	**46,7%**	**74,2%**			60,1%	**72,5%**		

Tabela 5.6: Valores para o operador de mutação híbrido do algoritmo genético em cada conjunto de dados. As siglas são as mesmas da tabela 5.1.

		acur.	prec.	parc.	aval.
16 táxons – 256 caracteres	estrito	91,6%	93,3%	296,4	$1,66 \times 10^7$
	bootstrap	91,7%	92,6%		
32 táxons – 256 caracteres	estrito	65,12%	73,1%	619,1	$1,87 \times 10^7$
	bootstrap	80,7%	86,9%		
32 táxons – 512 caracteres	estrito	63,1%	67,6%	1323,9	$1,83 \times 10^7$
	bootstrap	82,9%	89,5%		
32 táxons – 1024 caracteres	estrito	47,6%	54,6%	2950,3	$1,80 \times 10^7$
	bootstrap	86,6%	91,7%		
Hexapoda	estrito	54,3%	71,8%	227,7	$2,33 \times 10^7$
	bootstrap	61,2%	73,7%		

5.1 DISCUSSÃO

5.1.1 Sobre tempo e avaliações

Com os parâmetros definidos, fica evidente que a busca por *hill climbing* é capaz de obter respostas analisando menos árvores filogenéticas. Contudo, esse resultado não foi verificado no tempo de execução, de forma que os testes com HC eram notoriamente mais lentos que os realizados com o algoritmo genético. Nenhuma medição de tempo foi feita durante essa etapa para quantificar a diferença real.

Algumas razões podem justificar essa discrepância entre o tempo observado e a quantidade de chamadas da função de avaliação. A primeira é que, embora os cálculos referentes à parcimônia correspondam à maior parte do tempo de processamento, essa talvez não seja a melhor maneira de aferir diferenças entre tempos de execução, já que os algoritmos de busca envolvem outras etapas (como troca de ramos e mutações), que também podem tomar parcelas significativas da computação.

A segunda envolve questões de implementação. Ainda que tanto a medição da parcimônia quanto os algoritmos de busca compartilhem boa parte do código, suas implementações não são perfeitas. Como um exemplo, o *hill climbing* com o operador SPR precisa verificar topologias repetidas para garantir que todos os vizinhos foram analisados, aumentando tanto o tempo de execução quanto a quantidade de avaliações.

A terceira diz respeito a parâmetros para o GASPAR. As figuras 5.1, 5.2 e 5.3 mostram os gráficos da relação entre a média da parcimônia do melhor indivíduo da população em cada geração do algoritmo genético para todos os conjuntos de dados e operadores. É evidente que a exploração rapidamente converge para próximo do resultado ótimo em muito menos gerações do que o limite estabelecido de 20 mil sem alteração desse valor. Dessa forma, boa parte da

computação é usada para incrementos pequenos no valor de otimização, indicando que esse limiar poderia ser menor sem grandes perdas.

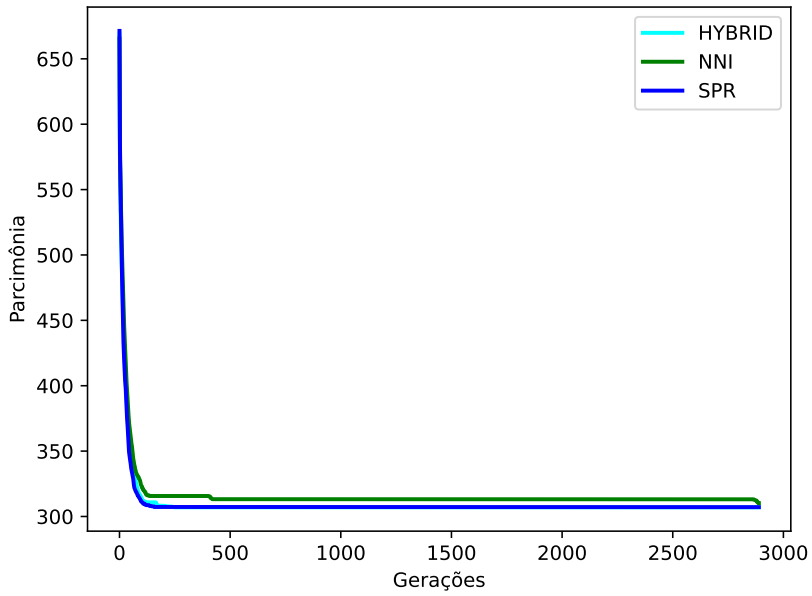


Figura 5.1: Média da parcimônia para cada geração do algoritmo genético no conjunto de 16 táxons. Últimas 20 mil gerações foram ocultadas.

5.1.2 Sobre acurácia e precisão

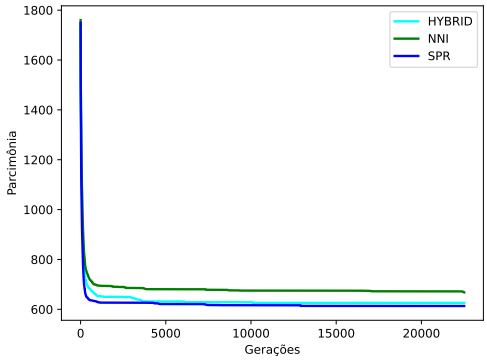
Tanto o *hill climbing* quanto o algoritmo genético, no melhor dos casos, são capazes de obter bons valores de acurácia e precisão. Em especial, esses tendem a um aumento quando é utilizado o consenso por regra da maioria com as topologias obtidas via *bootstrap*, cujo efeito é colapsar (remover) ramos com baixo suporte.

Em geral, isso significa tomar menos riscos na apresentação dos resultados, trocando certeza por uma árvore menos resolvida. A figura 5.4 mostra exemplos de filogenias geradas pelo GASPAR para o conjunto de Hexapoda, indicando em vermelho os ramos inferidos incorretamente (falsos positivos). Já a figura 5.5 mostra a árvore de referência com ramos coloridos representando bipartições faltantes nos exemplos anteriores (falsos negativos). Vale notar que o exemplo com *bootstrap* apresenta uma região particularmente menos resolvida, e inferiu uma bipartição correta a menos.

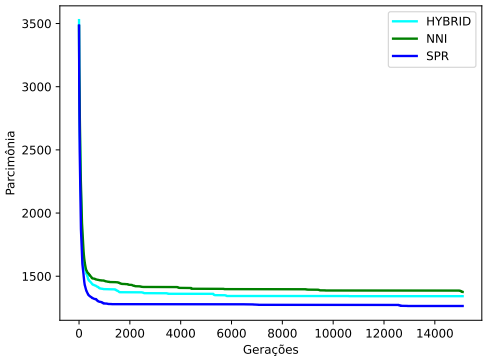
A máxima parcimônia é um método de otimização conservador (isto é, gera muitas politomias) e apresenta alta precisão, como mostrado por Schrago et al. (2018). A combinação do mesmo com a técnica de *bootstrap* garante ainda mais certeza do resultado, ao custo de menos resolução na resposta.

5.1.3 Sobre resultados variáveis

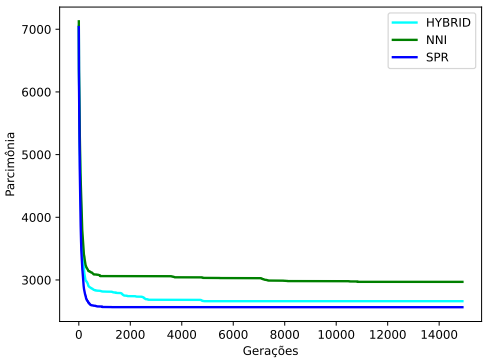
Uma consequência interessante do uso de algoritmos genéticos é uma maior variabilidade da resposta, como os testes no conjunto empírico evidenciam. Outras implementações de AGs voltados à filogenia se aproveitam dessa característica, a exemplo do GARLI (Zwickl, 2006)



(a) 256 caracteres.



(b) 512 caracteres.



(c) 1024 caracteres.

Figura 5.2: Média da parcimônia para cada geração do algoritmo genético nos conjuntos de 32 táxons. Últimas 20 mil gerações foram ocultadas.

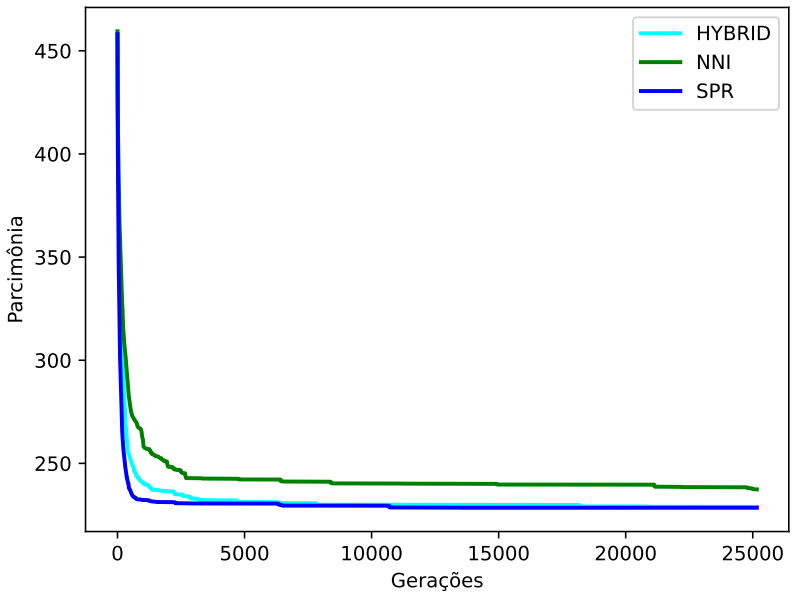
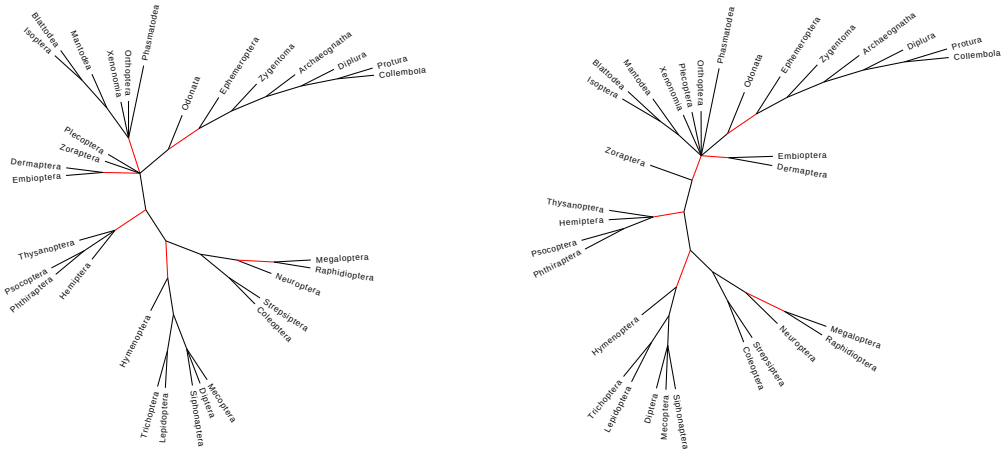


Figura 5.3: Média da parcimônia para cada geração do algoritmo genético no conjunto de Hexapoda. Últimas 20 mil gerações foram ocultadas.



(a) Consenso estrito.

(b) *Bootstrap* + consenso por regra da maioria.

Figura 5.4: Exemplos de árvores filogenéticas de consenso geradas pelo GASPAR para dados de Hexapoda. Em vermelho estão as bipartições inferidas incorretamente.

e do MetaPIGA (Lemmon e Milinkovitch, 2002), que utilizam várias réplicas e o paradigma multipopulação para atingir resultados ótimos.

Essa variabilidade no GASPAR dificulta a reprodutibilidade do resultado. Outros estudos evidenciam que, de fato, o processo de busca é aprimorado quando algoritmos genéticos são combinados com *hill climbing*, somando suas forças (Duvivier et al., 1996). O PARSIGAL de Moilanen (1999) adota essa alternativa.

Como evidenciado anteriormente, isso pode também se dar por uma escolha sub-ótima de parâmetros. Uma das principais dificuldades de lidar com algoritmos genéticos é a escolha adequada de parâmetros e operadores, de forma que mais testes se fazem necessários para determinar o potencial máximo do GASPAR.

5.2 CONCLUSÃO

Neste capítulo, foram apresentados os resultados dos testes com o GASPAR, junto a discussões sobre o significado e potenciais implicações dos mesmos. É possível verificar que o uso de algoritmos genéticos rende resultados bastante próximos aos da busca por *hill climbing*, ainda que com um pouco menos de certeza e realizando mais avaliações de parcimônia.

O próximo capítulo trata das considerações finais deste trabalho, levantando também pontos em que o mesmo pode ser aprimorado no futuro, bem como outros trabalhos que podem ser feitos a partir deste.

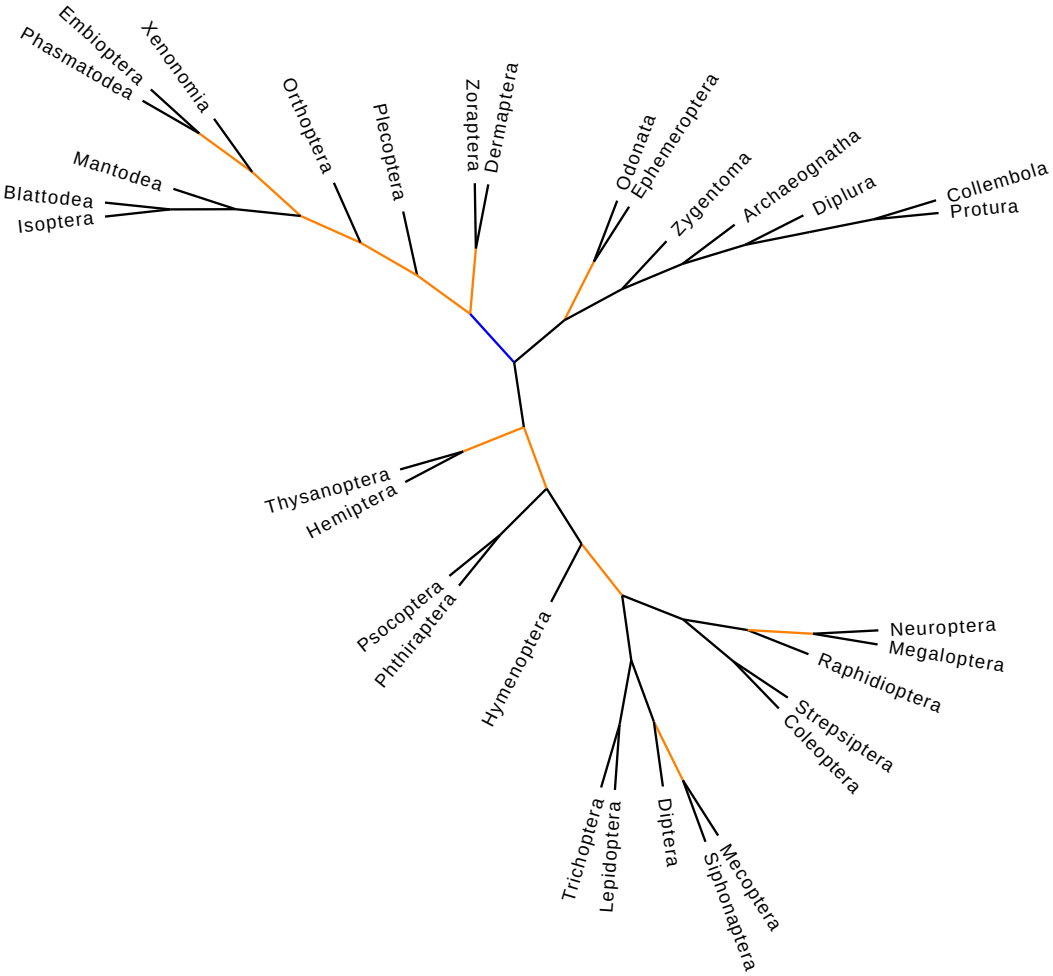


Figura 5.5: Árvore de referência para Hexapoda, conforme Barbosa et al. (2024). Em azul está a bipartição faltante na árvore gerada por regra da maioria sobre os dados de *bootstrap* e em laranja as bipartições faltantes nessa e no consenso estrito.

6 CONSIDERAÇÕES FINAIS

Ainda que os resultados mostrem que a busca por *hill climbing* supera levemente o uso do algoritmo genético, o mesmo se mostrou bastante promissor, e consegue chegar bem perto do desempenho alcançado pela técnica mais tradicional. Especialmente no exemplo empírico, ambos os algoritmos tiveram performances bastante similares sob o melhor operador (SPR). GAs também aparentam ser mais rápidos de executar e mais simples de implementar, ainda que dependam de mais computação "desperdiçada" para alcançar este resultado.

Todavia, seu uso não parece adequado à quantidade de táxons trabalhada, para a qual tanto HC quanto *branch and bound* se mostram mais apropriados. Vale ressaltar que o GARLI, principal inspiração do GASPAR, foi desenvolvido com foco em grandes volumes de dados, caso para o qual AGs e semelhantes parecem ser melhores.

Além do GARLI, a busca por *new technologies* implementada no programa TNT (Goloboff e Morales, 2023), pensada para matrizes grandes, utiliza diversos algoritmos (nominalmente: *tree fusing*, *sectorial searches* e *tree drifting*; Goloboff, 1999) cujo comportamento estocástico por vezes se assemelha ao de um algoritmo genético, indicando que essa seria a aplicação mais vantajosa dos mesmos.

Alguns artigos recentes em Paleontologia utilizaram em suas análises filogenéticas matrizes morfológicas com mais de 100 táxons, com resultados obtidos através de uma busca por *new technologies* seguida de *hill climbing* com o operador TBR (Pêgas, 2024; Fonseca et al., 2024). Ainda que pareça efetivo, nenhum dos trabalhos justifica o motivo desse fluxo ser o escolhido, mostrando uma lacuna no desenvolvimento de métodos adequados a essa quantidade de dados.

Em resumo, a aplicação de algoritmos genéticos (e em especial, do GASPAR) representa uma alternativa promissora aos métodos tradicionalmente usados na busca de árvores filogenéticas por morfologia, ainda que com possibilidade de melhora. Pelo menos para matrizes pequenas, não aparenta ser o mais adequado, e sugere-se sua aplicação para volumes de dados maiores.

6.1 MELHORIAS

Em um momento oportuno futuramente, novas funcionalinalidades podem ser implantadas para aprimorar o GASPAR. Atualmente, não há um mecanismo adequado para remoção de árvores duplicadas da resposta final, o que pode resultar em filogenias com topologias repetidas, sem que isso seja reportado durante a busca. Seria interessante também adicionar índices de consistência e retenção e ainda de cálculo de consenso, para que não seja necessário recorrer a um *software* externo.

Também em uma nova fase, pode-se tentar incluir uma estratégia de reprodutibilidade de saída. No momento, a alternativa é a definição de uma semente fixa para as etapas dependentes de algoritmos de geração de números pseudoaleatórios, mas ajustes nos parâmetros podem levar a um resultado ótimo em breve, a partir de mais testes e simulações.

6.2 TRABALHOS FUTUROS

Esta monografia não representa o final de desenvolvimento do GASPAR, mas é seu ponto de partida. Entre os possíveis temas de trabalhos que podem se originar a partir deste destacam-se:

- **Implementação paralela:** a população do AG atual é avaliada de modo sequencial, o que pode ser facilmente paralelizado a nível de *thread* para melhor desempenho. O mesmo pode ser feito para as réplicas do HC. O impacto do uso de instruções AVX2 no cálculo de parcimônia também não foi mensurado.
- **Outras funções de avaliação:** atualmente, o GASPAR está limitado a análises por máxima parcimônia, mas outros critérios podem ser adotados futuramente, como máxima parcimônia de pesos implícitos (Goloboff, 1993), índice de consistência estratigráfica (Huelsenbeck, 1994) e máxima verossimilhança. Também não é possível fazer análises com caracteres ordenados (aditivos).
- **Otimização multiobjetivo:** uma das principais vantagens de algoritmos genéticos é permitir a otimização por múltiplos critérios simultaneamente. A implementação de mais funções de avaliação permitiria a utilização de mais de uma ao mesmo tempo, aos moldes de programas como o MO-Phylogenetics (Zambrano-Vega et al., 2016).
- **Variação de parâmetros:** não foram feitos testes variando-se as configurações dos parâmetros do AG neste trabalho, concentrando-se principalmente no que está disponível na literatura. É possível que diferentes valores tenham impacto significativo nos resultados da busca.
- **Operadores de recombinação e mutação:** embora trabalhos correlatos apontem baixo impacto do operador de recombinação para análise filogenética, essa afirmação ainda não foi testada para o GASPAR. Outros operadores de mutação e recombinação, como os presentes no TNT também podem ser testados.

REFERÊNCIAS

Adams, III, E. N. (1972). Consensus techniques and the comparison of taxonomic trees. *Systematic Biology*, 21(4):390–397.

Barbosa, F. F., Mermudes, J. R. M. e Russo, C. A. (2024). Performance of tree-building methods using a morphological dataset and a well-supported Hexapoda phylogeny. *PeerJ*, 12:e16706.

Berger, S. A. e Stamatakis, A. (2010). Accuracy of morphology-based phylogenetic fossil placement under maximum likelihood. Em *ACS/IEEE International Conference on Computer Systems and Applications-AICCSA 2010*, páginas 1–9. IEEE.

Brauer, M. J., Holder, M. T., Dries, L. A., Zwickl, D. J., Lewis, P. O. e Hillis, D. M. (2002). Genetic algorithms and parallel processing in maximum-likelihood phylogeny inference. *Molecular Biology and Evolution*, 19(10):1717–1726.

Bremer, K. (1988). The limits of amino acid sequence data in angiosperm phylogenetic reconstruction. *Evolution*, 42(4):795–803.

Cavalli-Sforza, L. L. e Edwards, A. W. (1967). Phylogenetic analysis. models and estimation procedures. *American journal of human genetics*, 19(3 Pt 1):233.

Chor, B. e Tuller, T. (2005). Maximum likelihood of evolutionary trees: hardness and approximation. *Bioinformatics*, 21(suppl_1):i97–i106.

Colless, D. (1980). Congruence between morphometric and allozyme data for *Menidia* species: a reappraisal. *Systematic Zoology*, 29(3):288–299.

Congdon, C. B. e Greenfest, E. F. (2001). Gaphyl: A genetic algorithms approach to cladistics. Em *European Conference on Principles of Data Mining and Knowledge Discovery*, páginas 67–78. Springer.

Cotta, C. e Moscato, P. (2002). Inferring phylogenetic trees using evolutionary algorithms. Em *International Conference on Parallel Problem Solving from Nature*, páginas 720–729. Springer.

Day, W. H., Johnson, D. S. e Sankoff, D. (1986). The computational complexity of inferring rooted phylogenies by parsimony. *Mathematical biosciences*, 81(1):33–42.

Duvivier, D., Preux, P. e Talbi, E.-G. (1996). Climbing up NP-hard hills. Em *International Conference on Parallel Problem Solving from Nature*, páginas 574–583. Springer.

Edwards, A. W. e Cavalli-Sforza, L. L. (1963). The reconstruction of evolution. Em *Annals of Human Genetics*, volume 27, páginas 105–106.

Farris, J. S., Albert, V. A., Källersjö, M., Lipscomb, D. e Kluge, A. G. (1996). Parsimony jackknifing outperforms neighbor-joining. *Cladistics*, 12(2):99–124.

Felsenstein, J. (1973). Maximum likelihood and minimum-steps methods for estimating evolutionary trees from data on discrete characters. *Systematic Biology*, 22(3):240–249.

Felsenstein, J. (1981). Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of molecular evolution*, 17:368–376.

Felsenstein, J. (1985). Confidence limits on phylogenies: an approach using the bootstrap. *Evolution*, 39(4):783–791.

Felsenstein, J. (2004). *Inferring phylogenies*. Sinauer Associates, Inc.

Felsenstein, J. (2005). PHYLIP (Phylogeny Inference Package) version 3.6. `https://phylipweb.github.io/phylip/`. Distribuído pelo autor. Department of Genome Sciences, University of Washington, Seattle. Acessado em 22/07/2024.

Fitch, W. M. (1971). Toward defining the course of evolution: minimum change for a specific tree topology. *Systematic Biology*, 20(4):406–416.

Fitch, W. M. e Margoliash, E. (1967). Construction of phylogenetic trees: a method based on mutation distances as estimated from cytochrome c sequences is of general applicability. *Science*, 155(3760):279–284.

Fonseca, A. O., Reid, I. J., Venner, A., Duncan, R. J., Garcia, M. S. e Müller, R. T. (2024). A comprehensive phylogenetic analysis on early ornithischian evolution. *Journal of Systematic Palaeontology*, 22(1):2346577.

Giribet, G. (2015). Morphology should not be forgotten in the era of genomics–a phylogenetic perspective. *Zoologischer Anzeiger-A Journal of Comparative Zoology*, 256:96–103.

Goloboff, P. A. (1993). Estimating character weights during tree search. *Cladistics*, 9(1):83–91.

Goloboff, P. A. (1999). Analyzing large data sets in reasonable times: solutions for composite optima. *Cladistics*, 15(4):415–428.

Goloboff, P. A. e Morales, M. E. (2023). TNT version 1.6, with a graphical interface for MacOS and Linux, including new routines in parallel. *Cladistics*, 39(2):144–153.

Goloboff, P. A., Torres, A. e Arias, J. S. (2018). Weighted parsimony outperforms other methods of phylogenetic inference under models appropriate for morphology. *Cladistics*, 34(4):407–437.

Hendy, M. D. e Penny, D. (1982). Branch and bound algorithms to determine minimal evolutionary trees. *Mathematical Biosciences*, 59(2):277–290.

Hennig, W. (1965). Phylogenetic systematics. *Annual review of entomology*, 10(1):97–116.

Holland, J. H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence.* U Michigan Press.

Huelsenbeck, J. P. (1994). Comparing the stratigraphic record to estimates of phylogeny. *Paleobiology*, 20(4):470–483.

Huelsenbeck, J. P. (1995). Performance of phylogenetic methods in simulation. *Systematic biology*, 44(1):17–48.

Huerta-Cepas, J., Serra, F. e Bork, P. (2016). ETE 3: reconstruction, analysis, and visualization of phylogenomic data. *Molecular biology and evolution*, 33(6):1635–1638.

Keating, J. N., Sansom, R. S., Sutton, M. D., Knight, C. G. e Garwood, R. J. (2020). Morphological phylogenetics evaluated using novel evolutionary simulations. *Systematic Biology*, 69(5):897–912.

Kidd, K. K. e Sgaramella-Zonta, L. A. (1971). Phylogenetic analysis: concepts and methods. *American journal of human genetics*, 23(3):235.

Lee, M. S. e Palci, A. (2015). Morphological phylogenetics in the genomic age. *Current Biology*, 25(19):R922–R929.

Lemmon, A. R. e Milinkovitch, M. C. (2002). The metapopulation genetic algorithm: An efficient solution for the problem of large phylogeny estimation. *Proceedings of the National Academy of Sciences*, 99(16):10516–10521.

Maddison, W. (1989). Reconstructing character evolution on polytomous cladograms. *Cladistics*, 5(4):365–377.

Margush, T. e McMorris, F. R. (1981). Consensus*n*-trees. *Bulletin of Mathematical Biology*, 43(2):239–244.

Moilanen, A. (1999). Searching for most parsimonious trees with simulated evolutionary optimization. *Cladistics*, 15(1):39–50.

Mongiardino Koch, N., Garwood, R. J. e Parry, L. A. (2021). Fossils improve phylogenetic analyses of morphological characters. *Proceedings of the Royal Society B*, 288(1950):20210044.

Pattengale, N. D., Alipour, M., Bininda-Emonds, O. R., Moret, B. M. e Stamatakis, A. (2010). How many bootstrap replicates are necessary? *Journal of computational biology*, 17(3):337–354.

Pêgas, R. V. (2024). A taxonomic note on the tapejarid pterosaurs from the Pterosaur Graveyard site (Caiuá Group,? Early Cretaceous of Southern Brazil): evidence for the presence of two species. *Historical Biology*, páginas 1–22.

Rannala, B. e Yang, Z. (1996). Probability distribution of molecular evolutionary trees: a new method of phylogenetic inference. *Journal of molecular evolution*, 43:304–311.

Rieger, H. (2024). GASPAR – genetic algorithm for searches under parsimony. `https://github.com/henrieger/gaspar`. Acessado em 26/07/2024.

Robinson, D. F. e Foulds, L. R. (1981). Comparison of phylogenetic trees. *Mathematical biosciences*, 53(1-2):131–147.

Saitou, N. e Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular biology and evolution*, 4(4):406–425.

Schrago, C. G., Aguiar, B. O. e Mello, B. (2018). Comparative evaluation of maximum parsimony and bayesian phylogenetic reconstruction using empirical morphological data. *Journal of evolutionary biology*, 31(10):1477–1484.

Sneath, P. H. e Sokal, R. R. (1962). Numerical taxonomy. *Nature*, 193:855–860.

Tschopp, E. e Upchurch, P. (2018). The challenges and potential utility of phenotypic specimen-level phylogeny based on maximum parsimony. *Earth and Environmental Science Transactions of the Royal Society of Edinburgh*, 109(1-2):301–323.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P. e colaboradores do SciPy 1.0 (2020). SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272.

Zambrano-Vega, C., Nebro, A. J. e Aldana-Montes, J. F. (2016). MO-Phylogenetics: a phylogenetic inference software tool with multi-objective evolutionary metaheuristics. *Methods in Ecology and Evolution*, 7(7):800–805.

Zwickl, D. J. (2006). *Genetic algorithm approaches for the phylogenetic analysis of large biological sequence datasets under the maximum likelihood criterion*. Tese de doutorado, The University of Texas at Austin. 125 pgs.