Um Consenso Completamente Resolvido entre Árvores Filogenéticas Completamente Resolvidas

José Augusto Amgarten Quitzau

Dissertação de Mestrado

Instituto de Computação
Universidade Estadual de Campinas

Um Consenso Completamente Resolvido entre Árvores Filogenéticas Completamente Resolvidas

José Augusto Amgarten Quitzau

Janeiro de 2005

Banca Examinadora:

- Prof. Dr. João Meidanis (Orientador)
- Profa. Dra. Estela Maris Rodrigues
  Insituto de Ciências Matemáticas e de Computação - USP
- Prof. Dr. Zanoni Dias
  Instituto de Computação - UNICAMP
- Prof. Dr. Jorge Stolfi (Suplente)
  Instituto de Computação - UNICAMP

Um Consenso Completamente Resolvido entre Árvores Filogenéticas Completamente Resolvidas

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por José Augusto Amgarten Quitzau e aprovada pela Banca Examinadora.

Campinas, 24 de fevereiro de 2005.

Prof. Dr. João Meidanis (Orientador)

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

© José Augusto Amgarten Quitzau, 2007.
Todos os direitos reservados.

Sumário

A relação evolutiva entre espécies de seres vivos é normalmente representada através de um diagrama conhecido como *árvore filogenética*. Embora existam inúmeros métodos de construção deste tipo de diagrama, com base nos mais variados tipos de dados biológicos, nenhum deles garante obter a árvore que melhor representa a relação evolutiva entre as espécies de um conjunto. Alguns métodos ainda podem produzir várias árvores distintas para um mesmo conjunto, sendo incapazes de decidir qual a que melhor explica a relação entre os seres representados. Cabe então aos biólogos comparar as árvores e decidir qual a melhor entre elas.

Uma maneira de estudar semelhanças entre árvores construídas sobre um mesmo conjunto de espécies é a utilização de um *consenso* entre as árvores. Atualmente existem diversos métodos de consenso, cada um enfatizando características diferentes do conjunto de árvores.

Esta dissertação destaca a possibilidade de uso de métodos de consenso como métodos de construção, apresentando um teste bastante simples que ressalta a qualidade das árvores consenso em relação a árvores criadas pelos métodos tradicionais de construção de árvores filogenéticas. Além disso, apresenta um novo método de consenso focado não na comparação de árvores, mas na construção de uma árvore filogenética capaz de se aproximar mais da árvore correta do que a maior parte das árvores presentes no conjunto utilizado para construí-la.

Abstract

The evolutionary relationship between species is usually represented by a diagram known as *phylogenetic tree*. Despite of the existence of a huge number of different methods for building such a diagram, based on the most diverse sorts of biological data, none of these methods guarantees that the reconstructed tree is the tree which represents better the evolutionary relationship between the species in a given set. Some of the methods may even build different trees for the same input set of species. In these cases, they are unable to decide which of the trees represents better the relationship between the considered beings. In such cases, the task of comparing the produced trees and choosing the best one is left to biologists.

One way to study the similarity of phylogenetic trees is to build a *consensus* between them. Nowadays there are different consensus methods, each of them exploring a different characteristic of the set of trees.

This work focus on the possibility of use of consensus methods as reconstruction methods, presenting a very simple test, which points out the quality of consensus trees compared to trees built by traditional phylogeny reconstruction methods. After this, we present a consensus method dedicated to reconstructing trees, instead of just comparing them. We also show that trees built by this method are usually closer to the true tree than most trees used to build them.

Agradecimentos

Aos meus pais e minhas irmãs pelo carinho e apoio dado não só no mestrado, mas ao longo de toda a vida, e ao professor João Meidanis por ter me apresentado a área de bioinformática e, um ano depois, ter me aceito como orientando. Sem o apoio destas pessoas este trabalho certamente jamais teria sido concluído.

A todos aqueles que já foram meus professores, sem exceção. Tenho plena consciência de que este trabalho é o resultado da contribuição de cada um deles na minha formação. Em especial, gostaria de agradecer à Eliana, de Matemática, à Rose, de Biologia, e ao Chico, de Física e Informática, que despertaram o meu interesse pelas três áreas principais que suportam a bioinformática.

Ao pessoal com quem tive mais contato no LBI durante o período em que trabalhei lá. Em especial, Guilherme, Lin, Marcelo (MC), Marília, Patrícia e Zanoni, pelo ambiente descontraído e pelos jogos de stop durante as paradas de energia no IC.

A Augusto, Camila e Daniele por convites insanos feitos por telefone ou ICQ do tipo: “Zeppelin hoje” ou “Swingers na quarta, topa?”. Os convites normalmente coincidiam exatamente com os momentos em que eu precisaria estar mais compenetrado no trabalho e jamais foram recusados. Um ótimo remédio para manter a sanidade mental em certas horas.

E por falar em manter a sanidade mental, gostaria de agradecer a Flávia, Miguel e Sorô pelas idas repentinas à Universitária e pelas cervejadas de última hora na casa menos mobiliada de Barão Geraldo.

Ao Fabiano, por todas as vezes que me perguntou “E a tese?”.

A Bach por suas cantatas e Mendelssohn por sua sinfonia número 2, que me fizeram relaxar quando o trabalho estava estressante demais. E ao pessoal do Shaman, pelo CD Ritual e a Tobias Sammet por Avantasia, que me mantiveram acordado quando Bach e Mendelssohn me relaxaram demais.

Por último, mas não menos importante, a Guilherme, Fernando, Schubert, Milene e Sorô, meus companheiros de república, que não deixaram meu humor cair em momento algum, ao mesmo tempo em que estavam sempre esperando o menor dos deslizes para rir da minha cara.

Conteúdo

Lista de Tabelas

Lista de Figuras

Capítulo 1

Introdução

> The field of systematics has been in considerable turmoil as various investigators developed different methods of classification and argued their merits. I guarantee you that no one method or view has all the good points.
>
> Walter M. Fitch (1984)

Há bem mais de um século, um tipo de diagrama vem sendo utilizado para representar tanto a classificação quanto a evolução do mundo vivo. A construção de diagramas deste tipo, conhecidos como árvores filogenéticas, entre outros nomes, é um dos problemas para o qual a biologia computacional volta sua atenção já há um bom tempo. O principal desafio nesta área é construir o diagrama que representa com fidelidade absoluta a história evolutiva de um conjunto de seres vivos. Um grande passo nesta direção foi dado por Willi Hennig, um biólogo alemão que propôs, na década de 1960, que a classificação dos seres vivos deveria refletir sua história evolutiva, o que implica diretamente no fato de que árvores representando a classificação de seres vivos e árvores representando sua história evolutiva devem ser equivalentes. Hennig usou sua proposição para desenvolver o primeiro método explícito de construção de árvores filogenéticas, chamado de *argumentação de Hennig*, ou *Hennigian argumentation* [14] em inglês.

A proposta de Hennig é talvez o único caso de convergência de dois diagramas distintos aceito pela grande maioria dos biólogos. O que se tem assistido desde a argumentação de Hennig é o surgimento de inúmeros métodos de construção de filogenias, cada um explorando o seu conjunto de hipóteses sobre o processo evolutivo. Neste contexto, a frase extraída de um artigo de Fitch [10] que abre este capítulo é mais que pertinente. De fato, dentre os inúmeros métodos de construção de árvores filogenéticas, nenhum pode dizer que contrói a árvore original com certeza absoluta e cada cientista que adota um dos métodos é capaz de encontrar tantos argumentos em favor de seu método favorito quanto os demais cientistas são capazes de encontrar argumentos contra ele.

No meio desta babel de métodos de construção, alguns detalhes acabam passando despercebidos. Talvez o detalhe mais importante seja a existência de métodos para a combinação de várias árvores em uma única. Este detalhe não é meramente uma curiosidade, pois se temos em mãos vários métodos de construção de árvores, cada um fornecendo como saída uma árvore diferente, apesar de garantidamente próxima da árvore correta, faz sentido pensar que as diferenças entre as árvores são causadas pela imperfeição dos métodos usados para construi-las, o que indica que a combinação das partes comuns à maioria das árvores poderia dar origem a uma árvore tão boa ou melhor que as árvores do conjunto. Isso porque a árvore seria construída com "pedaços" das árvores que teriam mais chances de estarem corretos.

Esta dissertação tem como objetivo investigar a utilidade de métodos de consenso entre árvores filogenéticas como meio de construção de árvores mais precisas a partir de um conjunto inicial de árvores fornecidas por diversos métodos de construção e propor um consenso completamente resolvido entre árvores filogenéticas completamente resolvidas.

Todas as provas apresentadas neste texto foram desenvolvidas ao longo deste projeto, embora a maioria dos resultados principais já fosse conhecida, podendo ser encontrada na literatura. Destacamos os resultados relativos à escolha canônica de um dos subgrupos de um corte, como a definição de subgrupo pequeno, e todos os resultados relativos ao algoritmo e sua análise, que são, até onde sabemos, resultados originais.

Até o final deste capítulo serão tratados o uso de árvores filogenéticas como forma de representar classificações de seres vivos e sua história evolutiva. O Capítulo 2 introduz os conceitos de árvore filogenética e árvore filogenética completamente resolvida, além de apresentar duas formas de representar árvores filogenéticas baseadas em subconjuntos de seus conjuntos de folhas. O Capítulo 3 apresenta uma visão do estado da arte da área de consensos entre árvores filogenéticas, descrevendo brevemente alguns dos métodos mais conhecidos de consenso. O Capítulo 4 apresenta de forma sucinta alguns modelos de evolução e maneiras de usá-los para simular a evolução de moléculas de DNA, que no fundo está por trás de toda a evolução dos seres vivos. O Capítulo 5 apresenta os resultados dos testes que verificaram a utilidade de consensos entre árvores filogenéticas como métodos de construção de árvores filogenéticas. O Capítulo 6 é dedicado à apresentação de uma nova definição de consenso entre árvores filogenéticas, assim como um algoritmo para determinar este consenso dada uma coleção de árvores. Finalmente o Capítulo 7 apresenta um resumo das conclusões apresentadas durante a dissertação.

Grupo	**Elementos**
Celular organism	{*Escherichia coli*, *Zea mays*, *Oryza sativa*, *Arabidopsis thaliana*, *Saccharomyces cerevisiae*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Takifugu rubripes*, *Danio rerio*, *Mus musculus*, *Bos taurus*, *Homo sapiens*}
Eukaryota	{*Zea mays*, *Oryza sativa*, *Arabidopsis thaliana*, *Saccharomyces cerevisiae*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Takifugu rubripes*, *Danio rerio*, *Mus musculus*, *Bos taurus*, *Homo sapiens*}
Fungi/metazoa group	{*Saccharomyces cerevisiae*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Takifugu rubripes*, *Danio rerio*, *Mus musculus*, *Bos taurus*, *Homo sapiens*}
Bilateria	{*Caenorhabditis elegans*, *Drosophila melanogaster*, *Takifugu rubripes*, *Danio rerio*, *Mus musculus*, *Bos taurus*, *Homo sapiens*}
Coelomata	{*Drosophila melanogaster*, *Takifugu rubripes*, *Danio rerio*, *Mus musculus*, *Bos taurus*, *Homo sapiens*}
Euteleostomi	{*Takifugu rubripes*, *Danio rerio*, *Mus musculus*, *Bos taurus*, *Homo sapiens*}
Magnoliophyta	{*Zea mays*, *Oryza sativa*, *Arabidopsis thaliana*}
Eutheria	{*Mus musculus*, *Bos taurus*, *Homo sapiens*}
Poaceae	{*Zea mays*, *Oryza sativa*}
Clupeocephala	{*Takifugu rubripes*, *Danio rerio*}

Tabela 1.1: Uma família laminar definida sobre o conjunto {*Escherichia coli*, *Zea mays*, *Oryza sativa*, *Arabidopsis thaliana*, *Saccharomyces cerevisiae*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Takifugu rubripes*, *Danio rerio*, *Mus musculus*, *Bos taurus*, *Homo sapiens*}

1.1 Classificações do Mundo Vivo e Árvores Filogenéticas

O sistema de classificação do mundo vivo utilizado atualmente é baseado no sistema proposto por Carolus Linneaus (1707 – 1778) em seu trabalho *Systema naturae*. O sistema de Linneaus agrupa os seres vivos em diversas classes e permite que os elementos destas classes sejam agrupados em subclasses. Desta forma, são criadas classes de seres vivos organizadas de uma maneira hierárquica, o que confere a todo o sistema uma característica muito especial: se tomarmos duas classes quaisquer, mesmo que de níveis hierárquicos distintos, então ou uma das classes contém todos os seres vivos da outra, ou não há nenhum ser vivo comum a ambas. Usando uma linguagem um pouco mais técnica, o conjunto dos grupos do sistema de classificação Linneaus é o que chamamos uma *família laminar* e este fato dá a ele propriedades que serão exploradas no Capítulo 2.

Dentre as propriedades garantidas pelo fato de o sistema de Linneaus ser uma família laminar, talvez a mais notável seja a de que é possível representá-lo na forma de diagramas

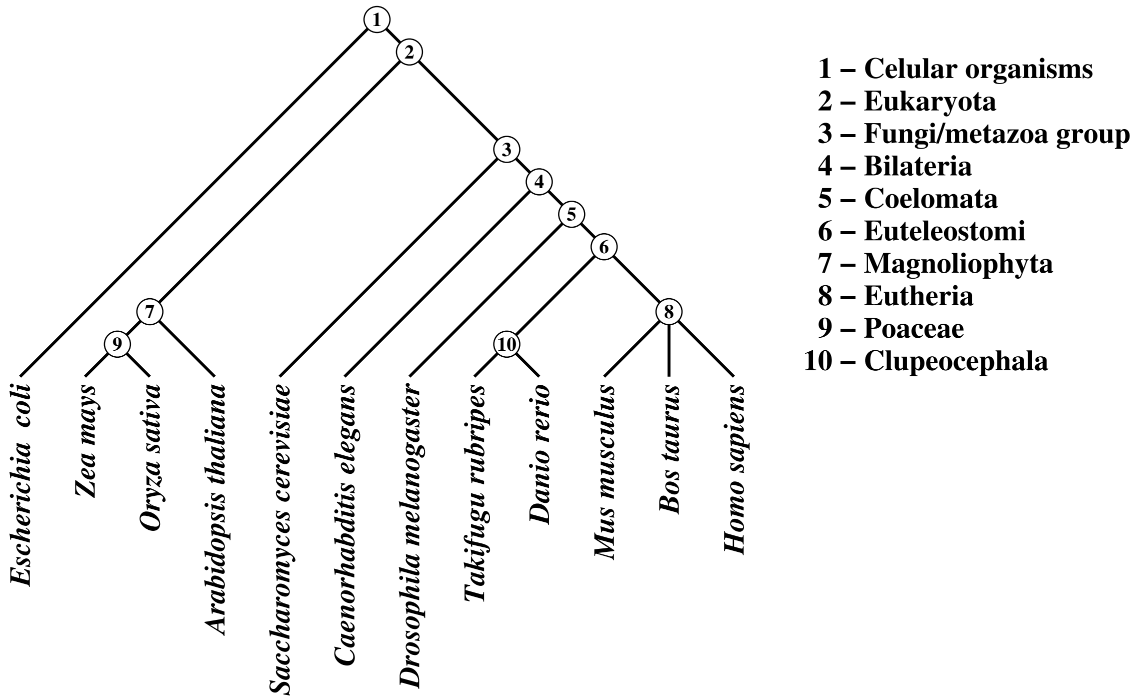

Figura 1.1: Diagrama representando a família laminar apresentada na Tabela 1.1.

como o apresentado na Figura 1.1. Nesta figura há dez grupos, todos subconjuntos do conjunto {*Escherichia coli*, *Zea mays*, *Oryza sativa*, *Arabidopsis thaliana*, *Saccharomyces cerevisiae*, *Caenorhabditis elegans*, *Drosophila melanogaster*, *Takifugu rubripes*, *Danio rerio*, *Mus musculus*, *Bos taurus*, *Homo sapiens*}. Os dez subconjuntos são apresentados na Tabela 1.1. Tanto a Figura 1.1 quanto os dados da Tabela 1.1 foram extraídos do diagrama criado pelo browser de taxonomia do NCBI [29]. Com um pouco de paciência, é possível verificar que a propriedade citada acima é verificada para todos os 45 pares de subconjuntos possíveis.

Cada grupo do sistema de Linneaus divide o conjunto de seres vivos em dois: o dos seres vivos que pertencem ao grupo e o dos seres vivos que não pertencem, ou seja, eles dividem o conjunto de seres vivos formando um par de subconjuntos. Chamaremos cada par de subconjuntos criados desta forma de um *corte* do conjunto de seres vivos. Propriedades dos cortes de um conjunto de seres vivos também serão exploradas no Capítulo 2.

1.2 Evolução e Árvores Filogenéticas

Quando falamos em evolução, é quase inevitável que venha à mente o nome de Charles Darwin. Embora a teoria de Darwin sobre a origem das espécies por meio de seleção

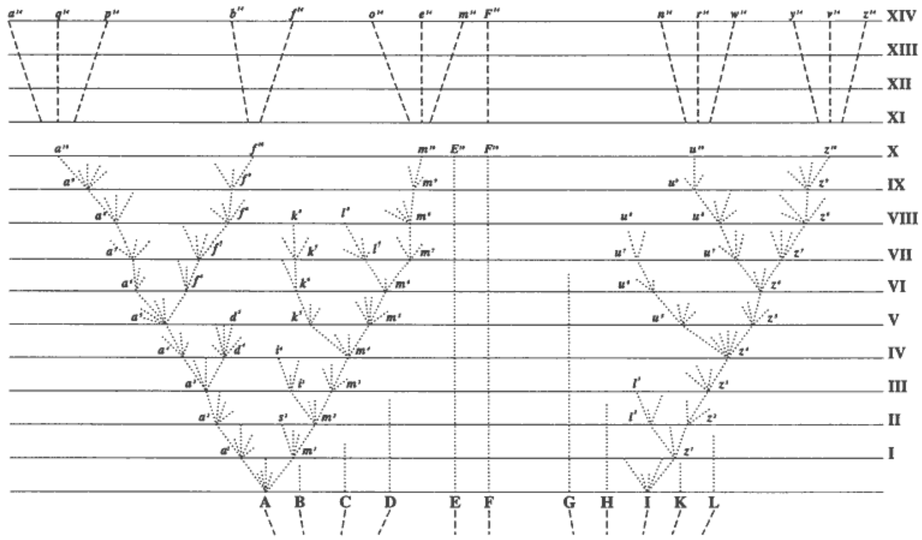

Figura 1.2: Diagrama usado por Charles Darwin no Capítulo 4 de seu livro "On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life" [6]. Para ilustrar o surgimento de novas espécies por variações de espécies pré-existentes, Darwin recorre a um diagrama formado por uma coleção de árvores filogenéticas.

natural não seja a única que tenta explicar o surgimento das espécies do mundo moderno a partir da variação de espécies ancestrais, ela é a mais aceita e, talvez por isso, a mais famosa de todas elas. De fato, embora não seja aceita com unanimidade, mesmo por motivos religiosos, é muito difícil encontrar uma pessoa que, mesmo nunca tendo ouvido falar de Darwin, não tenha ouvido nada a respeito de uma teoria que propõe que "o homem é descendente do macaco". Darwin apresentou formalmente sua teoria ao mundo em 1859 no livro *"On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life"*. Ele começa a expor suas idéias levantando várias situações em que a seleção de animais domésticos feita pelo ser humano acabou por levar a variedades extremamente diferentes da mesma espécie e, no Capítulo 4, introduz a teoria de que o mesmo tipo de seleção poderia ocorrer de forma natural, usando um exemplo ilustrado pelo diagrama na Figura 1.2, que nada mais é que uma coleção de árvores filogenéticas.

Quando comparamos o diagrama de Darwin com o apresentado na Figura 1.1, vemos que a semelhança entre os dois diagramas é grande o suficiente para que nos questionemos se se trata de mera coincidência. Para ver que de fato não é, devemos lembrar que os seres vivos são classificados de acordo com características comuns, como a presença ou ausência de uma certa estrutura, o modo como obtém energia, como se reproduzem, etc.

Tomemos agora o grupo *eukaryota* da Figura 1.1 como exemplo. Para pertencer a este grupo, um ser vivo precisa ter um núcleo celular bem definido, o que é de fato o caso de todos os seres vivos neste exemplo, com exceção da bactéria *Escherichia coli*. O grupo *eukaryota* cria então uma separação no mundo vivo entre os seres que possuem um núcleo celular bem definido e os que não possuem. Parece razoável que se considere que seres com uma estrutura complexa como um núcleo celular tenham surgido depois dos seres que não possuem tal estrutura. Também parece razoável pensar que todas as espécies que possuem esta estrutura tenham surgido da variação de uma única espécie, que teria sido a primeira espécie na face da Terra a possuir um núcleo celular bem definido, uma vez que é bem pouco provável que duas espécies distintas desenvolvessem paralelamente uma estrutura tão complexa. O vértice 2 da Figura 1.1 pode assim representar não somente a classe *eukaryota*, mas a espécie ancestral comum a todos os seres vivos desta classe.

Embora Linneaus tenha sugerido sua sistematização do mundo vivo em 1735 e Darwin tenha publicado sua teoria em 1859 e embora já no século XIX tenham surgido representações que se situam entre classificação e origem das espécies, como a que vemos na Figura 1.3, a relação entre os integrantes de um grupo de Linneaus e os descendentes de um ancestral comum, que agora nos parece tão clara, só foi sugerida por volta da década de 1960 pelo biólogo alemão Willi Hennig. Como dito no início deste texto, Hennig sugeriu que toda classificação de um grupo de seres vivos deve refletir a história evolutiva do grupo. Ele observou que, se considerarmos um grupo de seres vivos, veremos que algumas características apresentam-se de forma variada entre os seres vivos considerados e que somente uma destas variedades pode ser a forma verificada no ancestral comum a todos os seres do grupo. Baseado nesta constatação, Hennig propôs o primeiro método de construção de árvores evolutivas, conhecido como *argumentação de Hennig* ou *Hennigian argumentation*, que consiste em determinar, para cada característica considerada para um grupo de seres vivos, qual a variedade primitiva (ou plesiomórfica) e quais as variedades derivadas (ou apomórficas) [22]. Os grupos determinados pelas variedades derivadas devem ter as mesmas propriedades que os subgrupos do sistema de Linneaus e podem, portanto, ser usados para construir diagramas como os das Figuras 1.1, 1.2 e 1.3.

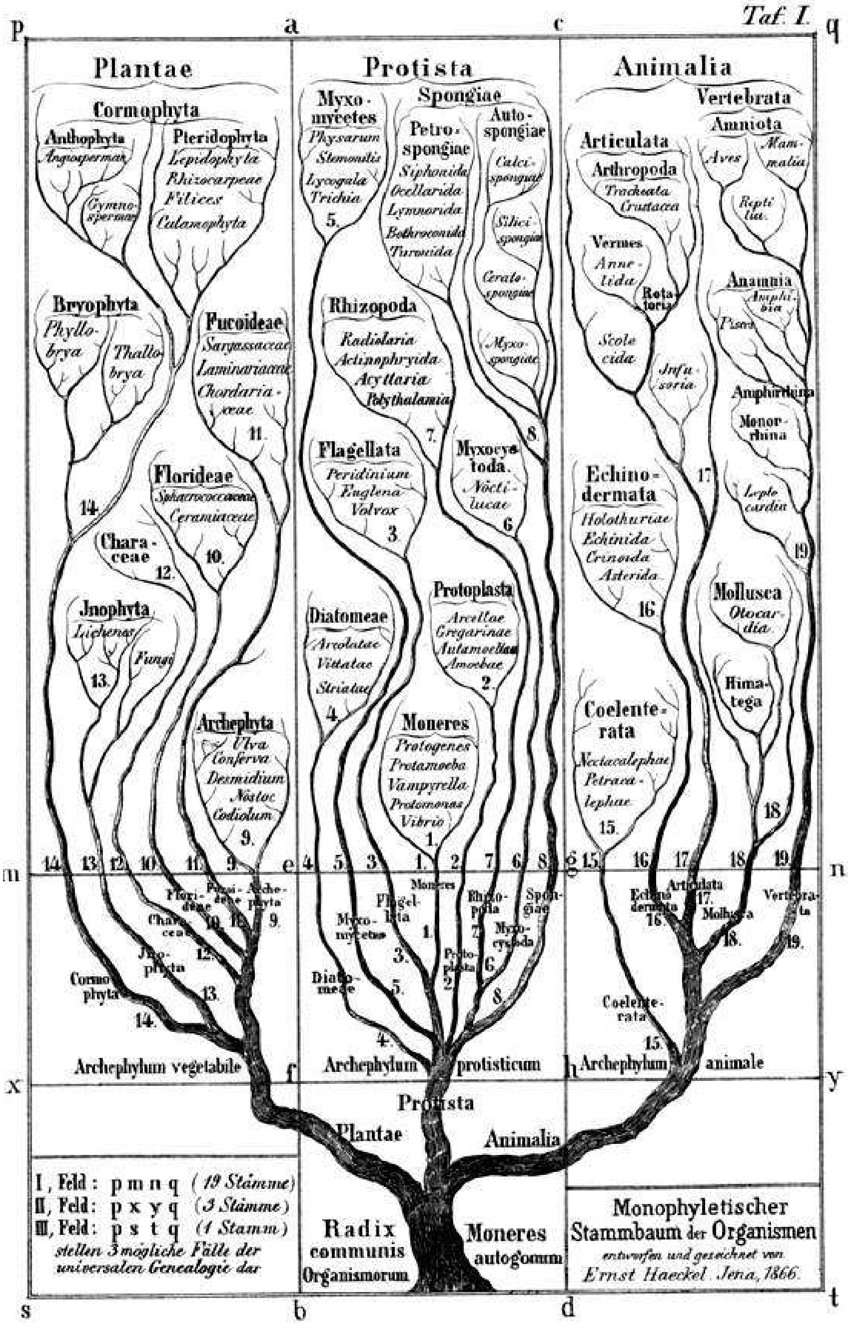

Figura 1.3: Diagrama usado pelo biólogo Ernst Haeckel em seu estudo "Monophyletischer Stammbaum der Organismen", feito em 1866 na cidade de Jena, Alemanha. Apesar de dar ênfase às classes de seres vivos, não incluindo explicitamente nenhuma espécie, o diagrama de Haeckel não tem a intensão de representar a classificação do mundo vivo, mas sim a origem das espécies.

Capítulo 2

Grafos, Cortes, Subgrupos e Árvores Filogenéticas

Uma *árvore filogenética* é um grafo não orientado, conexo e acíclico com no máximo um vértice de grau dois. Os vértices de grau um de uma árvore filogenética são chamados de *folhas* e representam seres vivos. Todos os demais vértices de uma árvore filogenética são denominados *nós internos* e representam espécies ancestrais das espécies representadas pelas folhas. Um dos nós internos pode ser identificado como o ancestral mais antigo de todo o conjunto das folhas da árvore. Neste caso, este nó é chamado de *raiz* da árvore e a árvore é uma *árvore filogenética com raiz*. Uma árvore filogenética com um vértice de graus dois é sempre uma árvore filogenética com raiz e a raiz é o vértice de grau dois. As arestas de uma árvore filogenética, por sua vez, podem ser diferenciadas entre *arestas folha*, que são as arestas adjacentes a uma folha, e *arestas internas*, que são todas as demais.

Um vértice de grau maior ou igual a quatro é chamado de uma *politomia* [18]. Em especial, a raiz de uma árvore filogenética é considerada uma politomia se tiver grau maior ou igual a três. Uma árvore sem politomias é chamada de uma *árvore filogenética completamente resolvida.* Uma árvore filogenética com politomias é denominada *parcialmente resolvida.* A Figura 2.1 mostra um exemplo de árvore filogenética parcialmente resolvida com raiz.

As folhas de uma árvore filogenética são rotuladas por elementos de um conjunto L de espécies. A relação entre folhas e elementos de L é dada por uma função bijetora, ou seja, toda folha está associada a um rótulo e cada rótulo está associado a uma única folha. Denotamos por $\mathcal{T}(L)$ o conjunto de todas as árvores filogenéticas cujas folhas são rotuladas por elementos de L. Devido a esta relação tão estreita dos elementos de L com as folhas de uma árvore filogenética, para evitar complicações desnecessárias neste texto, não faremos distinção entre os elementos de L e as folhas que eles rotulam. Ao longo

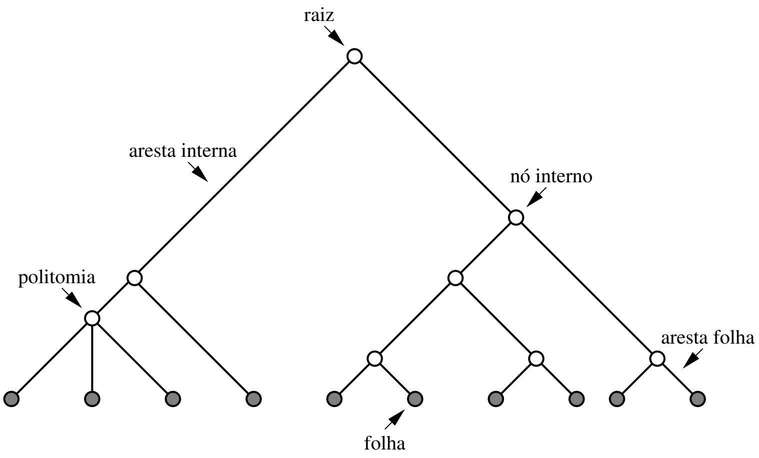

Figura 2.1: Um exemplo de árvore filogenética parcialmente resolvida com raiz, indicando alguns dos vértices e arestas especiais.

de todo o texto, os elementos de L serão considerados as folhas das árvores estudadas. Portanto, para qualquer árvore T em $\mathcal{T}(L)$, para fins de simplificação, consideraremos válida a relação:

$$L = \{v \in V(T) \mid d(v) = 1\}$$

Como veremos, é possível construir univocamente um corte a partir de um subgrupo, mas para construirmos subgrupos a partir de cortes de maneira coerente, é interessante poder ordenar os elementos de L, mesmo que eles não tenham uma ordem natural bem definida. Nestes casos, é sempre possível definir uma função arbitrária $\varphi : L \mapsto \mathbb{N}_L$ que corresponde a esta ordem, onde $\mathbb{N}_L \subset \mathbb{N}$ é o conjunto dos $|L|$ primeiros números naturais. A função φ induz uma ordem em qualquer subconjunto de L. Esta ordem ajudará a criar subgrupos canonicamente a partir de cortes.

Ao longo desta dissertação, poderemos ainda encontrar algumas situações onde será necessário trabalhar com subconjuntos específicos de $\mathcal{T}(L)$. Denotaremos por $\mathcal{T}_U(L)$ o maior subconjunto de $\mathcal{T}(L)$ que contém apenas árvores filogenéticas sem raiz; e por $\mathcal{T}_R(L)$ o maior subconjunto de $\mathcal{T}(L)$ que contém apenas árvores filogenéticas com raiz. A notação $\mathcal{T}^*(L)$ será usada para o maior subconjunto de $\mathcal{T}(L)$ composto apenas por árvores filogenéticas completamente resolvidas. As composições $\mathcal{T}_U^*(L)$ e $\mathcal{T}_R^*(L)$ também são possíveis e denotam respectivamente os maiores subconjuntos de $\mathcal{T}^*(L)$ formados unicamente por árvores sem raiz e com raiz. Claramente $\mathcal{T}_U(L)$ e $\mathcal{T}_R(L)$ formam uma bipartição do conjunto $\mathcal{T}(L)$, assim como $\mathcal{T}_U^*(L)$ e $\mathcal{T}_R^*(L)$ formam uma bipartição do conjunto $\mathcal{T}^*(L)$.

A definição de árvores filogenéticas como grafos é provavelmente a mais utilizada. Quando trabalhamos com consensos entre árvores filogenéticas, como veremos nos próximos capítulos, geralmente temos que "desmontar" um grupo de árvores e decidir quais partes escolheremos para formar a árvore consenso. Este tipo de operação pode começar a ser um pouco complicada usando grafos. Felizmente, há definições diferentes de árvores filogenéticas, usando conceitos de conjuntos, que são mais simples de serem usadas e que não só permitem simplificar a definição de consenso entre árvores filogenéticas, como também definir medidas de distância entre elas. Uma destas medidas é apresentada na Seção 2.1, logo a seguir.

2.1 Cortes e Sistemas de Cortes

As definições de corte, sistema de cortes e compatibilidade entre cortes apresentadas neste trabalho foram extraídas de uma monografia de Andreas Dress [8, Sessão 2.5].

Um *corte*, ou *split* em inglês, $S = \{A, B\}$ de um conjunto qualquer X é uma bipartição de X em dois subconjuntos não vazios A e B. O conjunto de todos os cortes possíveis do conjunto X é denotado por $\mathcal{S}(X)$ e qualquer subconjunto $\mathcal{S}$ de $\mathcal{S}(X)$ é denominado um *sistema de cortes* definido sobre X. Dois cortes $S, S' \in \mathcal{S}(X)$ são chamados *compatíveis* se e somente se existem subconjuntos $A \in S$ e $A' \in S'$ tais que $A \cap A' = \emptyset$; caso contrário, S e S' são chamados *incompatíveis*.

Dada uma árvore filogenética $T \in \mathcal{T}(L)$, para qualquer aresta $e \in E(T)$, escolhendo-se um vértice qualquer $v \in V(T)$ tal que v é uma folha de T, é possível dividir o conjunto L em dois subconjuntos:

$$\begin{aligned} A &= \{x \in L \mid \text{o caminho de } x \text{ a } v \text{ em } T \text{ não passa por } e\} \\ B &= \{x \in L \mid \text{o caminho de } x \text{ a } v \text{ em } T \text{ passa por } e\} \end{aligned}$$

É simples perceber que $A \cup B$ é o conjunto de folhas de T e que $A \cap B = \emptyset$. Desta forma, $S = \{A, B\}$ é um corte do conjunto de folhas de T, logo, do conjunto L. Note que a escolha arbitrária do vértice não influi na composição do corte, uma vez que a ordem dos subconjuntos em um corte é arbitrária, ou seja, $\{A, B\} = \{B, A\}$. Assim, podemos definir o sistema de cortes $\mathcal{S}_L(T)$, ou simplesmente $\mathcal{S}(T)$ quando o conjunto L for claro no contexto, composto por todos os cortes de L relacionados a arestas de T.

Um resultado importante relacionando cortes e árvores filogenéticas é que, se $S, S' \in \mathcal{S}_L(T)$, então S é compatível com S'. Outro resultado igualmente importante é que, qualquer que seja $T \in \mathcal{T}(L)$, a árvore T pode ser determinada a partir de $\mathcal{S}_L(T)$. Além disso, também é sabido que tanto o número de arestas de uma árvore filogenética completamente resolvida sem raiz com n folhas quanto a maior cardinalidade possível para um conjunto de cortes compatíveis dois a dois definidos sobre um mesmo conjunto de n elementos é $2n - 3$.

Dado $l \in L$, um corte do tipo $\{\{l\}, L \setminus \{l\}\}$ é chamado *corte trivial* de L. O sistema de cortes formado por todos cortes triviais de um conjunto L é denotado por $\mathcal{S}^*(L)$. Denotamos por $\Theta(L)$ o conjunto de todos os sistemas de cortes definidos sobre L compostos apenas por cortes compatíveis e que obrigatoriamente contêm $\mathcal{S}^*(L)$. Definimos a relação $\alpha \subseteq \Theta(L) \times \Theta(L)$ da seguinte maneira:

$$\alpha = \{(\mathcal{S}, \mathcal{S}') \in \Theta(L) \times \Theta(L) \mid \mathcal{S}' \subset \mathcal{S} \text{ e } |\mathcal{S}'| = |\mathcal{S}| - 1\}$$

A relação α relaciona um sistema de cortes qualquer $\mathcal{S}$ a todos os sistemas $\mathcal{S}'$ que podem ser obtidos de $\mathcal{S}$ pela remoção de um de seus cortes. Uma vez definida a relação α, é possível definir um grafo $G_{\Theta(L)}$ da seguinte maneira:

$$\begin{aligned} V(G_{\Theta(L)}) &= \Theta(L) \\ E(G_{\Theta(L)}) &= \{\mathcal{S}\mathcal{S}' \mid (\mathcal{S}, \mathcal{S}') \in \alpha \text{ ou } (\mathcal{S}', \mathcal{S}) \in \alpha\} \end{aligned}$$

Um vértice em $G_{\Theta(L)}$ tem a propriedade de diferir de seus vizinhos por exatamente um corte. Assim, qualquer caminho entre dois vértices neste grafo define uma seqüência de inserções e remoções de cortes que transforma o sistema de cortes de uma das extremidades do caminho no sistema de cortes da outra extremidade.

Teorema 2.1.1 *Qualquer que seja $\mathcal{S} \in \Theta(L)$, existe um caminho $\mathcal{S}^* \rightsquigarrow \mathcal{S}$ em $G_{\Theta(L)}$.*

Prova Provaremos o teorema por indução no tamanho n do conjunto $|\mathcal{S} \setminus \mathcal{S}^*|$.

Base: Para $n = 0$ a hipótese se verifica, pois o único conjunto em $\Theta(L)$ para o qual $|\mathcal{S} \setminus \mathcal{S}^*| = 0$ é o próprio conjunto $\mathcal{S}^*$.

Hipótese de Indução: Se $|\mathcal{S} \setminus \mathcal{S}^*| < n$, então existe um caminho $\mathcal{S}^* \rightsquigarrow \mathcal{S}$ em $G_{\Theta(L)}$.

Passo: Seja $\mathcal{S}$ um sistema de cortes tal que $|\mathcal{S} \setminus \mathcal{S}^*| = n \geq 1$. Seja $\mathcal{S}'$ um sistema de cortes tal que $|\mathcal{S} \setminus \mathcal{S}'| = 1$. Note que $|\mathcal{S}' \setminus \mathcal{S}^*| = n - 1$, portanto, por hipótese de indução, existe um caminho $\mathcal{S}^* \rightsquigarrow \mathcal{S}'$ em $G_{\Theta(L)}$. Além disso, $|\mathcal{S} \setminus \mathcal{S}'| = 1$, portanto, $(\mathcal{S}, \mathcal{S}') \in \alpha$ e $\mathcal{S}\mathcal{S}' \in E(G_{\Theta(L)})$. Assim, existe um caminho $\mathcal{S}^* \rightsquigarrow \mathcal{S}$ em $G_{\Theta(L)}$.

□

Teorema 2.1.2 *$G_{\Theta(L)}$ é conexo.*

Prova Por conseqüência do Teorema 2.1.1, para quaisquer $\mathcal{S}_1, \mathcal{S}_2 \in \Theta(L)$, existem os caminhos $\mathcal{S}^* \rightsquigarrow \mathcal{S}_1$ e $\mathcal{S}^* \rightsquigarrow \mathcal{S}_2$ em $G_{\Theta(L)}$. Como $G_{\Theta(L)}$ é um grafo não-orientado, existe também o caminho $\mathcal{S}_1 \rightsquigarrow \mathcal{S}^* \rightsquigarrow \mathcal{S}_2$.

□

Sejam T e U duas árvores pertencentes a $\mathcal{T}(L)$. Definimos a *distância de cortes* entre T e U, denotada por $\rho(T, U)$, como sendo o comprimento do menor caminho que liga $\mathcal{S}(U)$ a $\mathcal{S}(T)$ em $G_{\Theta(L)}$. Provaremos a seguir que $\rho(T, U) = |\mathcal{S}(T)| + |\mathcal{S}(U)| - 2\,|\mathcal{S}(T) \cap \mathcal{S}(U)|$.

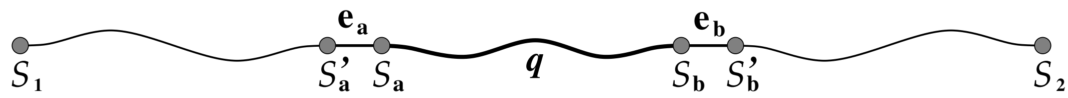

Figura 2.2: Caminho encontrado em $G_{\Theta(L)}$ durante a prova do Lema 2.1.3

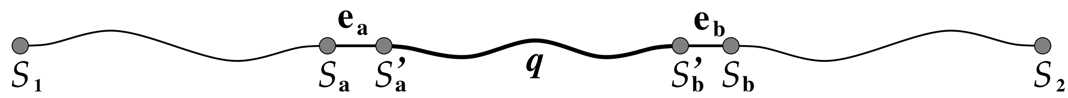

Figura 2.3: Caminho encontrado em $G_{\Theta(L)}$ durante a prova do Lema 2.1.4

Lema 2.1.3 *Seja p um caminho de comprimento mínimo em $G_{\Theta(L)}$ que une os vértices $\mathcal{S}_1$ e $\mathcal{S}_2$. Então p não contém nenhuma aresta $\mathcal{S}'\mathcal{S}$ tal que $\mathcal{S} \setminus \mathcal{S}' = \{x\}$ e $x \notin \mathcal{S}_1 \cup \mathcal{S}_2$.*

Prova Seja E_x o subconjunto de $E(G_{\Theta(L)})$ definido da seguinte forma:

$$E_x = \{\mathcal{S}'\mathcal{S} \in E(G_{\Theta(L)}) \mid \mathcal{S} \setminus \mathcal{S}' = \{x\} \text{ e } x \notin \mathcal{S}_1 \cup \mathcal{S}_2\}.$$

Suponha que haja um caminho p de comprimento mínimo entre $\mathcal{S}_1$ e $\mathcal{S}_2$ que contenha ao menos uma aresta $e \in E_x$. Seja $e_a = \mathcal{S}'_a\mathcal{S}_a \in E_x$ a aresta em E_x mais próxima de $\mathcal{S}_1$ em p. Note que como nenhum dos sistemas de cortes das extremidades do caminho contém o corte x, então deve haver outra aresta pertencente a E_x entre $\mathcal{S}_a$ e $\mathcal{S}_2$. Seja $e_b = \mathcal{S}'_b\mathcal{S}_b \in E_x$ a aresta em E_x mais próxima de $\mathcal{S}_a$. Seja q o sub-caminho de p que une $\mathcal{S}_a$ e $\mathcal{S}_b$. Temos então a situação esquematizada pela Figura 2.2.

Note que, se p é um caminho de comprimento mínimo entre $\mathcal{S}_1$ e $\mathcal{S}_2$, então $q \cup \{\mathcal{S}'_a, \mathcal{S}'_b\}$ é o único caminho de comprimento mínimo entre $\mathcal{S}'_a$ e $\mathcal{S}'_b$ contendo tanto e_a quanto e_b. Note também que todo vértice em q contém o corte x. Assim, é possível definir o caminho $\dot{q}$ da seguinte maneira:

$$\dot{q} = \{\dot{\mathcal{S}} \mid \mathcal{S} \in q \text{ e } \dot{\mathcal{S}} = \mathcal{S} \setminus \{x\}\}$$

É fácil ver que os comprimentos de q e $\dot{q}$ são iguais, mas $\dot{\mathcal{S}}_a = \mathcal{S}'_a$ e $\dot{\mathcal{S}}_b = \mathcal{S}'_b$. Assim, existe um caminho, $\dot{q}$, entre $\mathcal{S}'_a$ e $\mathcal{S}'_b$ mais curto que o sub-caminho de p. Portanto, nem o sub-caminho de p e, é claro, nem o próprio p são caminhos de comprimento mínimo.

□

Lema 2.1.4 *Seja p um caminho de comprimento mínimo em $G_{\Theta(L)}$ que une os vértices $\mathcal{S}_1$ e $\mathcal{S}_2$. Então p não contém nenhuma aresta $\mathcal{S}'\mathcal{S}$ tal que $\mathcal{S} \setminus \mathcal{S}' = \{x\}$ e $x \in \mathcal{S}_1 \cap \mathcal{S}_2$.*

Prova Seja E_x o subconjunto de $E(G_{\Theta(L)})$ definido da seguinte forma:

$$E_x = \{\mathcal{S}'\mathcal{S} \in E(G_{\Theta(L)}) \mid \mathcal{S} \setminus \mathcal{S}' = \{x\} \text{ e } x \in \mathcal{S}_1 \cap \mathcal{S}_2\}.$$

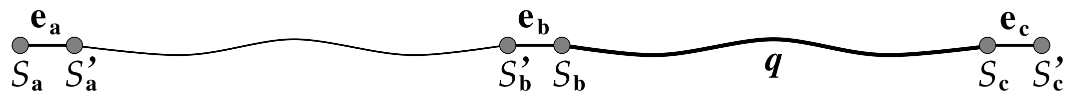

Figura 2.4: Caminho encontrado em $G_{\Theta(L)}$ durante a prova do Lema 2.1.5

Suponha que haja um caminho p de comprimento mínimo entre $\mathcal{S}_1$ e $\mathcal{S}_2$ que contenha ao menos uma aresta $e \in E_x$. Seja $e_a = \mathcal{S}'_a\mathcal{S}_a \in E_x$ a aresta em E_x mais próxima de $\mathcal{S}_1$ em p. Note que como ambos os sistemas de cortes das extremidades do caminho contém o corte x, então deve haver outra aresta pertencente a E_x entre $\mathcal{S}_a$ e $\mathcal{S}_2$. Seja $e_b = \mathcal{S}'_b\mathcal{S}_b \in E_x$ a aresta em E_x mais próxima de $\mathcal{S}'_a$. Seja q o sub-caminho de p que une $\mathcal{S}'_a$ e $\mathcal{S}'_b$. Temos então a situação esquematizada pela Figura 2.3.

Note que, se p é um caminho de comprimento mínimo entre $\mathcal{S}_1$ e $\mathcal{S}_2$, então $q \cup \{\mathcal{S}_a, \mathcal{S}_b\}$ é o único caminho de comprimento mínimo entre $\mathcal{S}_a$ e $\mathcal{S}_b$ contendo tanto e_a quanto e_b. Note também nenhum dos vértices em q contém o corte x, mas todos os cortes de ambas as extremidades são compatíveis com x, pois x está em ambos os sistemas de corte, sendo então compatível com todos os demais cortes nestes sistemas, e, segundo o Lema 2.1.3 nenhum sistema de cortes em p possui um corte que não está nem em $\mathcal{S}_1$ e nem em $\mathcal{S}_2$. Assim, x é compatível com qualquer sistema de cortes em q, sendo possível definir o caminho $\dot{q}$ da seguinte maneira:

$$\dot{q} = \{\dot{\mathcal{S}} \mid \mathcal{S} \in q \text{ e } \dot{\mathcal{S}} = \mathcal{S} \cup \{x\}\}$$

É fácil ver que os comprimentos de q e $\dot{q}$ são iguais, mas $\dot{\mathcal{S}}'_a = \mathcal{S}_a$ e $\dot{\mathcal{S}}'_b = \mathcal{S}_b$. Assim, existe um caminho, $\dot{q}$, entre $\mathcal{S}_a$ e $\mathcal{S}_b$ mais curto que o sub-caminho de p. Portanto, nem o sub-caminho q de p e nem o próprio p são caminhos de comprimento mínimo.

$\square$

Lema 2.1.5 *Seja p um caminho de comprimento mínimo entre $\mathcal{S}_1$ e $\mathcal{S}_2$ em $G_{\Theta(L)}$. Para cada elemento x em $(\mathcal{S}_1 \cup \mathcal{S}_2) \setminus (\mathcal{S}_1 \cap \mathcal{S}_2)$ existe exatamente uma aresta $\mathcal{S}'\mathcal{S} \in p$ tal que $\mathcal{S} \setminus \mathcal{S}' = \{x\}$.*

Prova Seja p um caminho de comprimento mínimo entre $\mathcal{S}_1$ e $\mathcal{S}_2$ em $G_{\Theta(L)}$. Como cada elemento de $(\mathcal{S}_1 \cup \mathcal{S}_2) \setminus (\mathcal{S}_1 \cap \mathcal{S}_2)$ está em apenas um dos sistemas de cortes nas extremidades do caminho p, existe para cada um deles ao menos uma aresta cuja diferença entre as extremidades seja o próprio elemento. Resta provar que esta aresta é única.

Suponha então que exista um elemento $x \in (\mathcal{S}_1 \cup \mathcal{S}_2) \setminus (\mathcal{S}_1 \cap \mathcal{S}_2)$ e que as arestas $e_1 = \mathcal{S}'_1\mathcal{S}_1$ e $e_2 = \mathcal{S}'_2\mathcal{S}_2$ são tais que a diferença entre suas extremidades seja $\{x\}$. Além disso, suponha que não haja nenhuma aresta cuja diferença entre as extremidades seja $\{x\}$ entre e_1 a e_2 no caminho p. Lembremos que apenas uma das extremidades de p possui o corte x. Portanto, se há duas arestas cuja diferença das extremidades seja $\{x\}$, então deve haver ao menos uma terceira, caso contrário, ou ambas as extremidades de p conteriam x ou x não pertenceria a nenhuma delas. Seja $e_3 = \mathcal{S}_3'\mathcal{S}_3$ a aresta deste tipo mais próxima do caminho entre e_1 e e_2. Nomeemos estas três arestas, em relação ao menor sub-caminho de p que contenha as três, da seguinte forma:

e_a : A aresta mais externa cujo vértice mais externo contenha x.

e_b : A aresta central.

e_c : A aresta mais externa cujo vértice mais externo não contenha x.

Temos a situação esquematizada pela Figura 2.4. Seja q o sub-caminho de p que une $\mathcal{S}_b$ a $\mathcal{S}_c$. Como p é um caminho de comprimento mínimo, q também é. Mas, como na prova do Lema 2.1.3, aqui também é possível definir um caminho $\dot{q} = \{\dot{\mathcal{S}} \mid \mathcal{S} \in q \text{ e } \dot{\mathcal{S}} = \mathcal{S} \setminus \{x\}\}$. Como $\dot{\mathcal{S}}_b = \mathcal{S}_b'$ e $\dot{\mathcal{S}}_c = \mathcal{S}_c'$, tanto o sub-caminho de p que une $\mathcal{S}_b'$ a $\mathcal{S}_c'$ quanto o próprio caminho p não são mínimos como supúnhamos.

□

Teorema 2.1.6 *Um caminho de comprimento mínimo entre dois vértices $\mathcal{S}_1$ e $\mathcal{S}_2$ em $G_{\Theta(L)}$ tem exatas $|\mathcal{S}_1| + |\mathcal{S}_2| - 2\,|\mathcal{S}_1 \cap \mathcal{S}_2|$ arestas.*

Prova Seja p um caminho de comprimento mínimo entre $\mathcal{S}_1$ e $\mathcal{S}_2$ em $G_{\Theta(L)}$. Pelo Lema 2.1.3, não há nenhuma aresta $\mathcal{S}'\mathcal{S}$ em p tal que $\mathcal{S} \setminus \mathcal{S}' = \{x\}$ e $x \notin \mathcal{S}_1 \cup \mathcal{S}_2$. Pelo Lema 2.1.4, não há nenhuma aresta $\mathcal{S}'\mathcal{S}$ em p tal que $\mathcal{S} \setminus \mathcal{S}' = \{x\}$ e $x \in \mathcal{S}_1 \cap \mathcal{S}_2$. Pelo Lema 2.1.5, para cada elemento $x \in (\mathcal{S}_1 \cup \mathcal{S}_2) \setminus (\mathcal{S}_1 \cap \mathcal{S}_2)$ há exatamente uma aresta $\mathcal{S}'\mathcal{S}$ em p tal que $\mathcal{S} \setminus \mathcal{S}' = \{x\}$. Assim, seja $\rho(\mathcal{S}_1, \mathcal{S}_2)$ o comprimento de p:

$$\begin{aligned}
\rho(\mathcal{S}_1, \mathcal{S}_2) &= |(\mathcal{S}_1 \cup \mathcal{S}_2) \setminus (\mathcal{S}_1 \cap \mathcal{S}_2)| \\
&= |(\mathcal{S}_1 \setminus (\mathcal{S}_1 \cap \mathcal{S}_2)) \cup (\mathcal{S}_2 \setminus (\mathcal{S}_1 \cap \mathcal{S}_2))| \\
&= |\mathcal{S}_1 \setminus (\mathcal{S}_1 \cap \mathcal{S}_2)| + |\mathcal{S}_2 \setminus (\mathcal{S}_1 \cap \mathcal{S}_2)| \\
&= |\mathcal{S}_1| - |\mathcal{S}_1 \cap \mathcal{S}_2| + |\mathcal{S}_2| - |\mathcal{S}_1 \cap \mathcal{S}_2| \\
&= |\mathcal{S}_1| + |\mathcal{S}_2| - 2\,|\mathcal{S}_1 \cap \mathcal{S}_2|
\end{aligned}$$

□

Waterman [28, Capítulo 14] apresenta outra prova para o valor da distância de cortes (ρ) entre duas árvores. Robinson e Foulds [24] chegam a um resultado semelhante definindo a distância entre grafos de árvores filogenéticas com base no número de operações chamadas contrações e extensões de Bourque necessárias para transformar uma árvore em outra.

2.2 Subgrupos e n-Árvores

Cortes e sistemas de cortes são elementos bem definidos para árvores filogenéticas sem raiz. No entanto, a maioria das definições de consensos entre árvores filogenéticas é definida primeiramente para um grupo de árvores filogenéticas com raiz. Esta seção é dedicada à apresentação de conceitos semelhantes a cortes e sistemas de cortes, mas dedicados a árvores filogenéticas com raiz.

As definições feitas no Capítulo 2 são válidas também para esta seção, porém, ao contrário da Seção 2.1, aqui não nos preocupamos com árvores filogenéticas sem raiz. As definições de n-árvore e função de altura para subgrupos foram extraídas de trabalhos de T. Margush e F. R. McMorris [16] e R. C. Powers [21].

A idéia aqui é agrupar os elementos de um conjunto qualquer X em "classes" e "subclasses" de maneira que, se a interseção de duas classes não for vazia, então uma classe é obrigatoriamente subclasse da outra, o que nos dá uma classificação hierárquica dos elementos do conjunto X, assim como, por exemplo, o sistema de classificação dos seres vivos sugerido por Linnaeus no século XVIII.

Um subconjunto de L é também denominado um *subgrupo* de L. Na literatura em inglês, podemos encontrar tanto o termo *subgroup* quanto o termo *cluster* para designar subgrupos. Um subgrupo S de L tal que $|S| \in \{1, |L|\}$ é chamado de *subgrupo trivial* de L. Todos os demais subgrupos de L são chamados de *não-triviais*. Em outras palavras, S é um subgrupo trivial de L se e somente se $S = L$ ou se $S = \{x\}$ para algum $x \in L$.

Uma família Ψ de subconjuntos de um mesmo conjunto qualquer X que satisfazem a propriedade

$$\text{Para todo } R, S \in \Psi, R \cap S \in \{R, S, \emptyset\}$$

é tradicionalmente conhecida como uma família laminar [11] sobre X e é possível descrever uma árvore filogenética com raiz usando um tipo especial de família laminar denominado n-árvore.

Conjuntos de subgrupos de um mesmo conjunto X, que obedecem a quatro critérios que apresentaremos mais adiante no texto: o que estabelece que o conjunto é uma família laminar e mais três, são chamados de *n-árvores* sobre X, como chamaremos neste trabalho, ou *n-trees*, tal como foram chamados pela primeira vez em 1972 por Bobisud e Bobisud [16].

A letra n presente do termo "n-árvore" refere-se ao número de elementos do conjunto sobre o qual a árvore está definida. Como neste trabalho as n-árvores serão definidas não só sobre L, mas também sobre outros conjuntos, como subconjuntos de L, por exemplo, não usaremos n como uma variável sem deixar claro o seu significado. Assim, a não ser que deixemos claro no texto, não usaremos a variável n como a cardinalidade de conjuntos e, na grande maioria dos casos, denotaremos a cardinalidade de um conjunto X por $|X|$, como fizemos até este ponto.

Um conjunto Ψ de subgrupos de L é denominado uma n-árvore sobre L se e somente se as quatro condições abaixo forem verificadas em Ψ:

1. $\emptyset \notin \Psi$.
2. $L \in \Psi$.
3. $\{x\} \in \Psi$ para todo $x \in L$.
4. $A \cap B \in \{A, B, \emptyset\}$ para todos os subgrupos $A, B \in \Psi$.

Dois subgrupos para os quais a condição 4 é verificada são chamados *compatíveis.*

Seja Ψ uma n-árvore. Uma *função de altura* definida em Ψ é uma função $\eta : \Psi \to \mathbb{N}$ satisfazendo:

- $\eta(L) = 0$.
- Se $A, B \in \Psi$ e $A \subset B$, então $\eta(A) > \eta(B)$.

É possível definir várias funções de altura para os elementos de uma n-árvore. Duas funções, em especial, são bastante utilizadas. São elas:

Função de Altura Canônica: Seja Ψ uma n-árvore sobre o conjunto L e $S \in \Psi$ um subgrupo. Considere uma seqüência $S_0, S_1, \ldots, S_k$ de subgrupos de Ψ onde $S_i \subset S_j$ para $0 \leq i < j \leq k$, $S_0 = S$ e $S_k = L$. Seja $\bar{k}$ o maior valor de k para o qual exista tal seqüência. A função de altura canônica é definida como $\eta_0(S) = \bar{k}$.

Função de Altura de Cardinalidade: Seja Ψ uma n-árvore sobre o conjunto L. A função de altura de cardinalidade, $\eta_\sharp$, como o próprio nome sugere, é definida como a diferença entre as cardinalidades de L e de um subgrupo qualquer $S \in \Psi$. De maneira resumida, $\eta_\sharp(S) = |L| - |S|$.

Seja Ψ um conjunto de subgrupos qualquer e S um subgrupo qualquer. Definimos o conjunto $\Psi[S]$ da seguinte maneira:

$$\Psi[S] = \{A \in \Psi \mid A \subseteq S\}.$$

Teorema 2.2.1 *Se Ψ é uma n-árvore e S um subgrupo qualquer de Ψ, então o conjunto de subgrupos $\Psi[S]$ é uma n-árvore sobre S.*

Prova Para provar que $\Psi[S]$ é uma n-árvore, temos que verificar que as quatro condições necessárias para que um conjunto de subgrupos seja uma n-árvore são válidas para o conjunto $\Psi[S]$.

(1) $\emptyset \notin \Psi[S]$**:** Esta condição é satisfeita, pois $\Psi[S]$ é constituída apenas por elementos de Ψ e $\emptyset \notin \Psi$.

(2) $S \in \Psi[S]$**:** Esta condição é claramente satisfeita, pois $S \in \Psi$ e $S \subseteq S$.

(3) para todo $x \in S, \{x\} \in \Psi[S]$**:** Qualquer que seja $x \in S$, $\{x\} \in \Psi$, pois Ψ é uma n-árvore. Além disso, claramente $\{x\} \subseteq S$, logo, $\{x\} \in \Psi[S]$.

(4) para todo $A, B \in \Psi[S], A \cap B \in \{A, B, \emptyset\}$**:** Devido ao fato de Ψ ser uma n-árvore, esta propriedade vale para qualquer par de subgrupos $A, B \in \Psi$. Como $\Psi[S]$ é formada somente por subgrupos de Ψ, se $A, B \in \Psi[S]$, então $A, B \in \Psi$, logo, para qualquer par de subgrupos em $\Psi[S]$ a propriedade é válida.

□

Sejam Ψ e Υ n-árvores. Se todo subgrupo de Υ for subgrupo de Ψ, então dizemos que Υ é uma *n-subárvore* de Ψ. É interessante notar que não necessariamente as duas n-árvores estão definidas sobre um mesmo conjunto, mas certamente o conjunto no qual a n-árvore Υ está definida é um subgrupo de Ψ. Se S é um subgrupo pertencente a Ψ, denominamos n-subárvore *induzida por* S em Ψ o conjunto $\Psi[S]$.

Dada uma árvore $T \in \mathcal{T}_R(L)$ e um vértice $v \in V(T)$, definimos o subgrupo S_v da seguinte forma:

$$S_v = \{x \in L \mid \text{o caminho de } x \text{ à raiz de } T \text{ contém } v\}$$

Podemos ainda definir o conjunto de subgrupos relativos aos vértices de uma árvore $T \in \mathcal{T}_R(L)$, denotado por $\mathcal{N}_T$, da seguinte maneira:

$$\mathcal{N}_T = \{S_v \mid v \in T\}$$

Teorema 2.2.2 *O conjunto de subgrupos $\mathcal{N}_T$ é uma n-árvore sobre L.*

Prova Para provar que $\mathcal{N}_T$ é uma n-árvore sobre L, temos que verificar que as quatro condições necessárias para que um conjunto de subgrupos seja uma n-árvore são válidas para o conjunto $\mathcal{N}_T$.

(1) $\emptyset \notin \mathcal{N}_T$**:** Se existisse um vértice v para o qual S_v fosse vazio em T, então v seria um vértice que não está conectado a nenhuma das folhas de T. Mas T é um grafo conexo, logo, para quaisquer dois vértices u e v em T existe ao menos um caminho de u a v em T, logo, não existe nenhum vértice v em $V(T)$ para o qual S_v é vazio e $\emptyset \notin \mathcal{N}_T$.

(2) $L \in \mathcal{N}_T$**:** Seja $r \in V(T)$ a raiz da árvore T. Como r está no caminho de qualquer folha até a raiz, então $S_r = L$. Assim, $L \in \mathcal{N}_T$.

(3) para todo $x \in L, \{x\} \in \mathcal{N}_T$**:** Se $v \in L$, então $d(v) = 1$ pela própria definição de L. Se o grau de v é 1, então o único caminho que contém uma folha, v, e a raiz de T é o caminho da raiz a v. Assim, se $v \in L$, então $S_v = \{v\}$. Portanto, para todo elemento v de L existe um subgrupo $\{v\}$ em $\mathcal{N}_T$.

(4) para todo $S_u, S_v \in \mathcal{N}_T, S_u \cap S_v \in \{S_u, S_v, \emptyset\}$**:** Tome dois vértices $u, v \in V(T)$ quaisquer. Se $S_u \cap S_v \neq \emptyset$, existe $w \in S_u \cap S_v$. Mas, como T é um grafo acíclico, só há um caminho da raiz a w e ambos u e v pertencem a este caminho. No entanto, se na ida de w à raiz u aparece antes de v, então $S_u \subset S_v$, caso contrário $S_v \subset S_u$.

□

2.2.1 n-Árvores Completamente Resolvidas

Dizemos que uma n-árvore Ψ é *completamente resolvida* se e somente se a inclusão em Ψ de qualquer subgrupo não vazio que não pertença a Ψ faz com que uma das quatro condições necessárias para que Ψ seja uma n-árvore deixe de ser satisfeita. Como a inclusão de um subgrupo não muda o fato de que qualquer subgrupo trivial pertencer a Ψ e a inclusão de um subgrupo vazio não é permitida, podemos simplificar a definição de uma n-árvore completamente resolvida Ψ como sendo uma n-árvore para a qual a inclusão de qualquer subgrupo não pertencente a Ψ e não vazio fere a condição de que para qualquer $A, B \in \Psi$, temos $A \cap B \in \{A, B, \emptyset\}$.

Teorema 2.2.3 *Uma n-árvore Ψ é completamente resolvida se e somente se para qualquer subgrupo $S \in \Psi$ com cardinalidade maior que um existirem dois subgrupos $A, B \in \Psi$ tais que $A \cup B = S$ e $A \cap B = \emptyset$.*

Prova ($\Rightarrow$)

Suponha que exista em Ψ ao menos o subgrupo S com cardinalidade maior que um que não possua os subgrupos $A, B \in \Psi$ tais que $A \cup B = S$ e $A \cap B = \emptyset$, mas que Ψ seja completamente resolvida mesmo assim. Seja $L_S \subseteq \Psi$ tal que $\bigcup_{R \in L_S} R = S$ com cardinalidade mínima. Note que, como para todo $x \in S$ existe o subgrupo trivial $\{x\} \in \Psi$, então L_S existe e $|L_S| \leq |S|$. Como ainda não existem $A, B \in \Psi$ tais que $A \cup B = S$ e $A \cap B = \emptyset$, temos $3 \leq |L_S| \leq |S|$.

Não é difícil perceber que qualquer elemento do conjunto $\{X \cup Y \mid X, Y \in L_S \text{ e } X \neq Y\}$ pode ser incluído em Ψ sem que a propriedade 4 deixe de ser verificada. Chegamos a uma contradição, pois se Ψ é completamente resolvida, então não deveria existir nenhum subgrupo que pudesse ser adicionado a Ψ sem que Ψ deixasse de ser uma n-árvore. Isso significa que, se Ψ é completamente resolvida, então para qualquer subgrupo $S \in \Psi$ existem subgrupos $A, B \in \Psi$ tais que $A \cup B = S$ e $A \cap B = \emptyset$.

($\Leftarrow$)

Suponha agora que Ψ seja uma n-árvore e não seja completamente resolvida, mas que seja verdade que para qualquer subgrupo $S \in \Psi$ com cardinalidade maior que um existam dois subgrupos $A, B \in \Psi$ tais que $A \cup B = S$ e $A \cap B = \emptyset$. Se Ψ realmente não é completamente resolvida, então podemos acrescentar ao menos um subgrupo a Ψ. Seja R um subgrupo que ainda pode ser acrescentado a Ψ sem que Ψ deixe de ser uma n-árvore. Seja S um subgrupo de cardinalidade mínima em Ψ tal que $R \subset S$. Por hipótese, existem dois subgrupos $A, B \in \Psi$ tais que $A \cup B = S$ e $A \cap B = \emptyset$.

Como R pode ser acrescentado a Ψ, $R \cap A \in \{R, A, \emptyset\}$. Assim, ao menos uma das condições abaixo deve ser verdadeira:

$R \cap A = \emptyset$: Esta situação não é possível, pois se a interseção entre R e A for vazia, então $R \subset B$ e S não é o menor subgrupo em Ψ para o qual $R \subset S$.

$R \cap A = R$, portanto $R \subset A$: Isto não pode ser verdade, pois $A = S \setminus B$ e B não é vazio, assim, $|A| < |S|$ e, neste caso, S não seria o menor subgrupo em Ψ para o qual $R \subset S$.

$R \cap A = A$, portanto $A \subset R$: Como $R \subset S$, então $R \setminus A \subset S \setminus A$. Como supomos que $A \subset R$, então $R \setminus A \neq \emptyset$. Mas $S \setminus A = B$, logo, $R \setminus A \subset B$, o que significa que $R \cap B \neq \emptyset$. Restam então duas possibilidades para $R \cap S$, mas por semelhança com o caso anterior, já sabemos que $R \cap B \neq R$. Ficamos então com a hipótese $B \subset R$, mas como já supomos que $A \subset R$, então $A \cup B \subseteq R$, o que é impossível, pois $A \cup B = S$ e S foi escolhido de tal forma que $R \subset S$.

Chegamos à conclusão de que, se para qualquer subgrupo $S \in \Psi$ com cardinalidade maior que um existirem dois subgrupos $A, B \in \Psi$ tais que $A \cup B = S$ e $A \cap B = \emptyset$, então não existe nenhum subgrupo que possa ser acrescentado a Ψ, o que significa que Ψ é completamente resolvida.

$\square$

Teorema 2.2.4 *O número de subgrupos de uma n-árvore completamente resolvida sobre L é $2|L| - 1$.*

Prova Provaremos este teorema por indução no tamanho de L.

Base: Se $|L| = 1$, só existe uma n-árvore possível, composta por apenas um subgrupo trivial. De fato, $2|L| - 1 = 2 - 1 = 1$

Hipótese de Indução: Se $1 \leq |L| < n$, então o número de subgrupos de uma n-árvore completamente resolvida sobre L é $2\,|L| - 1$.

Passo: Seja L um conjunto de taxa tal que $|L| = n$ e seja Ψ uma n-árvore completamente resolvida sobre L. Como $L \in \Psi$ por definição e Ψ é completamente resolvida, pelo Teorema 2.2.3 sabemos que existem dois subgrupos L' e L'' em Ψ tais que $L' \cup L'' = L$ e $L' \cap L'' = \emptyset$. Sejam $\Psi[L']$ e $\Psi[L'']$ as n-subárvores induzidas por L' e L'', respectivamente. É fácil notar que tanto $\Psi[L']$ quanto $\Psi[L'']$ são completamente resolvidas, pois qualquer subgrupo que possa ser adicionado em uma destas duas árvores também pode ser adicionado a Ψ e Ψ não seria completamente resolvida neste caso. Não é difícil perceber também que, pelo fato de $L' \cap L'' = \emptyset$, a igualdade

$$\Psi = \{L\} \cup \Psi[L'] \cup \Psi[L'']$$

é válida. Como $L' \neq \emptyset$ e $L'' \neq \emptyset$, $|L'| < n$ e $|L''| < n$, assim, por hipótese de indução, $|\Psi[L']| = 2\,|L'| - 1$ e $|\Psi[L'']| = 2\,|L''| - 1$ e podemos calcular o número de subgrupos em Ψ da seguinte forma:

$$\begin{aligned} \Psi &= \{L\} \cup \Psi[L'] \cup \Psi[L''] \\ |\Psi| &= 1 + |\Psi[L']| + |\Psi[L'']| \\ &= 1 + 2\,|L'| - 1 + 2\,|L''| - 1 \\ &= 2(|L'| + |L''|) - 1 \\ &= 2\,|L| - 1 \end{aligned}$$

□

2.3 Relações entre Sistemas de Cortes e n-Árvores

Sistemas de cortes estão para árvores filogenéticas sem raiz assim como n-árvores estão para árvores filogenéticas com raiz. O mesmo se pode dizer dos conceitos de cortes e subgrupos. Em teoria dos grafos, certas relações entre árvores sem raiz e árvores com raiz são muito fáceis de serem percebidas:

- Com a remoção da raiz e arestas adjacentes e a inserção de uma aresta é possível transformar qualquer árvore filogenética completamente resolvida com raiz e mais de uma folha em uma árvore filogenética completamente resolvida sem raiz com o mesmo número de folhas.

- A remoção de uma aresta e a inserção de um vértice e duas arestas pode transformar uma árvore completamente resolvida sem raiz em uma de $2n - 3$ possíveis árvores

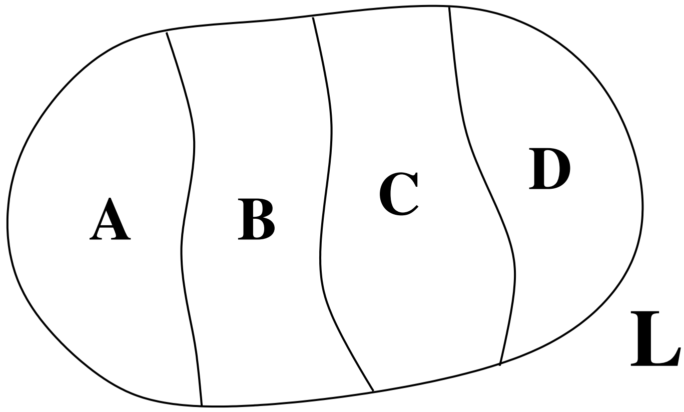

Figura 2.5: Exemplo de um conjunto L qualquer com quatro subconjuntos disjuntos dois a dois tais que a união entre eles é igual ao próprio L.

completamente resolvidas com raiz, onde n é o número de folhas das árvores em questão.

- A remoção de um nó interno de uma árvore filogenética sem raiz dá origem a três ou mais árvores filogenéticas com raiz. Se a árvore de partida for completamente resolvida, então dá origem a exatamente três.

Esta seção é dedicada ao estabelecimento de relações semelhantes entre subgrupos e cortes e entre n-árvores e sistemas de cortes. As propriedades apresentadas logo a seguir serão de grande importância para o algoritmo construtor de árvores consenso apresentado no Capítulo 6.

Como dito anteriormente, determinar um corte a partir de um subgrupo é uma tarefa bastante simples, uma vez que, dado um subgrupo S, o corte $\{S, L \setminus S\}$ é o único corte possível que contém S como um de seus subgrupos. Determinar um subgrupo a partir de um corte é uma tarefa um pouco mais complexa, uma vez que há uma escolha a ser feita entre dois subgrupos. O problema maior é que podemos "perder" propriedades dos cortes de onde eles foram tirados. Como exemplo, tome o conjunto L e seus subconjuntos A, B, C, e D, tais como mostrados na Figura 2.5. Estabelecemos o sistema de cortes $\{\{A, B \cup C \cup D\}, \{A \cup B, C \cup D\}, \{A \cup B \cup C, D\}\}$. Com um pouco de trabalho é possível verificar que estes três cortes são compatíveis dois a dois. Se escolhermos os subgrupos $B \cup C \cup D$, $A \cup B$ e D, não teremos no conjunto de subgrupos a mesma propriedade observada no sistema de cortes, pois os subgrupos $B \cup C \cup D$ e $A \cup B$ não são compatíveis. É interessante notar que se tivéssemos escolhido o subgrupo A ao invés de $B \cup C \cup D$, o

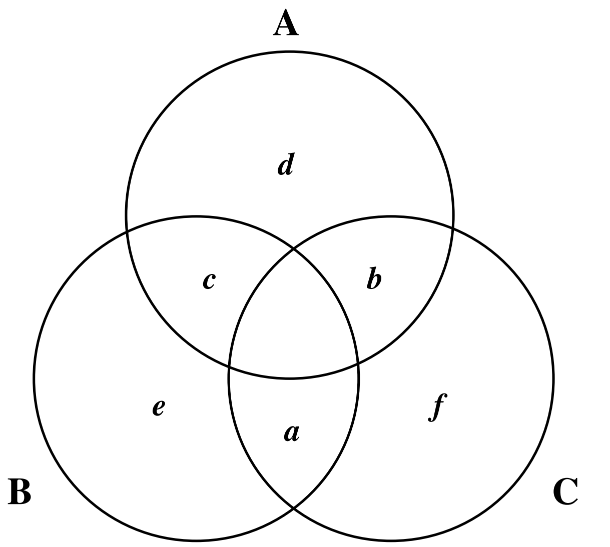

Figura 2.6: Representação esquemática dos subgrupos A, B e C usados na prova do Lema 2.3.1. As variáveis em itálico representam o mínimo atingido pela função φ no subconjunto correspondente. Assim, $a = \min(\varphi((B \cap C) \setminus A))$ e $d = \min(\varphi(A \setminus (B \cup C)))$, por exemplo.

conjunto de subgrupos também teria a propriedade do sistema de cortes de os elementos serem compatíveis dois a dois.

Resta saber se sempre é possível obter um conjunto de subgrupos compatíveis dois a dois a partir de um sistema de cortes compatíveis dois a dois. Para esclarecer esta dúvida, definimos primeiramente o conjunto $\varphi(R)$, onde R é um subgrupo do conjunto L, da seguinte maneira:

$$\varphi(R) = \{\varphi(r) \mid r \in R\},$$

onde φ é a enumeração dos elementos de L definida na página 10.

Dados dois subgrupos distintos R e S de L, dizemos que R é *menor* que S, e denotamos esta relação por $R < S$, se e somente se $|R| < |S|$ ou $|R| = |S|$ e $\min(\varphi(R \setminus S)) < \min(\varphi(S \setminus R))$. Caso contrário, dizemos que R é *maior* que S e denotamos esta relação por $R > S$.

Lema 2.3.1 *Sejam A, B e C três subgrupos distintos de L tais que $|A| = |B| = |C|$. Se $A < B$ e $B < C$, então $A < C$.*

Prova A Figura 2.6 representa esquematicamente os subgrupos A, B e C de L. As variáveis em itálico representam os valores dos mínimos para os subconjuntos de A, B ou C onde elas se encontram. Por definição, assumiremos que $\min(\emptyset) = \infty$.

Note que, pela definição de subgrupos menores, provar que, se $A < B$ e $B < C$, então $A < C$ é equivalente a provar que, se

$$\min(\varphi(A \setminus B)) < \min(\varphi(B \setminus A))$$

e

$$\min(\varphi(B \setminus C)) < \min(\varphi(C \setminus B)),$$

então

$$\min(\varphi(A \setminus C)) < \min(\varphi(C \setminus A)).$$

Podemos utilizar os elementos ilustrados na Figura 2.6 para reescrever as sentenças acima da seguinte maneira: Se

$$\min(b, d) < \min(a, e)$$

e

$$\min(c, e) < \min(b, f),$$

então

$$\min(c, d) < \min(a, f).$$

Levando em consideração que, se $\min(b, d) < \min(a, e)$, então $\min(a, b, d, e) = \min(b, d)$, chegamos finalmente à conclusão de que provar que $A < C$ se $A < B$ e $B < C$ é equivalente a provar que, se

$$\min(a, b, d, e) = \min(b, d)$$

e

$$\min(b, c, e, f) = \min(c, e),$$

então

$$\min(a, c, d, f) = \min(c, d).$$

Para facilitar a notação, definimos $m_1 = \min(a, b, d, e) = \min(b, d)$, $m_2 = \min(b, c, e, f) = \min(c, e)$ e $m_3 = \min(a, c, d, f)$ e provaremos o lema mostrando que sempre que m_1 assume um dos valores b ou d e m_2 assume um dos valores c ou e, o valor assumido por m_3 é $\min(c, d)$.

$m_1 = b$ e $m_2 = c$: Neste caso, sabemos que $c \leq b$ e $c \leq f$, pois $\min(b, c, e, f) = c$. Sabemos também que $b \leq a$, pois $\min(a, b, d, e) = b$. Como $c \leq b$, temos $c \leq a$. Além disso, $c \leq b$ e $b \leq d$ implicam em $c \leq d$. Assim, $m_3 = \min(a, c, d, f) = c = \min(c, d)$.

$m_1 = b$ e $m_2 = e$: Este caso na verdade nunca ocorre, pois se $\min(a, b, d, e) = b$, então $b \leq e$, mas $\min(b, c, e, f) = e$, o que indica que $e \leq b$. Daí conclui-se que $e = b = \infty$ e como $b \leq d$, temos $d = \infty$ mas isto significa que $A \subseteq B$, o que é uma contradição.

$m_1 = d$ e $m_2 = c$: Este caso é um pouco mais difícil de analizar, pois c e d nunca são comparados diretamente, mas apenas um de dois casos pode ocorrer:

Caso 1 ($c \leq d$): Como $\min(b, c, e, f) = c$, temos $c \leq b$ e $c \leq f$. Além disso, $b \leq a$, pois $\min(a, b, d, e) = b$. Assim, $c \leq a$ e $m_3 = \min(a, c, d, f) = c = \min(c, d)$.

Caso 2 ($d < c$): Desta vez, sabemos que $d < f$, pois $d < c$ e $c = min(b, c, e, f)$. Como d também é o mínimo dentre (a, b, d, e), sabemos que $d \leq a$. Assim, $m_3 = \min(a, c, d, f) = d = \min(c, d)$.

$m_1 = d$ e $m_2 = e$: Sabemos que $d \leq a$ e $d \leq e$ porque $\min(a, b, d, e) = d$. Como $e \leq f$ e $e \leq c$ porque $\min(b, c, e, f) = e$, então $d \leq c$ e $d \leq f$, assim, $m_3 = \min(a, c, d, f) = d = \min(c, d)$.

$\square$

Teorema 2.3.2 *Sejam A, B e C três subgrupos distintos de L. Se $A < B$ e $B < C$, então $A < C$.*

Prova A comparação entre dois subgrupos depende da relação de suas cardinalidades, assim, dividimos a prova deste teorema em quatro casos:

$|A| < |B|$ e $|B| < |C|$: Neste caso, $|A|$ é claramente menor que $|C|$ e $A < C$ por definição.

$|A| = |B|$ e $|B| < |C|$: Assim como no caso anterior, $|A|$ é claramente menor que $|C|$ e $A < C$ por definição.

$|A| < |B|$ e $|B| = |C|$: Novamente, $|A|$ é menor que $|C|$ e $A < C$ por definição.

$|A| = |B|$ e $|B| = |C|$: $A < C$ pelo Lema 2.3.1.

$\square$

Seja $S = \{A, B\}$ um corte de L. Não é difícil notar que A e B são subgrupos distintos de L, assim, $A > B$ ou $A < B$, de tal forma que podemos definir S_p como sendo o menor dos subgrupos de S. Chamamos S_p de *subgrupo pequeno* de S. Seja R um corte de L diferente de S. Dizemos que R é *menor* que S, e denotamos esta relação por $R < S$, se e somente se $R_p < S_p$, caso contrário, dizemos que R é *maior* que S, e denotamos esta relação por $R > S$.

Teorema 2.3.3 *Dois cortes distintos R e S de um mesmo conjunto L são compatíveis se e somente se os subgrupos pequenos R_p e S_p forem compatíveis.*

Prova $(\Rightarrow)$

Para facilitar a notação, denotaremos o subgrupo maior de um corte S por S_g, assim, $S = \{S_p, S_g\}$, onde S_p é o subgrupo pequeno do corte e $S_g = L \setminus S_p$.

Provaremos que os subgrupos pequenos de cortes compatíveis são também compatíveis por contradição. Para isso, suponha que R e S são dois cortes compatíveis, mas que R_p e S_p não são subgrupos compatíveis. Deste modo, $R_p \cap S_p \neq \emptyset$, $R_p \not\subset S_p$ e $S_p \not\subset R_p$. Para que R e S sejam compatíveis, é preciso que R_p e S_g ou S_p e R_g ou S_g e R_g tenham a interseção vazia, pela própria definição de compatibilidade entre cortes. Os três casos são analisados a seguir:

$R_p \cap S_g = \emptyset$: Se R_p e S_p não são compatíveis, então $R_p \setminus S_p \neq \emptyset$. Como $S_p \cup S_g = L$, $R_p \setminus S_p = R_p \cap S_g$, assim $R_p \cap S_g \neq \emptyset$.

$R_g \cap S_p = \emptyset$: Também não é verdade, por analogia ao caso anterior.

$R_g \cap S_g = \emptyset$: Não é verdade, pois ambos estão contidos no conjunto L e, se ambos são subgrupos maiores de cortes compatíveis distintos, então a cardinalidade de ambos é maior ou igual a $\frac{|L|}{2}$, sendo que a igualdade pode ser verificada para no máximo um deles pelo fato de se tratarem de cortes compatíveis e distintos, assim, se supusermos que a interseção entre R_g e S_g é vazia, chegamos à conclusão que $|R_g \cup S_g| > |L|$, o que é uma contradição.

Analisando todos os casos, chegamos à conclusão que, se R_p e S_p não são subgrupos compatíveis, então R e S não são cortes compatíveis.

$(\Leftarrow)$

Para provar que se os subgrupos pequenos R_p e S_p forem compatíveis os cortes R e S também são, provaremos a contra-positiva, ou seja, provaremos que, se dois cortes R e S não são compatíveis então os subgrupos pequenos R_p e S_p também não são.

Para tanto, basta lembrar que, pela definição de compatibilidade entre cortes, se R e S não são compatíveis, então nenhum par de subconjuntos formado por um dos subconjuntos de R e um dos subconjuntos de S possui interseção não-vazia. Como $R_p \in R$ e $S_p \in S$, então $R_p \cap S_p \neq \emptyset$.

□

Seja $T \in \mathcal{T}_U(L)$ uma árvore filogenética sem raiz. Denotamos por $\mathcal{F}(T)$ o conjunto formado pelos subgrupos pequenos de todos os cortes em $\mathcal{S}(T)$.

Chamamos de *n-árvore maximal* de $\mathcal{F}(T)$ qualquer n-árvore contida em $\mathcal{F}(T)$ que não seja n-subárvore de nenhuma outra n-árvore contida em $\mathcal{F}(T)$. A Figura 2.7 ilustra o conjunto $\mathcal{F}(T)$ para duas árvore filogenéticas. Nela são destacadas as propriedades apresentadas no Lema 2.3.4 e no Teorema 2.3.5.

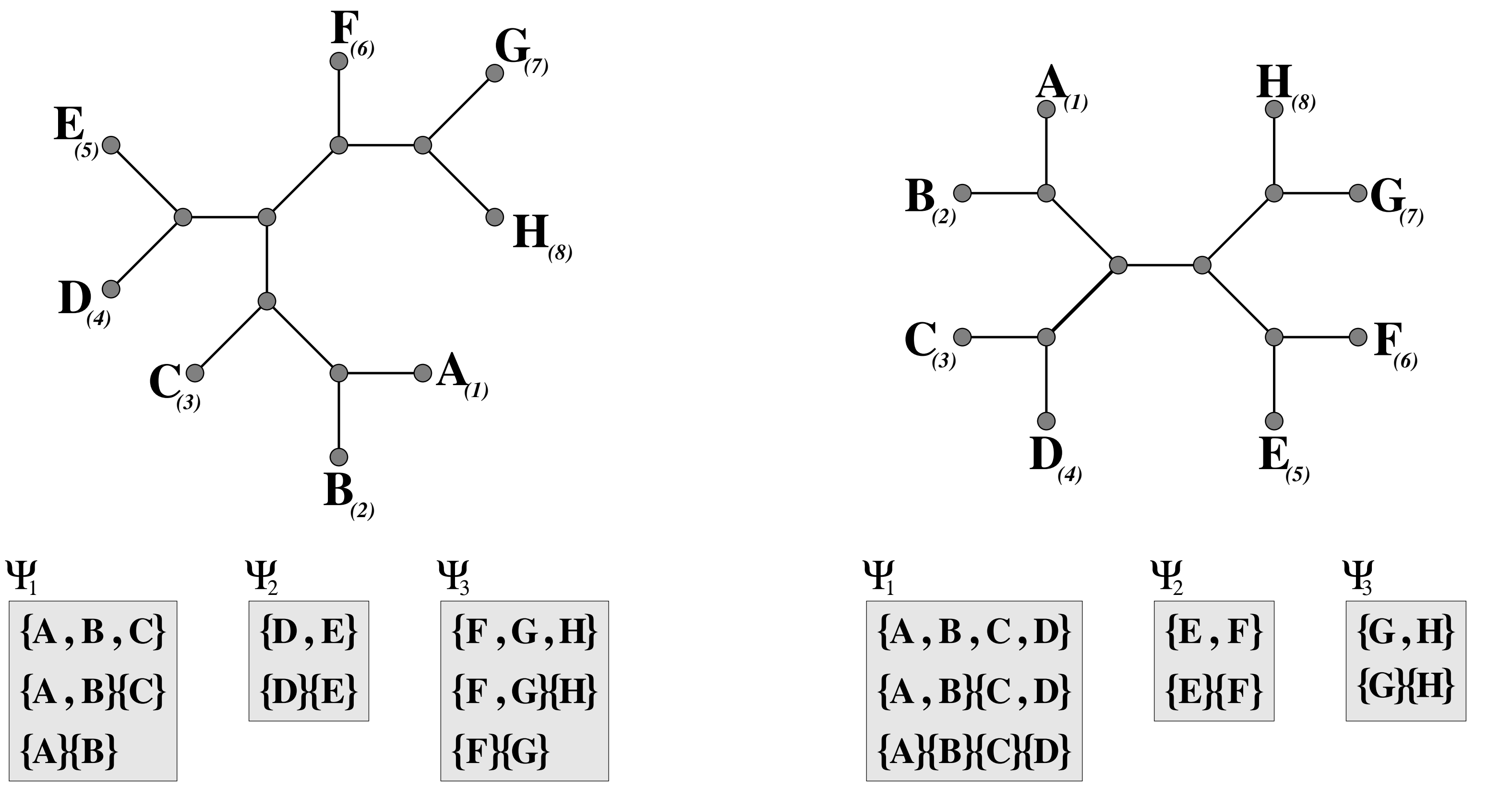

Figura 2.7: Exemplo do conjunto $\mathcal{F}(T)$ para duas árvores filogenéticas. O conjunto $\mathcal{F}(T)$ é apresentado na forma de suas n-árvores maximais Ψ_1, Ψ_2 e Ψ_3, que são disjuntas como determina o Lema 2.3.4. Note também que, como as árvores são completamente resolvidas, há apenas três n-árvores maximais e todas elas são completamente resolvidas, como determina o Teorema 2.3.5. No exemplo da esquerda, a cardinalidade dos subgrupos é suficiente para determinar os subgrupos pequenos, de modo que não é necessário conhecer a função φ. Já no exemplo da direita, a função φ, cujos valores são mostrados entre parênteses, determina que $\{A, C, B, D\}$ é o subgrupo pequeno e não $\{E, F, G, H\}$, pois $\min(\varphi(\{A, B, C, D\} \setminus \{E, F, G, H\})) = \min(\{1, 2, 3, 4\}) = 1$ e $\min(\varphi(\{E, F, G, H\} \setminus \{A, B, C, D\})) = \min(\{5, 6, 7, 8\}) = 5$ e $1 < 5$.

Lema 2.3.4 *Seja L um conjunto de cardinalidade maior que dois. Se $T \in \mathcal{T}_U(L)$ é uma árvore filogenética sem raiz, então o conjunto $\mathcal{F}(T)$ é a união disjunta de um número finito de n-árvores definidas sobre subconjuntos de L.*

Prova Como arestas folhas sempre determinam cortes triviais em $\mathcal{S}(T)$, e subgrupos pequenos de cortes triviais são subgrupos triviais, para qualquer elemento x de L existe um subgrupo $\{x\}$ em $\mathcal{F}(T)$, não importando qual seja a árvore T, pois L tem pelo menos três elementos.

Se considerarmos para cada grupo X em $\mathcal{F}(T)$ o conjunto $\mathcal{F}(T)[X]$ chegaremos rapidamente à conclusão de que este conjunto é uma n-árvore sobre X, pois:

- X pertence ao conjunto.
- Qualquer elemento x de X é também elemento de L, portanto $\{x\}$ pertence a $\mathcal{F}(T)$ e ao conjunto definido.
- Como $\mathcal{F}(T)$ é formado por subgrupos pequenos de cortes, $\emptyset$ não pertence a $\mathcal{F}(T)$ pela definição de corte.
- Como os subgrupos de $\mathcal{F}(T)$ são subgrupos pequenos de cortes compatíveis, todos eles são compatíveis entre si pelo Teorema 2.3.3.

É claro que todo subgrupo de $\mathcal{F}(T)$ está em no mínimo uma n-árvore, pois o conjunto definido anteriormente é definido para qualquer subgrupo em $\mathcal{F}(T)$. Como $\mathcal{F}(T)$ é finito, o conjunto de n-árvores definidas sobre subgrupos de $\mathcal{F}(T)$ também é finito. Assim, $\mathcal{F}(T)$ é a união de um conjunto finito de n-árvores. Também é fácil perceber que toda n-árvore maximal de $\mathcal{F}(T)$ é da forma $\mathcal{F}(T)[X]$ para certo $X \in \mathcal{F}(T)$.

Sejam duas n-árvores maximais de $\mathcal{F}(T)$ definidas sobre A e B respectivamente. Como A e B são subgrupos pequenos de cortes compatíveis, temos $A \cap B = \emptyset$, caso contrário teríamos $A \subset B$ ou $B \subset A$, o que significaria, respectivamente, que $\mathcal{F}(T)[A]$ é uma n-subárvore de $\mathcal{F}(T)[B]$ ou $\mathcal{F}(T)[B]$ é uma n-subárvore de $\mathcal{F}(T)[A]$. Assim, o conjunto das n árvores maximais $\mathcal{F}(T)$ é formado por n-árvores disjuntas, uma vez que os subgrupos nos quais duas n-árvores maximais estão definidos não têm interseção.

Por outro lado, qualquer subgrupo X em $\mathcal{F}(T)$ pertence a pelo menos uma n-árvore maximal, pois $X \in \mathcal{F}(T)[X]$ e, se $\mathcal{F}(T)[X]$ não for maximal, é porque ela é n-subárvore de uma n-árvore maximal.

Chegamos à conclusão de que o conjunto de árvores maximais de $\mathcal{F}(T)$ cobre todos os subgrupos de $\mathcal{F}(T)$ e que todas as árvores maximais são disjuntas. Assim, $\mathcal{F}(T)$ é a união disjuntas das árvores maximais de $\mathcal{F}(T)$.

$\square$

Teorema 2.3.5 *Seja L um conjunto de cardinalidade maior que dois e $T \in \mathcal{T}_U(L)$. Então T é completamente resolvida se e somente se $\mathcal{F}(T)$ tiver exatamente três n-árvores maximais e estas árvores forem completamente resolvidas.*

Prova ($\Rightarrow$)

Suponha que T seja uma árvore filogenética completamente resolvida e sem raiz, mas que $\mathcal{F}(T)$ não tenha as propriedades acima. Temos dois casos:

Caso 1: *O número de n-árvores maximais em $\mathcal{F}(T)$ é diferente de 3.*

Como todos os subgrupos em $\mathcal{F}(T)$ são compatíveis, as n-árvores maximais de $\mathcal{F}(T)$ são disjuntas, ou seja, elas não compartilham nenhum subgrupo. Assim, a união entre os maiores subgrupos das n-árvores maximais de $\mathcal{F}(T)$ é L e a interseção entre eles é vazia.

O número de árvores maximais não pode ser um, pois toda árvore maximal em $\mathcal{F}(T)$ é definida sobre um subgrupo contido em $\mathcal{F}(T)$ e sabemos que, como os subgrupos em $\mathcal{F}(T)$ são subgrupos pequenos de cortes definidos pelas arestas de T em L, para cada elemento $x \in L$ há um subgrupo unitário que contém x em $\mathcal{F}(T)$ e o maior subgrupo em $\mathcal{F}(T)$ não é maior que $\frac{|L|}{2}$, assim, não há subgrupo em $\mathcal{F}(T)$ cuja n-subárvore induzida seja o próprio conjunto $\mathcal{F}(T)$.

Suponha que o número de n-árvores maximais é 2. Assim, há dois subgrupos A e B em $\mathcal{F}(T)$ tais que $A \cup B = L$ e $A \cap B = \emptyset$. Se isto é verdade, então A e B pertencem ao mesmo corte de L, logo, não podem ambos estar ao mesmo tempo em $\mathcal{F}(T)$, uma vez que apenas um deles é o subgrupo pequeno do corte.

Resta a hipótese de que o número de n-árvores maximais em $\mathcal{F}(T)$ é maior que 3. Sejam A e B dois subconjuntos de L sobre os quais duas n-árvores maximais distintas de $\mathcal{F}(T)$ estão definidas. Novamente temos duas possibilidades:

1.a) $|A \cup B| < \frac{|L|}{2}$**:** Se isto for verdade, então o subgrupo $A \cup B$, que não está em $\mathcal{F}(T)$, poderia ser inserido neste conjunto sem problemas, pois $A \cup B$ é claramente compatível tanto com A quanto com B quanto com qualquer outro subgrupo em $\mathcal{F}(T)$. Mas se o subgrupo $A \cup B$ é compatível com qualquer subgrupo em $\mathcal{F}(T)$, então o corte $\{A \cup B, L \setminus (A \cup B)\}$ é compatível com todos os cortes de $\mathcal{S}(T)$, pelo Teorema 2.3.3, e pode ser adicionado a este sistema de cortes sem problemas, o que significa que $\mathcal{S}(T)$ não é maximal, assim como T não é completamente resolvida.

1.b) $|A \cup B| \geq \frac{|L|}{2}$**:** Como $\mathcal{F}(T)$ tem mais de três n-árvores maximais, podemos encontrar mais dois subconjuntos C e D de L sobre os quais duas outras n-árvores maximais de $\mathcal{F}(T)$ estão definidas. Como $|A \cup B| \geq \frac{|L|}{2}$, $|C \cup D| \leq \frac{|L|}{2}$ e, usando o mesmo raciocínio do caso anterior, chegamos à conclusão que também neste caso T não é completamente resolvida.

Caso 2: *Ao menos uma das n-árvores maximais de $\mathcal{F}(T)$ não é completamente resolvida.*

Seja A o subconjunto de L sobre o qual uma das n-árvores maximais não completamente resolvidas de $\mathcal{F}(T)$ está definida. Pela definição de n-árvore, A pertence a esta n-árvore, logo $A \in \mathcal{F}(T)$. Então, A é o subgrupo pequeno do corte $\{A, L \setminus A\}$ e $|A| \leq \frac{|L|}{2}$. Se a n-árvore maximal de $\mathcal{F}(T)$ definida sobre A não é completamente resolvida, então ela admite a inclusão de ao menos um subgrupo B compatível com todos os demais subgrupos. Como B não pertence à n-árvore, mas pode ser incorporado a ela, $|B| < |A|$. Assim, B certamente é o subgrupo pequeno de $\{B, L \setminus B\}$ que, pelo Teorema 2.3.3, é compatível com todos os cortes do sistema $\mathcal{S}(T)$ que, portanto, não é maximal, o que significa que T também não é completamente resolvida.

($\Leftarrow$)

Suponha agora que haja exatamente três n-árvores maximais completamente resolvidas em $\mathcal{F}(T)$.

Sejam A, B e $C \in \mathcal{F}(T)$ os três subgrupos de L sobre os quais as três n-árvores maximais de $\mathcal{F}(T)$ estão definidas. Quaisquer dois subgrupos em $\mathcal{F}(T)$ definem cortes compatíveis em $\mathcal{S}(T)$ pela definição de $\mathcal{F}(T)$.

O número de subgrupos em $\mathcal{F}(T)$, dado que as três n-árvores maximais de $\mathcal{F}(T)$ são completamente resolvidas, é:

$$2\,|A| - 1 + 2\,|B| - 1 + 2\,|C| - 1 = 2(|A| + |B| + |C|) - 3 = 2\,|L| - 3$$

e este é o maior número de cortes compatíveis dois a dois que um sistema de cortes definido sobre L pode ter, como foi dito na página 11. Assim, se $\mathcal{F}(T)$ tem exatamente três n-árvores maximais e estas n-árvores são completamente resolvidas, então o sistema de cortes $\mathcal{S}(T)$ é máximo, o que significa que T é completamente resolvida.

□

Capítulo 3

Consensos entre Árvores Filogenéticas

É bastante comum que biólogos, ao tentar construir a árvore filogenética para um conjunto de seres vivos, acabe com um conjunto de resultados diferentes, muitas vezes igualmente prováveis. Isso pode acontecer por diversas razões:

- Várias árvores filogenéticas podem ser construídas usando o mesmo método de reconstrução e dados de origem diferentes, como dados de origem biomolecular (seqüências de aminoácidos ou nucleotídeos) ou dados de origem morfológica.
- Várias árvores filogenéticas podem ser criadas usando o mesmo conjunto de dados, mas métodos de construção distintos.
- Em alguns casos, principalmente quando o método de reconstrução tenta encontrar a árvore filogenética que otimiza uma certa função, várias árvores filogenéticas diferentes podem ser criadas usando o mesmo método e o mesmo conjunto de dados.

Nestes casos, pode ser bastante interessante combinar todas as árvores obtidas como uma forma de construção de uma única árvore filogenética representando a história evolutiva de todos os seres vivos do conjunto estudado. Para isso, é necessário utilizar o que chamamos de um *método de consenso* entre árvores filogenéticas. De maneira bastante formal, um método de consenso é uma função da forma

$$C : \mathcal{T}^k \mapsto \mathcal{T},$$

onde $\mathcal{T}^k$ representa o produto cartesiano $\mathcal{T} \times \mathcal{T}$ aplicado $k-1$ vezes.

De maneira um pouco menos formal, um método de consenso é uma função que associa uma árvore filogenética a um conjunto de k árvores filogenéticas. Na literatura sobre consensos, os elementos de $\mathcal{T}^k$ são normalmente denominados *profiles*, nomenclatura que não usaremos nesta dissertação.

Não é muito comum encontrar textos defendendo o uso de árvores consenso como método de construção de árvores filogenéticas. Um dos raros exemplos é o artigo de Cynthia Phillips e Tandy Warnow, em que elas apresentam a árvore mediana assimétrica [20]. No artigo, Phillips e Warnow entendem os problemas com o uso de consensos para a construção de árvores filogenéticas, mas assumem que boa parte destes problemas surge pelo fato de que, até o momento, poucos métodos de consenso se preocupam em produzir árvores tão resolvidas quanto possível, ou seja, com o menor número de politomias.

A utilidade de consensos como forma de construção de árvores filogenéticas é contestada normalmente de diversas formas:

Conjuntos de Dados: Alguns biólogos questionam a utilidade de consensos entre árvores filogenéticas pelo fato de os métodos de construção de consenso não se basearem nos dados que originam o conjunto de árvores. De fato, isto é verdade para a grande maioria dos casos, embora Bryant [4] apresente alguns métodos de consenso que decompõem as árvores em dados como distâncias entre os elementos de L e utilizam métodos de reconstrução de árvores filogenéticas usando como dados de entrada um conjunto de dados baseado na combinação dos dados oriundos desta decomposição. Por outro lado, embora os métodos baseados apenas nas topologias das árvores desconsiderem os dados originais, eles se baseiam em partições do conjunto de seres vivos inferidas a partir dos dados originais através de métodos (supostamente) confiáveis, ou seja, apesar de não usados diretamente, os dados originais não são completamente ignorados.

Apresentação de Contra-Exemplos: Vários cientistas duvidam que a combinação de árvores possa construir uma árvore confiável pelo fato de terem encontrado exemplos de conjuntos de árvores em que o uso de um determindado método de consenso errou ao escolher as partes que deveriam fazer parte da árvore consenso. Em especial, temos o exemplo descrito por Michael J. Sharkey e Jason W. Leathers [26], onde o método da regra da maioria, descrito na Seção 3.4, falha ao reconstruir uma árvore filogenética com base em três outras árvores obtidas através de dados reais. A argumentação de Sharkey e Leathers é falha, no entanto, em vários pontos. Primeiramente, eles usam dados modificados de dados obtidos empiricamente, mas não dizem qual o grau de modificação. Qualquer conjunto de dados pode ser modificado até que criemos um conjunto que satisfaça o que queremos demonstrar. Seriam eles incapazes de demostrar a ocorrência do erro usando apenas dados empíricos? Outro problema da argumentação é o fato de o consenso ter sido construído com base em três árvores apenas, um dos menores tamanhos possíveis para um conjunto de

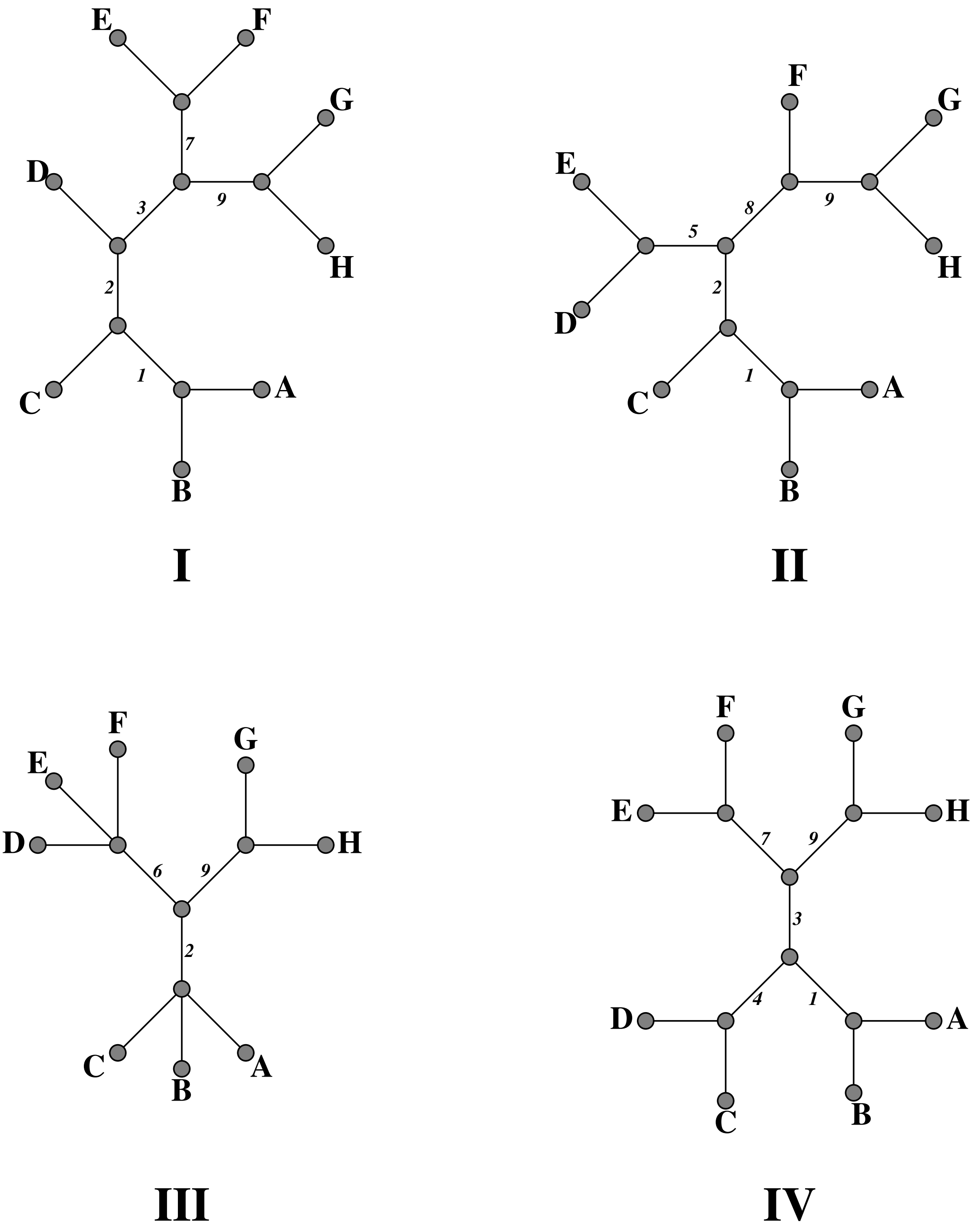

Figura 3.1: Coleção de árvores filogenéticas utilizadas nos exemplos dos métodos de consenso apresentados neste capítulo. As árvores são construídas sobre o conjunto $L = \{A, B, C, D, E, F, G, H\}$. Os números das arestas indicam os cortes correspondentes definidos sobre L. Arestas que determinam cortes triviais não são numeradas na figura.

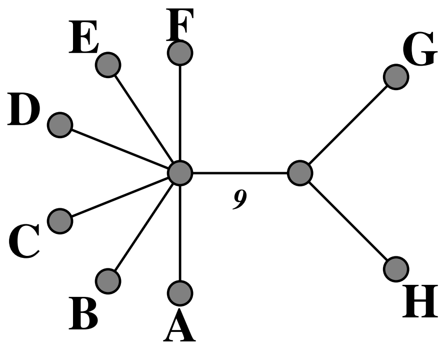

Figura 3.2: Consenso estrito entre as árvores apresentadas na Figura 3.1

árvores filogenéticas, enquanto se espera que a qualidade do consenso cresça juntamente com o número de árvores no conjunto. Finalmente, eles afirmam que "quase todos os conjuntos onde o consenso da regra da maioria difere do consenso estrito irá ilustrar o problema", mas não fornecem nenhum outro exemplo. Em resumo, o que eles mostram em seu artigo não dá indícios suficientes para que um defensor das árvores consenso mude de idéia.

Comparação entre Árvores: Este grupo é um subconjunto do primeiro grupo. Trata-se do grupo de cientistas que acreditam que árvores consenso são úteis sim, mas apenas para a visualização das diferenças entre as árvores do conjunto. Esta visão é bastante clara no artigo de Arne Anderberg e Anders Tehler [1]. Na verdade, este grupo é um subconjunto do primeiro porque normalmente estes cientistas se baseiam no fato de que árvores consenso nada mais são do que a combinação de diferentes topologias, o que não deixa de ser uma verdade, mas poucos métodos até agora foram projetados para um fim que não seja a comparação entre árvores de um conjunto, de modo que o uso de consensos para a construção de árvores filogenéticas ainda é uma questão pouco explorada e que não deve ser negligenciada.

Apresentaremos a seguir alguns dos métodos de consenso mais conhecidos. Em todos os casos em que exemplos são apresentados, o conjunto original de árvores é o apresentado na Figura 3.1.

3.1 Consenso Estrito

Não é por acaso que o método de consenso estrito recebe este nome. De todas as definições de consenso, esta é sem dúvida a que mais restringe o uso de subgrupos encontrados nas árvores filogenéticas da coleção, permitindo apenas que subgrupos encontrados em todas as árvores filogenéticas sejam usados para construir o consenso.

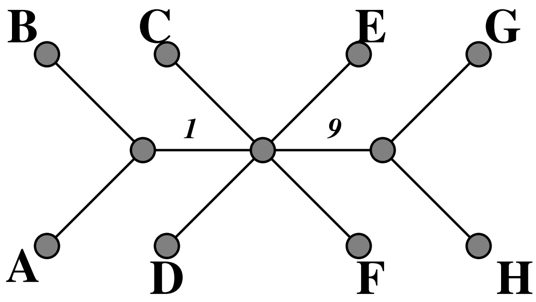

Figura 3.3: Consenso de Componentes Combináveis entre as árvores apresentadas na Figura 3.1

O consenso estrito foi definido para árvores filogenéticas com raiz por Sokal e Rohlf em 1981 e usado pela primeira vez por Schuh e Polhemus no mesmo ano. Apesar de ser definido para árvores filogenéticas com raiz, a definição do método de consenso estrito pode ser estendida para árvores filogenéticas sem raiz apenas trocando a palavra "subgrupo" pela palavra "corte" em sua definição.

Uma característica importante do consenso estrito, que não é compartilhada por todas as definições de consenso, é o fato de os subgrupos (ou cortes) da árvore estrita serem compatíveis com todos os subgrupos de todas as árvores da coleção. Além disso, uma outra grande vantagem do consenso estrito é que somente uma árvore por coleção corresponde ao consenso. Por outro lado, há uma perda de resolução da árvore estrita se comparada com as árvores da coleção, o que significa que as árvores estritas costumam ter mais nós politômicos ou nós politômicos de graus maiores que as árvores das coleções às quais elas correspondem.

3.2 Componentes Combináveis

O nome deste de consenso traduzido para os termos usados no contexto deste trabalho seria "Subgrupos Compatíveis" e ele é a conseqüência direta da constatação feita por Bremer [2] de que alguns dos subgrupos (chamados de *componentes* por Bremer) que não estão presentes em todas as árvores filogenéticas podem ser compatíveis (ou *combináveis*, segundo Bremer) com todos os subgrupos que fazem parte da árvore estrita. Isto pode ocorrer quando pelo menos uma das árvores filogenéticas da coleção possuir uma politomia e alguma das demais árvores possuir um subgrupo capaz de resolver a politomia total ou parcialmente.

A árvore do consenso de componentes combináveis é criada usando todos o subgrupos encontrados na coleção de árvores que são compatíveis com todas as árvores filogenéticas da coleção ao mesmo tempo. Claramente todos os subgrupos do consenso estrito são incluídos também por este método na árvore consenso, além de alguns subgrupos que, apesar de não serem encontrados em todas as árvores, são compatíveis com todos os demais subgrupos presentes na árvore consenso. Por permitir que apenas componentes compatíveis com todas as árvores do conjunto façam parte da árvore consenso ao mesmo tempo que permite que um número maior de subgrupos seja incluído na árvore consenso, o consenso de componentes combináveis é também chamado de consenso *semi-estrito* [14] e Bryant, em sua classificação para métodos de consensos para filogenias [4], ainda apresenta o nome de *Loose Consensus Tree* para este método.

Assim como no caso do consenso estrito, a definição do consenso de componentes combináveis pode ser estendida de árvores com raiz para árvores sem raiz apenas mudando a palavra "subgrupo" para a palavra "corte" na definição. O consenso de componentes combináveis para árvores sem raiz pode ser definido como sendo formado por todos os cortes encontrados na coleção de árvores filogenéticas que são compatíveis com todas as árvores do conjunto.

3.3 Consenso de Nelson

O consenso de Nelson é baseado numa premissa muito simples: a de que a chance de duas árvores filogenéticas que concordam em um subgrupo estarem erradas a respeito da existência deste subgrupo é mínima. Assim, a árvore consenso de Nelson é formada por todos os subgrupos que aparecem em mais do que uma árvore na coleção de árvores filogenéticas. O grande problema com o consenso de Nelson é que a premissa na qual ele se baseia é falsa para muitas coleções de árvores. Subgrupos errados podem ser encontrados em mais de uma árvore filogenética e a conseqüência maior disso é o surgimento de subgrupos freqüentes incompatíveis entre si. Neste caso, nenhuma árvore pode ser construída.

David Bryant chama a atenção para uma observação feita pelo próprio Nelson, que as "armadilhas desta filosofia são muitas, e algumas delas são profundas". De fato, a não existência de uma árvore consenso de Nelson para várias coleções de árvore levou Page a criar a sua variante deste método, criando controvérsias na comunidade de estudiosos das árvores filogenéticas a respeito do que seria de fato o consenso de Nelson. Bremer, no mesmo artigo em que descreve o consenso de componentes combináveis, apresenta resumidamente o consenso de Nelson e cita o fato de que Page tenta resolver o problema dos subgrupos conflitantes.

O que Page propõe, e que Bryant sensatamente chama de *consenso de Nelson-Page*, é, sempre que não for possível usar todos os subgrupos replicados, construir uma árvore com o melhor subconjunto possível.

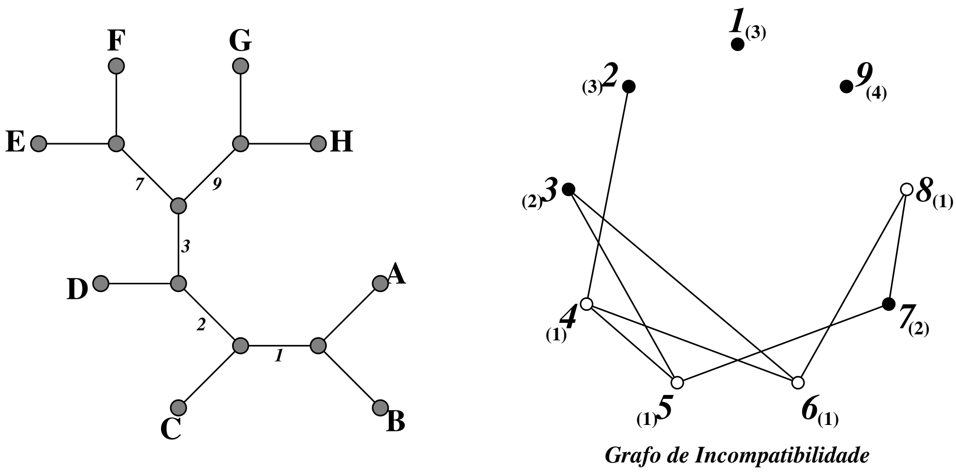

Figura 3.4: Consenso de Nelson entre as árvores apresentadas na Figura 3.1. Pelo grafo de incompatibilidade, podemos ver que a árvore de Nelson corresponde à de Nelson-Page, o que exemplifica o fato de as duas árvores serem iguais quando a de Nelson está definida para a coleção. No grafo de incompatibilidade, vértices pretos indicam os vértices pertencentes ao conjunto independente I que maximiza $W(I)$ (consulte a Seção 3.3.1 para maiores detalhes). Entre parênteses, temos o número do árvores em que cada corte aparece.

3.3.1 Consenso de Nelson-Page

A árvore consenso de Nelson-Page é definida no artigo de Bryant [4] como sendo a árvore determinada pelo conjunto $\mathcal{S}$ de subgrupos não triviais mutuamente compatíveis que maximiza o valor de

$$\sum_{S \in \mathcal{S}} w(S),$$

onde $w(S)$ é definido como sendo o número de árvores da coleção nas quais S pode ser encontrado menos 1. Se há mais de um conjunto $\mathcal{S}$ possível, então a árvore consenso de Nelson-Page é determinada pela intersecção de todos os conjuntos possíveis.

Assim como em todos os casos anteriores, o método de Nelson-Page pode ser estendido para conjuntos de árvores filogenéticas sem raiz apenas trocando a palavra “subgrupo” pela palavra “corte” e Philips e Warnow [20] apresentam um método para a construção de árvores de Nelson-Page, às quais elas se referem em seu artigo como simplesmente árvores de Nelson. O método corresponde aos três passos abaixo:

Passo 1: O conjunto de cortes não triviais das árvores na coleção de árvores filogenéticas é obtido e para cada corte C é atribuído o valor $w(C)$, definido para cortes de maneira

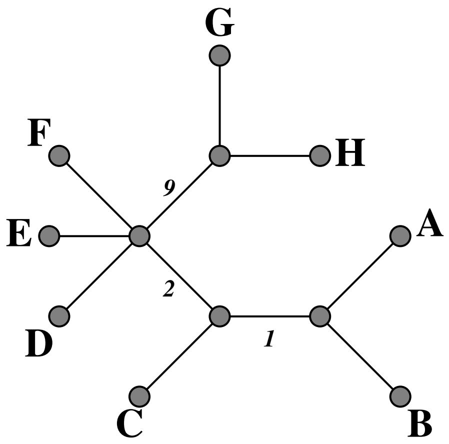

Figura 3.5: Consenso da Regra da Maioria para as árvores apresentadas na Figura 3.1

análoga à definição para subgrupos. Constrói-se então um grafo onde os vértices são os cortes não triviais encontrados na coleção de árvores e dois vértices são adjacentes se e somente se os cortes não forem compatíveis (este grafo é denominado grafo de *incompatibilidade*). Note que a união de cada conjunto independente I deste grafo com o conjunto de cortes triviais de L define uma árvore filogenética sem raiz T_I, não necessariamente completamente resolvida.

Passo 2: Para cada conjunto independente I no grafo, seja

$$W(I) = \sum_{v \in I} w(v).$$

Defina a coleção

$$\mathcal{T} = \{T_I \mid I \text{ é um conjunto independente do grafo que maximiza } W(I)\}.$$

Passo 3: Retorne o consenso estrito de $\mathcal{T}$.

3.4 Regra da Maioria

O método de consenso conhecido como *Regra da Maioria* foi proposto em 1981 por T. Margush e F. R. McMorris usando o conceito de n-árvores. A definição da árvore criada pelo método da Regra da Maioria, segundo o artigo original onde o método foi proposto [16], é a seguinte:

> Seja $\mathcal{C} = \{\Psi_1, \cdots, \Psi_m\}$ uma coleção de n-árvores. A *regra da maioria* de $\mathcal{C}$, denotada por $M(\mathcal{C})$, é o subconjunto do conjunto de subgrupos de L onde $A \in M(\mathcal{C})$ se e somente se $A \in \Psi_i$ para mais da metade das Ψ_is, $1 \leq i \leq m$.

Margush e McMorris provaram que o conjunto definido acima é também uma n-árvore. A prova é bastante simples e consiste em verificar que os subgrupos triviais estão em $M(\mathcal{C})$, pois são comuns a todas as árvores, o conjunto vazio não está em $M(\mathcal{C})$, pois não pertence a nenhuma delas e, sejam A e B dois subgrupos de L. Se tanto A quanto B são encontrados em mais de metade das n-árvores, então existe ao menos uma n-árvore na coleção que contenha tanto A quanto B, o que significa que A e B são compatíveis, assim, qualquer par de subgrupos em $M(\mathcal{C})$ é formado por subgrupos compatíveis, de forma que $M(\mathcal{C})$ tem todas as propriedades exigidas para uma n-árvore.

Assim como os demais métodos apresentados, o método da regra da maioria pode ser facilmente estendido para árvores sem raiz, apenas trocando o termo "n-árvore" pelo termo "sistema de cortes" e o termo "subgrupo" pelo termo "corte". Note que inclusive a prova de que os subgrupos de $M(\mathcal{C})$ são compatíveis, de Margush e McMorris, descrita no parágrafo anterior, pode ser facilmente transformada numa prova para sistemas de cortes.

3.5 Árvore Mediana Assimétrica

Como veremos no Capítulo 6, o método da árvore mediana assimétrica tem algo em comum com o método proposto nesta dissertação, no sentido em que ele atribui a cada árvore possível um valor e, depois, seleciona do conjunto de árvores possíveis as que maximizam este valor. Porém, o valor atribuído a cada árvores é diferente e o tipo de árvore resultante também, pois nem sempre a árvore mediana assimétrica é completamente resolvida.

O método da árvore mediana assimétrica, ou *asymmetric median tree*, foi sugerido por Philips e Warnow [20] no ano de 1996. Elas primeiro definem o valor $val(T, \mathcal{T})$ de uma árvore consenso T em relação a uma coleção de árvores filogenéticas $\mathcal{T}$ da seguinte maneira:

> Seja $w(C)$ (o peso do corte C) o número de árvores T numa coleção de árvores $\mathcal{T}$ para as quais $C \in \mathcal{S}(T)$; isto é, $w(C) = |\{T \in \mathcal{T} \mid C \in \mathcal{S}(T)\}|$. Seja $\mathcal{S}^*(\mathcal{T}) = \bigcap_{T\in\mathcal{T}} \mathcal{S}(T)$. O valor de uma árvore consenso T em relação a $\mathcal{T}$ é definido da seguinte maneira:
>
> $$val(T, \mathcal{T}) = \begin{cases} -\infty & \text{se } \mathcal{S}^*(\mathcal{T}) \setminus \mathcal{S}(T) \neq \emptyset, \\ -\infty & \text{se } \mathcal{S}(T) \setminus \bigcup_{T\in\mathcal{T}} \mathcal{S}(T) \neq \emptyset, \\ \sum_{C\in\mathcal{S}(T)\setminus\mathcal{S}^*(\mathcal{T})} w(C) & \text{caso contrário.} \end{cases}$$

Em seguida, elas definem a *Árvore Mediana Assimétrica*, denotada por $T_{a.med}$, como sendo a árvore que maximiza $val(T, \mathcal{T})$.

Note que, ao atribuir o valor $-\infty$ para algumas árvores, as primeiras linhas da definição de $val(T, \mathcal{T})$ evitam que estas árvores estejam na "disputa" pelo título de árvore

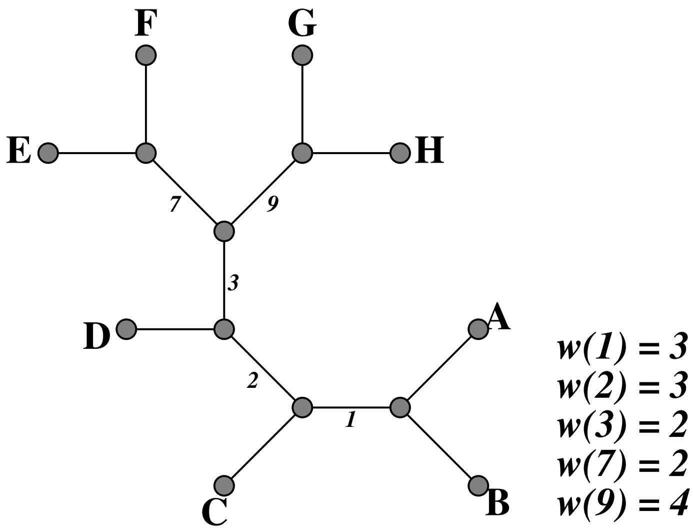

Figura 3.6: Árvore Mediana Assimétrica para a coleção de árvores apresentada na Figura 3.1. Na figura são mostrados os pesos dos cortes não triviais. Qualquer corte trivial tem peso igual a quatro para a coleção, que é formada por quatro árvores.

mediana assimétrica. Claro que isso não é por acaso. A primeira linha força o conjunto de candidatas a mediana assimétrica a possuir apenas árvores que tenham todos os cortes que são comuns a todas as árvores, assim, os cortes que fazem parte do consenso estrito entre as árvores na coleção $\mathcal{T}$ certamente farão parte da mediana assimétrica. A segunda linha, por outro lado, impede que a árvore mediana assimétrica contenha algum corte que não seja encontrado em nenhuma árvore da coleção.

A Figura 3.6 apresenta o exemplo de uma árvore mediana assimétrica para a coleção de árvores apresentada na Figura 3.1. É fácil observar que $val(T, \mathcal{T}) = 10$ para a árvore apresentada. Também é fácil observar que esta é a única árvore mediana assimétrica para a coleção, uma vez que a árvore é completamente resolvida e qualquer troca de um corte presente na árvore por um corte que não está nela representará a troca de um corte com peso maior que um por um corte com peso igual a um, diminuindo portanto o valor da árvore em relação à coleção.

Capítulo 4

Modelos de Evolução

Este capítulo destina-se a apresentar as bases teóricas do cálculo de algumas distâncias usadas nos testes apresentados nos Capítulos 5 e 6.

À medida em que se foi entendendo como funciona a variação de moléculas como o DNA e os peptídeos, também cresceu não só o interesse, como também a necessidade de se modelar matematicamente o processo no qual uma molécula ancestral se transformou naquela que observamos na atualidade. Dois modelos bastante simples, bem como a motivação para os respectivos estudos, são mostrados em suas formas puras nas Seções 4.1 e 4.2, considerando apenas eventos de troca entre os quatro tipos básicos de nucleotídeos, apresentados na Figura 4.3. Na Seção 4.3 são apresentadas formas de aumentar a capacidade de modelos de forma a adaptá-los a novas teorias.

4.1 O Modelo de Evolução de Jukes-Cantor

O modelo de evolução mais simples existente é o modelo proposto por T. H. Jukes e C. Cantor no ano de 1969. Neste modelo, a probabilidade α de um nucleotídeo mudar para outro diferente num intervalo pequeno de tempo Δt é proporcional ao tempo transcorrido. A relação entre α e tamanho do intervalo de tempo é dada por $\alpha = u\Delta t$, onde u é uma constante, válida apenas para intervalos de tempo abaixo de um certo limite.

Seja t_0 o momento que consideramos como sendo o começo de um processo evolutivo qualquer. A probabilidade de uma posição em uma seqüência conter o mesmo tipo de nucleotídeo no intante t_0 e num instante posterior t qualquer é dada pela função $X(t)$ e a probabilidade da mesma posição conter um nucleotídeo diferente no instante t é dada pela função $Y(t)$.

É possível obter as duas funções definindo-se recursivamente tanto $X(t + \Delta t)$ quanto $Y(t + \Delta t)$ em função de $X(t)$ e $Y(t)$. As seções seguintes se dedicam ao estudo dos dois casos separadamente, unindo-os no final para, através de um sistema de equações

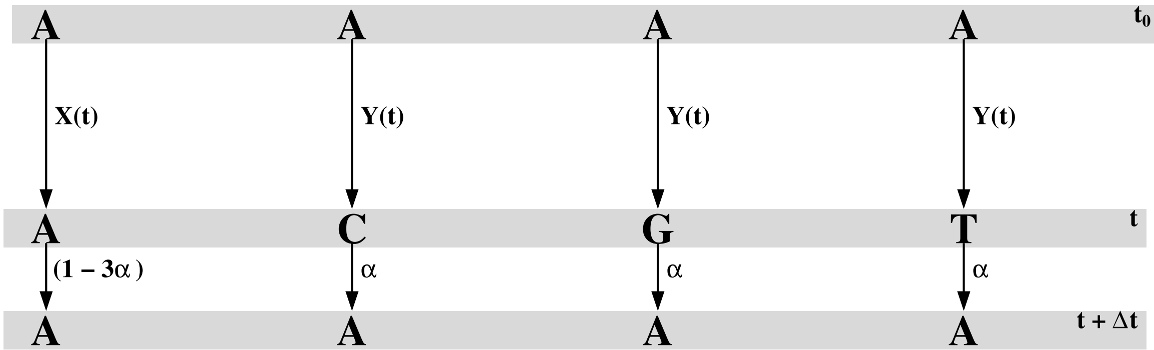

Figura 4.1: Um exemplo ilustrando os quatro cenários possíveis em que o nucleotídeo em determinada posição na seqüência original não difere do nucleotídeo na mesma posição da seqüência observada no tempo $t + \Delta t$ no modelo de Jukes e Cantor.

diferenciais, chegar às funções $X(t)$ e $Y(t)$.

4.1.1 Os Nucleotídeos Não Diferem

Embora os nucleotídeos não sejam diferentes em t_0 e $t + \Delta t$, devemos considerar quatro cenários, como mostra a Figura 4.1:

Cenário 1: O mesmo tipo de nucleotídeo pode ser observado nos três instantes: t_0, t e $t+\Delta t$. A probabilidade deste cenário é dada pela função $P_1(t+\Delta t) = (1-3\alpha)X(t)$.

Cenários 2, 3 e 4: Estes quatro cenários são semelhantes, com exceção do tipo de nucleotídeo que ocupa a posição na seqüência no instante t. Em todos eles o tipo de nucleotídeo encontrado na seqüência na posição em questão é o mesmo tanto no instante t_0 quanto no instante $t + \Delta t$, porém o tipo de nucleotídeo observado em t é diferente. Nestes casos, a probabilidade do cenário é dada por $P_2(t + \Delta t) = P_3(t + \Delta t) = P_4(t + \Delta t) = \alpha Y(t)$.

A probabilidade de um nucleotídeo numa determinada posição ser igual ao nucleotídeo original é dada pela soma das probabilidades dos cenários. Assim:

$$X(t + \Delta t) = (1 - 3\alpha)X(t) + 3\alpha Y(t)$$

Sabemos, porém, que α é determinado por $\alpha = u\Delta t$. Assim:

$$\begin{aligned} X(t + \Delta t) &= X(t) - 3u\Delta t X(t) + 3u\Delta t Y(t) \\ \frac{X(t + \Delta t) - X(t)}{\Delta t} &= -3uX(t) + 3uY(t) \end{aligned}$$

Quando Δt tende a zero, temos a seguinte equação diferencial:

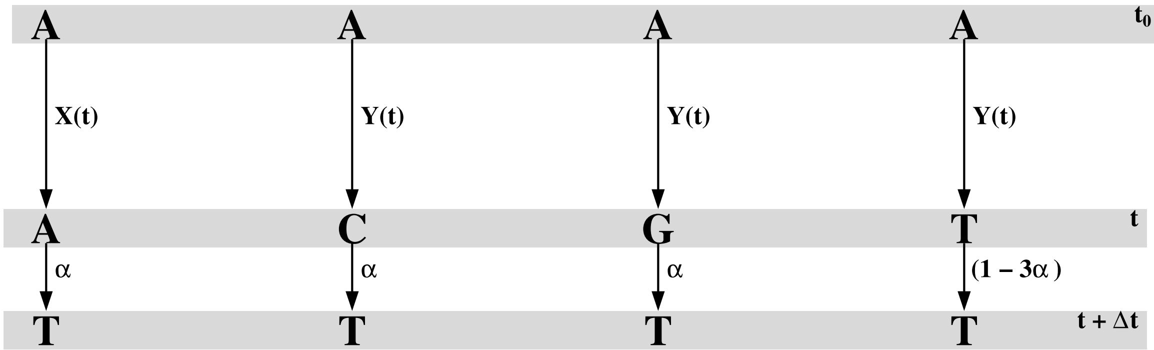

Figura 4.2: Um exemplo ilustrando os quatro cenários possíveis em que o nucleotídeo em determinada posição na seqüência original difere do nucleotídeo na mesma posição da seqüência observada no tempo $t + \Delta t$ no modelo de Jukes e Cantor.

$$\frac{dX(t)}{dt} = -3uX(t) + 3uY(t)$$

4.1.2 Os Nucleotídeos Diferem

Da mesma forma que no caso anterior, embora saibamos aqui que o nucleotídeo presente na posição em questão no tempo $t + \Delta t$ difira do nucleotídeo original, temos novamente quatro cenários, que são apresentados esquematicamente na Figura 4.2:

Cenário 1: Os nucleotídeos observados na posição em questão nos instantes t_0 e t são do mesmo tipo, de modo que uma mutação é observada entre t e $t + \Delta t$. Assim, a probabilidade deste cenário é dada pela função $P_1(t + \Delta t) = \alpha X(t)$.

Cenários 2 e 3: Nos cenários 2 e 3, os nucleotídeos observados nos três instantes são todos diferentes. O que distingue os dois cenários é somente o nucleotídeo observado no instante t. Nestes casos, a probabilidade dos cenários é dada pelas funções $P_2(t + \Delta t) = P_3(t + \Delta t) = \alpha Y(t)$.

Cenário 4: Neste cenário, uma mutação é observada entre os instantes t_0 e t. Os nucleotídeos observados entre t e $t + \Delta t$ são do mesmo tipo. A probabilidade deste cenário é dada por $P_4 = (1 - 3\alpha)Y(t)$.

Como no caso em que os nucleotídeos observados em t_0 e $t + \Delta t$ não diferem, aqui, probabilidade de um nucleotídeo numa determinada posição ser diferente do nucleotídeo original é dada pela soma das probabilidades dos quatro cenários apresentados acima.

$$Y(t + \Delta t) = \alpha X(t) + 2\alpha Y(t) + (1 - 3\alpha)Y(t)$$

Podemos novamente usar a relação $\alpha = u\Delta t$:

$$\begin{aligned} Y(t+\Delta t) &= u\Delta t X(t) + 2u\Delta t Y(t) + (1 - 3u\Delta t)Y(t) \\ Y(t+\Delta t) - Y(t) &= u\Delta t X(t) - u\Delta t Y(t) \\ \frac{Y(t+\Delta t) - Y(t)}{\Delta t} &= uX(t) - uY(t) \end{aligned}$$

Quando Δt tende a zero, temos a seguinte equação diferencial:

$$\frac{dY(t)}{dt} = uX(t) - uY(t)$$

4.1.3 Encontrando as Equações

Como resultado das duas seções anteriores, temos duas equações diferenciais que descrevem as derivadas de primeira ordem de $X(t)$ e $Y(t)$ em termos das próprias funções $X(t)$ e $Y(t)$. Estas duas equações dão origem ao sistema de equações diferenciais de primeira ordem apresentado abaixo:

$$\begin{cases} \frac{dX(t)}{dt} = -3uX(t) + 3uY(t) \\ \frac{dY(t)}{dt} = uX(t) - uY(t) \end{cases}$$

A solução para este sistema de equações obedecendo às condições iniciais $X(t_0) = 1$ e $Y(t_0) = 0$, pode ser encontrada no livro de Graur e Li [13, Capítulo 3].

$$\begin{aligned} X(t) &= \frac{1}{4} + \frac{3}{4}e^{-4u(t-t_0)} \\ Y(t) &= \frac{1}{4} - \frac{1}{4}e^{-4u(t-t_0)} \end{aligned}$$

Como o modelo de Jukes Cantor prevê apenas duas possibilidades para cada posição na seqüência: permanecer inalterada ou sofrer uma mutação, a soma das probabilidades de uma posição permanecer inalterada com as probabilidades das três mutações possíveis deve ser igual a um. De fato, é fácil observar que:

$$X(t) + 3Y(t) = 1$$

4.2 O Modelo de Dois Parâmetros de Kimura

O modelo de Jukes e Cantor é bastante simples e, por conter apenas um parâmetro, também é o mais fácil de ser aplicado, todavia, uma das primeiras constatações que se pode fazer a respeito de mutações em seqüências de nucleotídeos ou aminoácidos é

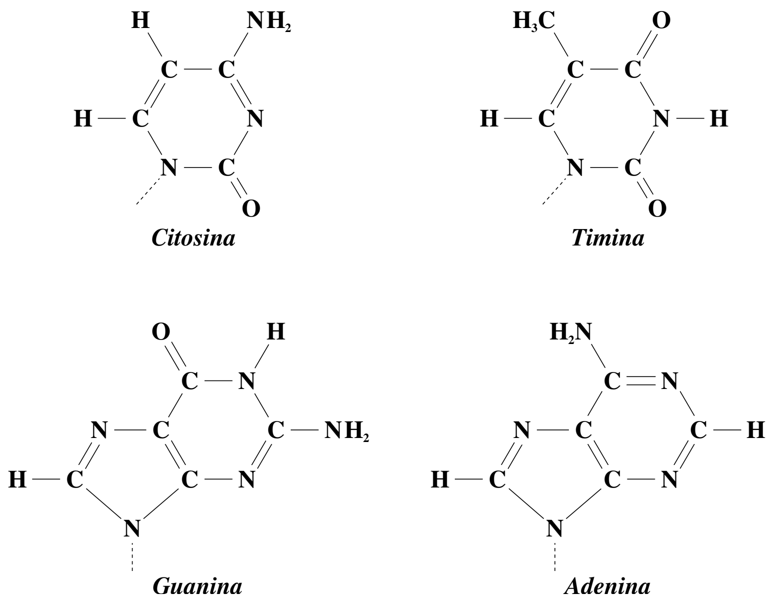

Figura 4.3: As quatro bases nitrogenadas. Nucleotídeos são compostos por um açúcar (uma ribose, no caso do RNA, ou uma desoxirribose, no caso do DNA) unido a um grupo fosfato e uma base nitrogenada. O fosfato e o açúcar têm a função de unir os nucleotídeos formando a estrutura da fita de DNA (ou RNA), enquanto a função de "armazenamento" do código genético propriamente dito fica a cargo das bases nitrogenadas, que, por semelhanças estruturais claras nesta figura, são divididas em dois grupos: as pirimidinas (Citosina e Timina) e as purinas (Adenina e Guanina). A linha tracejada em cada base indica o ponto de ligação com o açúcar.

que certos tipos de mutações parecem ser mais freqüentes que outros. Em seqüências de aminoácidos, esta característica é bem marcante no sentido em que aminoácidos com carga elétrica positiva dificilmente são trocados por aminoácidos com carga elétrica negativa. Da mesma forma, aminoácidos sem carga elétrica dificilmente são trocados por aminoácidos carregados. Em resumo, todos os tipos de substituições são possíveis, mas alguns são notavelmente mais freqüentes que outros.

Os quatro tipos de nucleotídeos que formam as moléculas de DNA podem ser divididos em dois grupos de nucleotídeos com estruturas químicas semelhantes: o das *purinas*, formado pelos nucleotídeos *Adenina* (A) e *Guanina* (G), e o das *pirimidinas*, formado pelos nucleotídeos *Timina* (T) e *Citosina* (C) (Veja a Figura 4.3 para maiores detalhes). Foi observado que, em seqüências de nucleotídeos, um certo tipo de mutação, denominado *transição*, onde ocorre a troca de um nucleotídeo qualquer pelo outro do mesmo grupo, são muito mais favoráveis que o outro tipo de mutação, denominada *transversão*, em que o novo nucleotídeo pertence ao outro grupo.

Em 1980, Kimura propôs um modelo probabilístico para substituições em seqüências de nucleotídeos que leva em consideração diferenças nas taxas de transições e transversões por posição na seqüência de DNA. O modelo de dois parâmetros de Kimura estabelece que a probabilidade de ocorrer uma transição num intervalo de tempo pequeno Δt, em uma determinada posição de uma seqüência de nucleotídeos é α. Assim como no modelo de Jukes e Cantor, no modelo de Kimura α também é proporcional ao tamanho de Δt e a proporcionalidade entre α e Δt é estabelecida pela expressão $\alpha = u\Delta t$. O segundo parâmetro do modelo de Kimura determina a probabilidade de ocorrer uma transversão num intervalo de tempo pequeno Δt. Essa probabilidade tem valor β e tem a proporcionalidade em relação ao tamanho do intervalo determinada pela expressão $\beta = v\Delta t$. Tanto u quanto v são constantes.

A presença de dois parâmetros no modelo de Kimura claramente torna um pouco mais complicada a determinação das equações do modelo, uma vez que no modelo de Kimura, não basta determinar apenas as equações para o caso em que a posição na seqüência permanece inalterada e para o caso em que ela se altera. As alterações agora podem ser de dois tipos e a probabilidade de cada um deles é diferente. Assim, o caso para o qual a posição da seqüência final difere da inicial é dividido em dois: o que a diferença se dá por uma transição e o que a diferença se dá por uma transversão.

Ao final desta seção, chegaremos a fórmulas para três funções:

$X(t)$ determina a probabilidade de o nucleotídeo numa posição de uma seqüência no instante t ser do mesmo tipo que o nucleotídeo encontrado nesta mesma posição no instante t_0.

$Y(t)$ determina a probabilidade de os tipos dos nucleotídeos encontrados numa mesma posição de uma seqüência nos instantes t_0 e t serem diferentes e pertencerem ao mesmo grupo, ou seja, a probabilidade de os nucleotídeos diferirem por um evento de *transição.*

$Z(t)$ determina a probabilidade de os tipos dos nucleotídeos encontrados numa mesma posição de uma seqüência nos instantes t_0 e t serem diferentes e pertencerem a grupos diferentes, ou seja, a probabilidade de os nucleotídeos diferirem por um evento de *transversão.*

Analisando cada uma das situações e seus respectivos cenários, chegaremos a três equações diferenciais que relacionam cada uma das derivadas de primeira ordem em relação a t das equações com as próprias equações.

4.2.1 Os Nucleotídeos Não Diferem

A primeira situação analisada será aquela em que um nucleotídeo em determinada posição na seqüência final não difere do nucleotídeo na mesma posição da seqüência inicial. Quatro

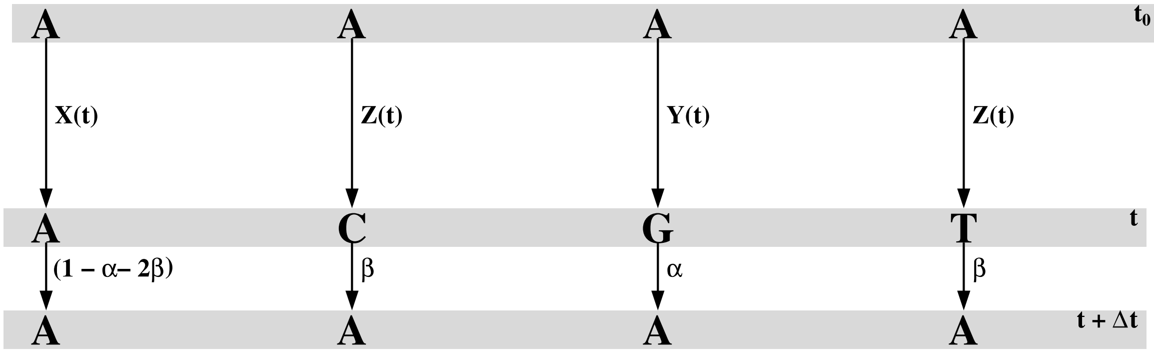

Figura 4.4: Um exemplo ilustrando os quatro cenários possíveis em que o nucleotídeo em determinada posição na seqüência original não difere do nucleotídeo na mesma posição da seqüência observada no tempo $t + \Delta t$ no modelo de dois parâmetros de Kimura.

cenários diferentes podem levar a esta situação:

Cenário 1: Os nucleotídeos observados nos três instantes (t_0, t e $t + \Delta t$) são do mesmo tipo. A probabilidade deste cenário, é dada por $P_1(t + \Delta t) = (1 - \alpha - 2\beta)X(t)$.

Cenário 2: A posição, apesar de ter iniciado com o mesmo tipo de nucleotídeo encontrado na seqüência final, era ocupada por um nucleotídeo de um tipo que não pertence ao mesmo grupo que o nucleotídeo observado no instante t. A probabilidade deste cenário, é dada por $P_2(t + \Delta t) = \beta Z(t)$.

Cenário 3: A posição, apesar de apresentar o mesmo tipo de nucleotídeo da seqüência original, apresentava o outro nucleotídeo do mesmo grupo no instante t, voltando a apresentar um nucleotídeo do mesmo tipo do inicial em $t + \Delta t$. A probabilidade deste cenário, é dada por $P_3(t + \Delta t) = \alpha Y(t)$.

Cenário 4: O cenário 4 é exatamente idêntico ao cenário 2, desta vez considerando o outro tipo de nucleotídeo do grupo ao qual o nucletídeo inicial não pertence. Assim: $P_4(t + \Delta t) = P_2(t + \Delta t) = \beta Z(t)$.

Somando-se todas as probabilidades, chegamos à seguinte equação:

$$X(t + \Delta t) = (1 - \alpha - 2\beta)X(t) + \alpha Y(t) + 2\beta Z(t)$$

Lembrando das equações que relacionam α e β com o tamanho do intervalo de tempo considerado, temos:

$$\begin{aligned}
X(t + \Delta t) &= (1 - u\Delta t - 2v\Delta t)X(t) + u\Delta t Y(t) + 2v\Delta t Z(t) \\
X(t + \Delta t) &= X(t) - (u + 2v)X(t)\Delta t + uY(t)\Delta t + 2vZ(t)\Delta t \\
\frac{X(t + \Delta t) - X(t)}{\Delta t} &= -(u + v)X(t) + uY(t) + 2vZ(t)
\end{aligned}$$

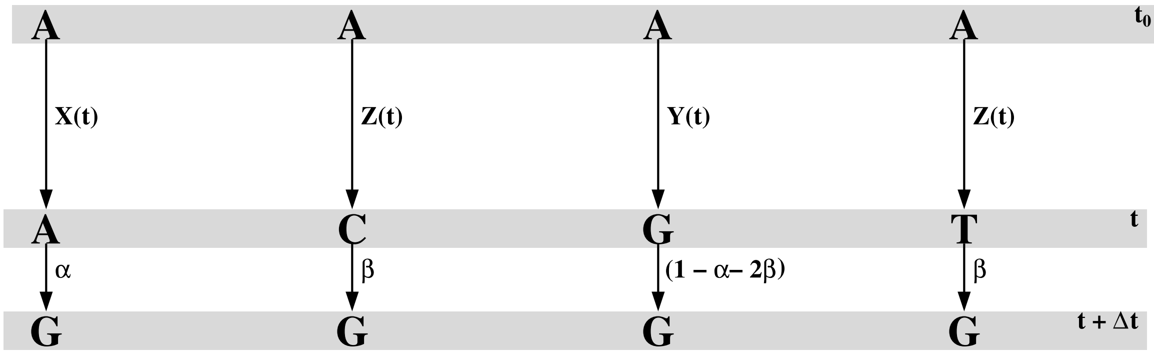

Figura 4.5: Um exemplo ilustrando os quatro cenários possíveis em que o nucleotídeo em determinada posição na seqüência original difere do nucleotídeo na mesma posição da seqüência observada no tempo $t + \Delta t$ por um evento de transição no modelo de dois parâmetros de Kimura.

Quando o tamanho de Δt tende a zero, temos:

$$\frac{dX(t)}{dt} = -(u + 2v)X(t) + uY(t) + 2vZ(t)$$

4.2.2 Os Nucleotídeos Diferem Por Um Evento de Transição

Também para este caso temos que analisar quatro cenários diferentes.

Cenário 1: No instante t, o nucleotídeo observado em determinada posição era do mesmo tipo do nucleotídeo observado na mesma posição no intante t_0. Entre t e Δt notamos uma diferença por um evento de transição. A probabilidade deste cenário é dada por $P_1(t + \Delta t) = \alpha X(t)$.

Cenário 2: Nos instantes t_0 e t, os nucleotídeos numa mesma posição se diferenciam por um evento de transversão, mas outro evento de transversão é observado entre t e $t + \Delta t$, desta vez para o outro nucleotídeo do mesmo grupo do nucleotídeo originalmente na posição em questão. Neste caso, a probabilidade do cenário é dada por $P_2(t + \Delta t) = \beta Z(t)$.

Cenário 3: No cenário 3, a transição é observada já no instante t, sendo que os tipos dos nucleotídeos na posição em questão nos tempos t e $t + \Delta t$ não diferem. Assim: $P_3(t + \Delta t) = (1 - \alpha - 2\beta)Y(t)$.

Cenário 4: Este último cenário é exatamente idêntico ao 2, desta vez considerando o outro tipo de nucleotídeo do grupo ao qual o nucletídeo inicial não pertence como nucleotídeo intermediário. Assim: $P_4(t + \Delta t) = P_2(t + \Delta t) = \beta Z(t)$.

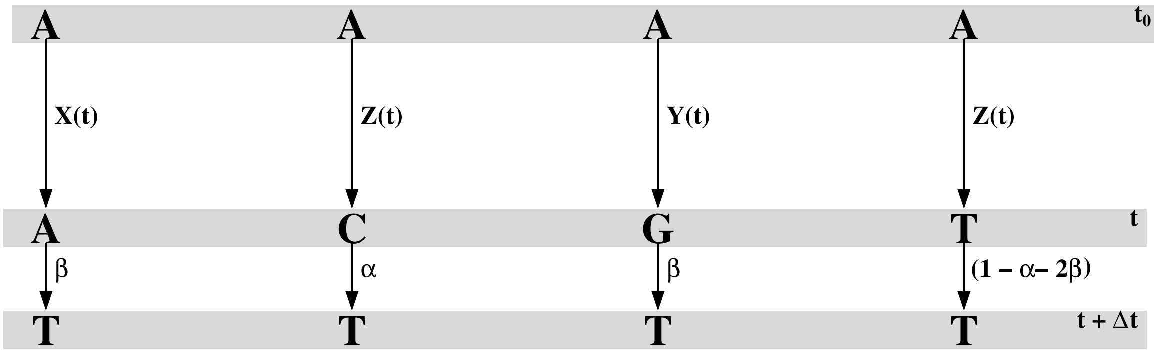

Figura 4.6: Um exemplo ilustrando os quatro cenários possíveis em que o nucleotídeo em determinada posição na seqüência original difere do nucleotídeo na mesma posição da seqüência observada no tempo $t + \Delta t$ por um evento de transversão no modelo de dois parâmetros de Kimura.

A equação procurada é mais uma vez obtida pela soma dos quatro cenários:

$$Y(t + \Delta t) = \alpha X(t) + (1 - \alpha - 2\beta)Y(t) + 2\beta Z(t)$$

Usamos novamente as equações que relacionam α e β com um intervalo de tempo e obtemos:

$$\begin{aligned}
Y(t + \Delta t) &= u\Delta t X(t) + (1 - u\Delta t - 2v\Delta t)Y(t) + 2v\Delta t Z(t) \\
Y(t + \Delta t) &= uX(t)\Delta t + Y(t) - (u + 2v)Y(t)\Delta t + 2vZ(t)\Delta t \\
\frac{Y(t + \Delta t) - Y(t)}{\Delta t} &= uX(t) - (u + 2v)Y(t) + 2vZ(t)
\end{aligned}$$

Mais uma vez, analisamos o caso em que Δt tende a zero.

$$\tfrac{dY(t)}{dt} = uX(t) - (u + 2v)Y(t) + 2vZ(t)$$

4.2.3 Os Nucleotídeos Diferem Por Um Evento de Transversão

Este último caso também possui quatro cenários diferentes a serem analisados. A Figura 4.6 fornece um exemplo de cada um destes cenários.

Cenário 1: Os nucleotídeos observados nos instantes t_0 e t são de um mesmo tipo. O evento de transversão só é observado entre os instantes t e $t + \Delta t$. A probabilidade deste cenário é dada pela função $P_1(t + \Delta t) = \beta X(t)$.

Cenário 2: Após um intervalo de tempo t, os tipos dos nucleotídeos numa mesma posição se diferenciavam por um evento de transversão, mas um evento de transição é observado entre t e $t + \Delta t$. Neste caso, a probabilidade do cenário é dada por $P_2(t + \Delta t) = \alpha Z(t)$.

Cenário 3: Uma transição é observada no tempo t, e a transversão só é observada mesmo entre t e $t + \Delta t$. Deste modo, podemos descrever este cenário através da equação: $P_3(t + \Delta t) = \beta Y(t)$.

Cenário 4: Após um intervalo de tempo t, os nucleotídeos numa mesma posição se diferenciavam por um evento de transversão. Desta vez, porém, não se observa diferença alguma ocorrida na posição em questão entre t e $t+\Delta t$. Neste caso, a probabilidade do cenário é dada por $P_4(t + \Delta t) = (1 - \alpha - 2\beta)Z(t)$.

A última das equações que procuramos também é obtida pela soma dos quatro cenários:

$$Z(t + \Delta t) = \beta X(t) + \alpha Z(t) + (1 - \alpha - 2\beta)Z(t) + \beta Y(t)$$

e pode ser simplificada da seguinte maneira:

$$Z(t + \Delta t) = \beta X(t) + (1 - 2\beta)Z(t) + \beta Y(t)$$

Uma última vez usamos as equações que relacionam α e β com o tamanho do intervalo de tempo e obtemos:

$$\begin{aligned} Z(t + \Delta t) &= v\Delta t X(t) + (1 - 2v\Delta t)Z(t) + v\Delta t Y(t) \\ Z(t + \Delta t) &= vX(t)\Delta t + Z(t) - 2vZ(t)\Delta t + vY(t)\Delta t \\ \frac{Z(t + \Delta t) - Z(t)}{\Delta t} &= vX(t) + vY(t) - 2vZ(t) \end{aligned}$$

Assim como as outras equações, esta também pode ser analisada no caso em que Δt tende a zero.

$$\tfrac{dZ(t)}{dt} = vX(t) + vY(t) - 2vZ(t)$$

4.2.4 Encontrando as Equações

Até agora o que temos são três equações diferenciais que relacionam as derivadas das três funções que queremos descobrir com as próprias funções. Para chegar às funções desejadas, devemos encontrar uma solução para o sistema abaixo:

$$\begin{cases} \frac{dX(t)}{dt} = -(u + 2v)X(t) + uY(t) + 2vZ(t) \\ \frac{dY(t)}{dt} = uX(t) - (u + 2v)Y(t) + 2vZ(t) \\ \frac{dZ(t)}{dt} = vX(t) + vY(t) - 2vZ(t) \end{cases}$$

Graur e Li [13, Capítulo 3] fornecem a seguinte solução para este sistema de equações, com condições iniciais $X(t_0) = 1$, $Y(t_0) = Z(t_0) = 0$:

$$\begin{aligned} X(t) &= \frac{1}{4} + \frac{1}{4}e^{-4v(t-t_0)} + \frac{1}{2}e^{-2(u+v)(t-t_0)} \\ Y(t) &= \frac{1}{4} + \frac{1}{4}e^{-4v(t-t_0)} - \frac{1}{2}e^{-2(u+v)(t-t_0)} \\ Z(t) &= \frac{1}{4} - \frac{1}{4}e^{-4v(t-t_0)} \end{aligned}$$

Como o modelo de dois parâmetros de Kimura prevê apenas três possibilidades para uma posição em uma seqüência: permanecer inalterada, sofrer uma transição ou sofrer uma transversão, sendo que o último caso pode ocorrer em duas variações para qualquer nucleotídeo, podemos verificar facilmente que a expressão

$$X(t) + Y(t) + 2Z(t) = 1$$

é sempre verdadeira para as funções $X(t)$, $Y(t)$ e $Z(t)$ encontradas anteriormente.

4.3 Variação de Parâmetros ao Longo das Seqüências

Modelos de substituição como os de Jukes e Cantor e o de dois parâmetros de Kimura, apesar de bastante simples, são bem fundamentados e se aproximam do modo como uma posição varia numa seqüência. De fato, não seria estranho se a evolução de regiões não funcionais de moléculas de DNA se desse de maneira muito parecida com a prevista pelo modelo de Kimura, por exemplo. Por outro lado, os trechos mais interessantes das moléculas de DNA, assim como as seqüências de aminoácidos que compõem as proteínas, têm funções que dependem do fato de certas posições serem ocupadas por determinado nucleotídeo ou aminoácido, o que torna a mutação em algumas posições da seqüência menos freqüentes que em outras, ou mesmo proibidas.

Quando paramos de focar a evolução de uma posição específica para avaliarmos a evolução da seqüência como um todo, é preciso ter consciência que a velocidade com que posições da seqüência variam ao longo do tempo não é a mesma para todas as posições e esta variação na taxa de evolução deve fazer parte de qualquer modelo que tente explicar a evolução de uma seqüência como um todo.

As seções seguintes resumem dois modelos bastante comuns para a variação da taxa de evolução entre as posições de uma seqüência.

4.3.1 Modelos RAS

O termo *RAS* é uma abreviação de *R*ate *A*cross *S*ites e se aplica a todo modelo que utiliza um modelo de substituição de posições em seqüências, como os apresentados anteriormente, em conjunto com uma função de distribuição. Um modelo RAS associa a cada posição numa seqüência uma coleção de parâmetros para o modelo de substituição escolhido e a variação dos parâmetros atribuídos segue uma função de distribuição qualquer.

Talvez o modelo RAS mais utilizado seja o chamado modelo *K2P*+Γ, que consiste na utilização do modelo de dois parâmetros de Kimura associado a uma distribuição gama de parâmetros ao longo da seqüência.

Uma das limitações dos modelos RAS é o fato de eles não permitirem que os parâmetros de uma posição na seqüência mude ao longo do processo evolutivo. Uma vez estabelecidos os parâmetros de cada posição na seqüência de acordo com a distribuição escolhida, estes parâmetros permanecem fixos. Embora estes modelos se aproximem mais dos casos reais do que os modelos em que as taxas de mutação por posição são iguais, uma vez que mutações ocorrem realmente com mais probabilidade em algumas posições do que em outras, modelos RAS não consideram o fato de que, embora a quantidade de posições variáveis em uma seqüência normalmente permaneça constante, foi observado que as posições da seqüência se revesam na taxa de mutação, hora com algumas posições variando mais, hora com outras, mas normalmente na mesma proporção.

4.3.2 Covarions

Modelos baseados em *covarions* surgiram para suprir a necessidade de simular a variação de taxas de mutação ao longo do processo evolutivo. Modelos utilizando covarions tendem a ser bem mais complexos do que os modelos RAS, que simplesmente substituem os parâmetros fixos dos modelos de evolução por vetores de parâmetros específicos de cada posição.

A idéia da existência dos covarions surgiu em 1967 [17], quando Fitch e Margoliash conseguiram inferir a seqüência de todos os ancestrais comuns a 29 espécies bem distribuídas na escala evolutiva, utilizando para isso a seqüência de uma das sub-unidades do citocromo *c*.

Com as seqüências inferidas em mãos, os pesquisadores puderam identificar qual de três modelos se aproximava mais da realidade: o primeiro dos modelos supunha que todas as posições dos códons eram igualmente variáveis e foi o que menos se aproximou dos dados reais. O segundo modelo supunha que alguns dos códons são invariáveis, mas os outros são igualmente variáveis. Finalmente o terceiro modelo, que mais e aproximou do dados reais, assume que há códons invariáveis, mas que os códons variáveis se divide em duas classes: os *variáveis* e os *hipervariáveis*.

Com base em seu experimento, Fitch e Margoliash, estimaram o número de códons invariáveis em 32, dividindo os outros 81 em um grupo com 65 códons e outro com 16. O primeiro destes dois grupos teve média de substituições de 3,2 e o segundo de 10,1.

Portanto, a taxa de substituições de códons é aproximadamente três vezes maior no grupo hipervariável do que no grupo variável. Claramente, este resultado sustenta a hipótese de que a necessidade de manter características estruturais da proteína não permite que todos os códons variem com a mesma probabilidade.

Em 1970, Fitch e Markowitz conduziram uma análise estatística similar para vários grupos de organismos. Ao excluir grupos de espécies sistematicamente do conjunto inicial de 29 de espécies, se depararam com um fato bastante interessante. O primeiro grupo a ser excluído foi um grupo formado por cinco espécies de fungos. Após a exclusão, a proporção de códons invariáveis estimada foi de aproximadamente 45%. Quando plantas foram excluídas, essa proporção cresceu para aproximadamente 60%. Quando somente mamíferos foram considerados, ela subiu ainda mais.

Fitch e Markowitz chegaram à conclusão de que a proporção de códons invariáveis era inversamente proporcional à distância genética (número de substituições de códons) das espécies mais remotamente relacionadas no grupo utilizado, também conhecida como *amplitude* do conjunto de espécies. Usando uma extrapolação linear, eles estimaram a proporção de códons invariáveis quando somente uma espécie é considerada. Este número ficou em torno de 90%, resultado que sugere que, em qualquer espécie, apenas aproximadamente 10% dos códons do citocromo *c* (cerca de 10 códons) estão sujeitos a mudanças evolucionárias a qualquer momento no curso da evolução. Fitch e Markowitz chamaram estes códons de *concomitantly variable códons* ou *covarions*.

A princípio, a relação entre o aumento no número de posições variáveis e o aumento da amplitude do conjunto de espécies pode parecer estranha. Isso se deve ao fato de haver diferentes grupos de covarions, com cada espécie pertencendo a um deles, e o grupo de covarions representados num grupo de espécies aumentar à medida em que aumenta a amplitude deste grupo.

De fato, Fitch mostrou que os grupos dos fungos e dos metazoários (*Drosophila*, peixe, etc.) têm covarions diferentes. Fitch e Markowitz sugerem que, numa determinada espécie, as substituições de códon são geralmente restritas aos covarions, mas ocasionalmente elas induzem um novo grupo de covarions, destruindo o grupo original. Uma razão possível para esta mudança de grupos de covarions é que uma substituição de aminoácido em alguma posição começa a impor uma restrição de substituição de aminoácidos em outras posições.

Por exemplo, estruturas tridimensionais das ribonucleases (RNases) do rato e do boi são bem conhecidas. RNases de ratos têm os aminoácidos glicina e serina nas posições 38 e 39, respectivamente. A glicina pode mutar para ácido aspártico, mas isto parece ser danoso porque o novo aminoácido reagiria com a lisina na posição 41 e puxaria este resíduo essencial para fora do sítio ativo desta enzima. Da mesma forma, a serina poderia mutar para arginina e não há motivo para que esta mudança não seja aceitável. Nas

RNases bovinas, os grupos são de fato ácido aspártico e arginina, mas a carga positiva da arginina neutraliza a carga negativa do ácido aspártico e provavelmente neutraliza o efeito danoso do ácido aspártico na lisina crítica da posição 41. Se isso for verdade, a substituição da serina pela arginina na posição 39 deve ter precedido a substituição da glicina pelo ácido aspártco na posição 38. Curiosamente, os aminoácidos nas posições 38 e 39 nas RNases suínas são de fato glicina e arginina, respectivamente. Isto ilustra como a posição de um grupo de covarions pode mudar: primeiramente, antes da fixação da arginina, a posição 38 não poderia aceitar o ácido aspártico, enquanto depois da fixação da arginina, o novo ácido aspártico fixado não pode mais ser trocado por um aminoácido neutro. Fitch e Markowitz dão ainda mais exemplos.

O conceito de covarions claramente indica que a taxa de substituição de aminoácidos não é a mesma para todas as posições da seqüência e que em uma determinada posição a taxa deve variar de acordo com o aminoácido na posição com a qual ele interage.

Talvez o modelo mais simples utilizando o conceito de covarions, seja o proposto por David Penny, Bennet McComish, Michael Charleston e Michael Hendy [19], que, utilizando um modelo oculto de Markov, conseguiram simular o comportamento de covarions com apenas dois parâmetros além do modelo de evolução utilizado.

4.4 Simulação do Processo Evolutivo com o Auxílio do Computador

A criação de seqüências artificiais representando uma árvore filogenética se dá através da simulação do processo evolutivo com o auxílio de modelos matemáticos existentes que descrevem o processo de evolução. O procedimento aqui descrito, em particular, é o utilizado pelo software *Seq-Gen* [23], mas não se distancia muito de outros softwares de simulação de seqüências.

Quatro elementos são necessários para a criação de seqüências artificiais por simulação de eventos evolucionários:

Árvore Filogenética Modelo: Com ou sem raiz, mas que tenha valores associados às arestas, determinando a "quantidade de evolução" ocorrida no espaço de tempo entre as duas extremidades das arestas. Normalmente estes valores correspondem à quantidade de eventos de mutação esperada para o espaço de tempo transcorrido entre a diferenciação dos dois seres nas extremidades das arestas.

Seqüência Inicial: A maioria dos softwares simplesmente gera uma seqüência inicial qualquer, às vezes de acordo com parâmetros estabelecidos pelo usuário, como a porcentagem dos nucleotídeos, por exemplo.

Modelo de Substituição: Modelo pelo qual as posições na seqüência variam. Normalmente qualquer modelo semelhante aos apresentados nas Seções 4.1 e 4.2. O *Seq-Gen*, em particular, exige que o modelo determine um processo evolutivo *reversível*, ou seja, que no modelo a probabilidade de uma seqüência qualquer a evoluir para uma outra seqüência b seja a mesma da seqüência b evoluir para a a. Os modelos de substituição de Jukes-Cantor e K2P são dois exemplos de modelos cujos processos são reversíveis [13].

Distribuição de Parâmetros por Posição: Distribuição usada para criar os valores dos parâmetros de cada posição utilizados no modelo de evolução.

A simulação é feita aresta a aresta até que todo o conjunto de arestas da árvore filogenética modelo seja coberto. Para cada aresta é calculada uma matriz de probabilidade de transição P, onde a soma de cada linha é 1 e a probabilidade de uma posição de ter o nucleotídeo i em uma extremidade da aresta e j na outra é dada pelo valor p_{ij}. Os valores em P são obtidos através de fórmulas semelhantes às apresentadas nas Seções 4.1 e 4.2, dependendo do modelo de substituição escolhido.

Em casos em que os parâmetros para o modelo de substituição variam de acordo com a posição na seqüência, geradores de números aleatórios que seguem a distribuição de parâmetros escolhida são usados para determinar estes parâmetros. Neste caso, uma matriz P é calculada para cada posição na seqüência, o que faz com que algumas posições variem mais que outras. Um exemplo deste caso é modelo $K2P + \Gamma$, onde os valores dos parâmetros α e β do modelo de dois parâmetros de Kimura variam de acordo com uma curva de distribuição gama, mantendo no entanto, neste caso, a razão transição/transversão determinada.

Capítulo 5

Uso de Consensos Completamente Resolvidos

Árvores consenso sempre foram vistas como boas ferramentas de comparação entre árvores filogenéticas provavelmente distintas. Por um bom tempo se acreditou que esta era uma das únicas utilidades de consensos entre árvores filogenéticas e, portanto, uma árvore consenso que fosse garantidamente completamente resolvida parecia um luxo desnecessário. Para muitos biólogos, pagar o preço de inserir cortes que não estivessem presentes em todas as árvores, mas sim na maioria delas, parece um ato não só desnecessário como, às vezes, prejudicial. De fato, se queremos olhar para um consenso e observar as semelhanças presentes em um conjunto de árvores, a presença de grupos particulares de certas árvores poderia nos dar uma idéia errada do que realmente é comum a todo o conjunto.

Se as árvores consenso tivessem uma utilidade na qual fizesse sentido perder um pouco da concordância da árvore consenso com alguns elementos do conjunto em prol da construção de uma árvore filogenética completamente resolvida, certamente seria mais fácil encontrar um método de consenso preocupado em criar uma árvore consenso completamente resolvida.

Os testes apresentados neste capítulo foram realizados logo no começo deste trabalho e tiveram como objetivo fornecer uma avaliação inicial da utilização de métodos de consenso como métodos de construção de árvores filogenéticas. A falta de um método de consenso que fornecesse em todos os casos uma árvore completamente resolvida limitou a análise dos resultados. Observamos, no entanto, que quando a construção de uma árvore consenso completamente resolvida foi possível, a árvore consenso estava mais próxima da árvore original que grande parte das árvores utilizadas para construí-la, o que parecia ser um resultado promissor.

Todos os testes foram realizados em um computador com processador Pentium4, da Intel, de 1.500MHz, com cache de 256KB e 256MB de memória. O sistema operacional usado foi o Linux/Fedora Core 2.

Nas seções seguintes são apresentados a descrição dos experimentos, seus resultados e as conclusões a que chegamos a partir deles.

5.1 Conjuntos de Dados

Testes envolvendo métodos de construção de árvores filogenéticas sofrem de um mal comum: a identificação de árvores reconstruídas corretamente esbarra no problema de não conhecermos nenhuma árvore filogenética que tenhamos certeza absoluta de estar correta. Podemos calcular índices que sugerem um grau de confiabilidade de uma árvore [14], mas nunca temos certeza de estar com a verdadeira árvore, aquela que representa a história evolutiva dos seres envolvidos em nossos estudos, em nossas mãos.

Estes testes não foram exceção. Eles tiveram as mesmas limitações que outros testes. Por este motivo, alguns cuidados foram tomados para tentar minimizar possíveis distorções nos resultados. Por exemplo, dois tipos de conjuntos de dados foram escolhidos para participar do mesmo procedimento de teste: um formado por conjuntos de dados obtidos por simulação de processos evolutivos usando uma seqüência ancestral artificial e uma topologia de árvore previamente definida; e um formado por dados reais. Dados artificiais foram obtidos do repositório mantido por Gascuel [12], enquanto os dados reais foram obtidos no repositório *Ribosomal Database Project* [5].

5.1.1 Dados Artificiais

A ausência da árvore filogenética correta é o fato que mais atrapalha os testes de eficiência de métodos de reconstrução de árvores filogenéticas. Neste contexto, não são raros os casos em que dados artificiais são usados para comprovar a eficácia de um método de reconstrução [23].

Embora dados artificiais tenham a vantagem de possuir a "resposta", ou seja, a árvore filogenética original com a qual podemos comparar a árvore reconstruída por um método qualquer, a sua eficácia depende fortemente da qualidade da topologia usada como base para a simulação de processos evolutivos e da proximidade entre o modelo de evolução usado para modificar as seqüências através da arestas da árvore modelo e o processo evolutivo real. O procedimento padrão para a criação de dados artificiais consiste nos seguintes passos:

1. Determinar uma topologia para a árvore original, ou seja, a árvore que será usada na simulação para a criação das seqüências artificiais e que será considerada a árvore correta na comparação dos métodos.

2. Determinar o comprimento das arestas para a topologia escolhida.

3. Determinar o modelo de evolução que será utilizado na criação das seqüências.

4. Usando o modelo de evolução escolhido, simular a evolução de uma seqüência de nucleotídeos com base na topologia determinada no passo 1.

O Passo 2 nos permite "moldar" a topologia de modo que a árvore simule a evolução segundo a hipótese do *relógio molecular*. A hipótese do relógio molecular sugere que há uma espécie de relógio que marca a regularidade com a qual as posições em uma seqüência variam. Segundo esta hipótese, posições em seqüências sofreriam mutações em intervalos praticamente regulares de tempo, o que indica que a diferença entre uma seqüência e seus ancestrais poderia dar indícios do tempo que os separa. Árvores que levam em conta a hipótese do relógio molecular têm uma característica bastante peculiar: como o tempo transcorrido da raiz até as folhas de uma árvore é o mesmo para todas as folhas, a soma dos comprimentos das arestas nos caminhos entre a raiz e cada uma das folhas deveria ser uma constante. Por este motivo, mesmo que a árvore resultante seja sem raiz, é possível encontrar uma raiz para esta árvore de tal modo que a distância desta raiz a qualquer folha seja uma constante.

Outro passo importante é o Passo 3, pois é nele que definimos o modelo usado na simulação do processo evolutivo e a confiança nos resultados dos testes depende diretamente da proximidade entre o processo evolutivo simulado e um processo real. Para este experimento, usamos dois modelos de evolução. Um deles é o modelo resultante da combinação do modelo de dois parâmetros de Kimura para a mutação de posições isoladas da seqüência com a distribuição Γ de taxas de evolução por posição, também conhecido como o modelo *K2P*+Γ. O outro modelo usado é conhecido como *Covarion*. Ambos os modelos foram descritos no Capítulo 4.

Os dados utilizados nestes testes foram obtidos do repositório mantido por Gascuel [12] e consistem de conjuntos de 100 seqüências de 600 nucleotídeos cada. Para evitar conclusões equivocadas derivadas de problemas na escolha da topologia, de cada um destes modelos escolhemos duas árvores: uma que simula a evolução de seqüências segundo a hipótese do relógio molecular e outra que desconsidera esta hipótese. Assim, a matriz de distâncias obtida do primeiro conjunto deve ser ultramétrica [25, Seção 6.5.2], ou estar próxima de uma matriz ultramétrica, enquanto as demais não têm necessariamente esta propriedade.

Em casos em que o método de reconstrução recebe como entrada uma matriz de distâncias, as distâncias foram calculadas com o auxílio do software `dnadist`, do pacote `PHYLIP` [9]. Nestes casos, distâncias foram obtidas pelos dois modelos de evolução descritos no Capítulo 4: o de Jukes e Cantor e o de Kimura de dois parâmetros, além de um terceiro modelo, conhecido como *Tamura-Nei* [27]. Ao final do procedimento de construção de árvores, tínhamos quatro conjuntos distintos de dados:

1. **K2P+Γ:** Neste caso, escolhemos propositalmente, dentre as 5.000 árvores construídas usando o método K2P+Γ, a que mais se afastava de uma árvore que representasse uma evolução tal como a descrita pela hipótese do relógio molecular.

2. **K2P+Γ com Relógio Molecular:** Não havia no conjunto de dados nenhuma árvore que se encaixasse exatamente na hipótese do relógio molecular, mas foi possível escolher uma árvore que se aproximasse de um caso perfeito em que a hipótese se verifica, graças aos parâmetros utilizados na construção da topologia da árvore usada como base na criação das seqüências artificiais.

3. **Covarion:** Através dos parâmetros utilizados na construção da filogenia original, pudemos escolher um conjunto de seqüências originadas pelo modelo evolutivo *covarion* que também não levassem em consideração a hipótese do relógio molecular.

4. **Covarion com Relógio Molecular:** O último dos conjuntos de dados foi criado com base em um conjunto de seqüências derivadas do modelo *covarion* de evolução molecular, utilizando como base uma árvore cujos comprimentos das arestas se aproximava ao de um conjunto de seqüências em que se verifica a hipótese do relógio molecular.

5.1.2 Dados Reais

Para os testes com dados reais, há a necessidade de se escolher um conjunto de dados bem estudado, pois quanto maior o número de estudos feitos sobre a mesma família de seqüências, maiores as chances de se encontrar uma árvore filogenética confiável, que possa ser usada como parâmetro para a comparação de métodos de construção.

Neste caso, como ponto de partida para a construção das árvores, tomamos um conjunto de 76 seqüências selecionadas pela equipe do *Ribosomal Database Project* [5] como um subconjunto representativo de um conjunto de 1.503 seqüências de RNA ribossômico originários da menor sub-unidade de ribossomos mitocondriais de espécies diversas do mundo vivo.

O alinhamento de seqüências disponível no banco de ribossomos para as 76 seqüências escolhidas consistia no mesmo alinhamento feito para o conjunto todo, de forma que várias colunas do alinhamento eram compostas apenas por espaços. Com o auxílio de um script desenvolvido em `Perl`, estas colunas foram eliminadas do alinhamento e, além disso, espécies cujas seqüências alinhadas continham uma quantidade de gaps nas extremidades

Construtor	Software	Detalhes
FMEJ	`fastMe`	*reconstrução por evolução mínima, distâncias por Jukes-Cantor*
FMEK	`fastMe`	*reconstrução por evolução mínima, distâncias por K2P*
MMEJ	`Mega`	*reconstrução por evolução mínima, distâncias por Jukes-Cantor*
MMEK	`Mega`	*reconstrução por evolução mínima, distâncias por K2P*
MMET	`Mega`	*reconstrução por evolução mínima, distâncias por Tamura-Nei*
MMP	`Mega`	*reconstrução por maximização de parcimônia através da troca de vizinhos*
MNJJ	`Mega`	*reconstrução por Neighbor-Joining, distâncias por Jukes-Cantor*
MNJK	`Mega`	*reconstrução por Neighbor-Joining, distâncias por K2P*
MNJT	`Mega`	*reconstrução por Neighbor-Joining, distâncias por Tamura-Nei*
PCO	`dnacomp`	*reconstrução por compatibilidade*
PML	`dnaml`	*reconstrução por máxima verossimilhança*
PMK	`dnamlk`	*reconstrução por máxima verossimilhança com relógio molecular*
PMP	`dnapars`	*reconstrução por maximização de parcimônia*
PNJJ	`neighbor`	*reconstrução por Neighbor-Joining, distâncias por Jukes-Cantor*
PNJK	`neighbor`	*reconstrução por Neighbor-Joining, distâncias por K2P*
PUPJ	`neighbor`	*reconstrução por UPGMA, distâncias por Jukes-Cantor*
PUPK	`neighbor`	*reconstrução por UPGMA, distâncias por K2P*
WNWJ	`weighbor`	*reconstrução por Neighbor-Joining ponderado, distâncias por Jukes-Cantor*
WNWK	`weighbor`	*reconstrução por Neighbor-Joining ponderado, distâncias por K2P*

Tabela 5.1: Construtores usados para criar os conjuntos de árvores filogenéticas usados nos testes de qualidade dos métodos de consenso como métodos de construção de árvores filogenéticas. A coluna "Construtor" apresenta o código pelo qual o construtor será indentificado em outros trechos da dissertação, a coluna "Software" fornece o nome do software correspondente e a coluna "Detalhes" fornece detalhes da execução do software, tais como o modelo de evolução usado para calcular distâncias entre espécies, quando preciso, ou o método usado para reconstrução.

maior que 10% do comprimento total foram descartadas. Ao final deste processo, restaram 72 seqüências. A lista completa das espécies selecionadas se encontra no Apêndice B.

Também neste caso, sempre que foram necessárias, matrizes de distâncias foram calculadas com o auxílio do software `dnadist`, do pacote `PHYLIP`. Assim como para os dados artificiais, as distâncias foram obtidas utilizando os modelos de evolução de Jukes e Cantor, Kimura de dois parâmetros e Tamura-Nei.

5.2 Procedimento

Determinamos um conjunto de pares software+método para a reconstrução de filogenias, mostrado na Tabela 5.1. Nos referiremos a estes pares software+método como *construtores de árvores*, ou simplesmente *construtores*. Mais detalhes sobre os softwares usados nos construtores podem ser encontrados no Apêndice C. Para cada um dos cinco conjuntos

Construtor	DS1	DS2	DS3	DS4	REAIS
FMEJ	1	1	1	1	1
FMEK	1	1	1	1	1
MMEJ	1	1	1	1	1
MMEK	1	1	1	1	1
MMET	1	1	1	1	1
MMP	27	29.625	403	4.479	8
MNJJ	1	1	1	1	1
MNJK	1	1	1	1	1
MNJT	1	1	1	1	1
PCO	100	100	100	100	100
PML	1	1	1	1	0
PMK	0	0	0	0	0
PMP	91	945	151	945	1
PNJJ	1	1	1	1	1
PNJK	1	1	1	1	1
PUPJ	1	1	1	1	1
PUPK	1	1	1	1	1
WNWJ	1	1	1	1	1
WNWK	1	1	1	1	1
TOTAL	233	30.685	669	5.539	123

Tabela 5.2: Número de árvores obtidas de cada conjunto de dados por cada construtor. O valor 0 indica que a execução do construtor não foi bem sucedida. A coluna "Construtor" indica o construtor usado, usando as siglas apresentadas na Tabela 5.1. As demais colunas apresentam o número de árvores obtidas de cada conjunto de dados.

de seqüências, usamos cada um dos construtores, sempre executando os softwares com os parâmetros default. Construtores cuja execução demorou mais que um dia foram interrompidos e resultados de execuções que não foram bem sucedidas foram descartados. Escolhemos uma árvore de cada construtor bem sucedido, tomando a primeira árvore produzida, no caso de construtores que originaram mais de uma árvore. A Tabela 5.2 apresenta o número de árvores produzidas por cada construtor para cada conjunto de dados. Criamos então, para cada conjunto de árvores, uma árvore consenso utilizando o software `consense` do pacote `PHYLIP`.

Um programa foi implementado em `JAVA` para determinar a distância de cortes (definida na página 12) entre duas árvores filogenéticas. Com os consensos em mãos, construímos uma matriz quadrada contendo as distâncias entre cada par de árvores em cada conjunto de dados. Extraímos das tabelas, para cada árvore filogenética T, a menor distância de T a uma outra árvore do conjunto, valor que denominamos *isolamento* de T, e a maior distância de T a uma outra árvore do conjunto, que chamamos de *excentricidade* de T.

5.3 Apresentação e Análise dos Resultados

Esta seção é dedicada à apresentação dos resultados obtidos após a execução do procedimento descrito na Seção 5.2. Para identificar as tabelas de dados de cada caso, usamos aqui a mesma nomenclatura usada para diferenciar os conjuntos de dados na Seção 5.1. As tabelas são apresentadas da seguinte maneira: as colunas e linhas identificadas pelas siglas *"ORIG"* e *"CONS"* correspondem às distâncias em relação à árvore original e à árvore consenso, respectivamente. As demais linhas e colunas são nomeadas de acordo com o construtor utilizado. Construtores são representados pelas siglas apresentadas na Tabela 5.1.

As tabelas também apresentam duas colunas extras, rotuladas por *"EXC"* e *"ISOL"*, que indicam, respectivamente, a excentricidade e o isolamento da árvore em relação ao conjunto contendo as árvores obtidas pelos construtores e a árvore consenso correspondente.

Para cada conjunto de dados artificiais descritos da Seção 5.1.1 e para o conjunto de dados reais descrito na Seção 5.1.2 foram construídas duas matrizes quadradas contendo as distâncias entre as árvores dos conjuntos. A distância calculada entre as árvores é a distância de cortes, definida na página 12. Uma das matrizes continha a distância entre pares considerando todo o conjunto de árvores filogenéticas, que chega a ter mais de 30.000 árvores, dependendo do conjunto de dados. A outra matriz continha distâncias de um conjunto bem menor de árvores, contendo apenas um representante de cada construtor. As tabelas apresentadas nesta seção representam matrizes do segundo tipo. As próximas seções se dedicam à discussão dos resultados obtidos para cada conjunto de dados.

5.3.1 Dados Artificiais Obtidos com o Modelo K2P+Γ

A Tabela 5.3 exibe os dados obtidos pela reconstrução de uma árvore filogenética usando seqüências obtidas artificialmente através da simulação do processo evolutivo usando o modelo de evolução de dois parâmetros de Kimura, descrito na Seção 4.2, com os parâmetros α e β variando entre as posições da seqüência segundo o modelo de distribuição gama. A topologia usada durante a simulação do processo evolutivo não pressupunha a hipótese do relógio molecular.

Algumas características dos dados desta tabela nos chamam a atenção. A primeira delas é que a árvore consenso é a terceira árvore de menor excentricidade no conjunto de dados, juntamente com as árvores obtidas pelos construtores `MMEJ` e `PMP`, perdendo em excentricidade apenas para as árvores obtidas pelos construtores `FMEJ` e `FMEK`. Além disso, é a sexta árvore mais próxima da árvore original, perdendo apenas para as árvores `MMEJ`, `MMP`, `PML` , `PMP` e `WNWK`.

A árvore criada pelo construtor `PCO` chama a atenção pela sua distância a todas as

	ORIG	CONS	FMEJ	FMEK	MMEJ	MMEK	MMET	MMP	MNJJ	MNJK	MNJT	PCO	PML	PMP	PNJJ	PNJK	PUPJ	PUPK	WNWJ	WNWK	EXC	ISOL
ORIG	0	48	58	56	46	58	54	44	52	56	52	148	38	42	54	56	112	114	50	42	148	38
CONS	48	0	26	26	30	34	34	46	28	18	26	144	44	46	16	18	100	102	22	22	144	16
FMEJ	58	26	0	20	48	54	54	50	46	32	46	140	48	50	30	32	100	102	36	36	140	20
FMEK	56	26	20	0	46	52	52	46	36	24	36	142	44	46	26	24	102	104	30	32	142	20
MMEJ	46	30	48	46	0	20	16	58	36	40	36	144	54	60	40	40	110	112	36	44	144	16
MMEK	58	34	54	52	20	0	4	68	32	48	28	148	60	66	48	48	116	118	44	48	148	4
MMET	54	34	54	52	16	4	0	64	32	48	28	148	60	62	48	48	116	118	46	50	148	4
MMP	44	46	50	46	58	68	64	0	58	52	56	146	30	8	50	52	112	114	50	44	146	8
MNJJ	52	28	46	36	36	32	32	58	0	30	8	148	54	56	32	30	112	114	44	46	148	8
MNJK	56	18	32	24	40	48	48	52	30	0	34	148	54	52	4	0	96	98	34	40	148	4
MNJT	52	26	46	36	36	28	28	56	8	34	0	148	50	54	36	34	114	116	42	44	148	8
PCO	148	144	140	142	144	148	148	146	148	148	148	0	146	144	148	148	162	164	146	146	164	140
PML	38	44	48	44	54	60	60	30	54	54	50	146	0	28	52	54	110	112	46	40	146	28
PMP	42	46	50	46	60	66	62	8	56	52	54	144	28	0	50	52	112	114	50	44	144	8
PNJJ	54	16	30	26	40	48	48	50	32	4	36	148	52	50	0	4	98	100	32	38	148	4
PNJK	56	18	32	24	40	48	48	52	30	0	34	148	54	52	4	0	96	98	34	40	148	4
PUPJ	112	100	100	102	110	116	116	112	112	96	114	162	110	112	98	96	0	6	100	100	162	6
PUPK	114	102	102	104	112	118	118	114	114	98	116	164	112	114	100	98	6	0	102	102	164	6
WNWJ	50	22	36	30	36	44	46	50	44	34	42	146	46	50	32	34	100	102	0	12	146	12
WNWK	42	22	36	32	44	48	50	44	46	40	44	146	40	44	38	40	100	102	12	0	146	12

Excentricidade Mínima: 140 / Isolamento Mínimo: 4

Tabela 5.3: Resultados dos testes com árvores construídas a partir de seqüências artificiais criadas utilizando o modelo de evolução de dois parâmetros de Kimura associado a uma distribuição Γ de taxas de mutação por posição na seqüência. As siglas "ORIG" e "CONS" indicam as linhas e colunas onde estão apresentadas as distâncias das árvores original e consenso, respectivamente. As demais linhas e colunas representam as distâncias em relação às árvores produzidas pelos construtores correspondentes na Tabela 5.1.

demais árvores. De fato, de todas as árvore da coleção, ela é uma das que possui a maior excentricidade e, de longe, o maior isolamento. O fato de ela ser a árvore que mais se distancia da árvore original indica que, na verdade, o método utilizado para construi-la não é um bom método para a reconstrução de árvores com base em seqüências que evoluem segundo o modelo utilizado na criação deste conjunto de seqüências.

As árvores usadas para obter os valores apresentados na Tabela 5.4 foram construídas usando softwares de reconstrução de árvores filogenéticas, tendo como entrada seqüências sintetizadas usando o mesmo modelo usado na construção do conjunto de dados apresentado na Tabela 5.3. A diferença entre os dois conjuntos de dados está no fato de que a topologia usada como base para a simulação do processo evolutivo neste conjunto se aproximava da topologia de uma árvore em que a hipótese do relógio molecular é observada.

O conjunto de dados gerados pelo modelo K2P+Γ sobre uma árvore que obedece a hipótese do relógio molecular, na verdade, se demonstrou de todo ruim. Os valores obtidos pela comparação com a árvore consenso, por exemplo, se encontram propositadamente omitidos da tabela devido ao fato de a árvore consenso obtida através do software `consense` não ser uma árvore completamente resolvida. É impossível comparar distâncias entre árvores completamente resolvidas com distâncias entre árvores parcialmente resolvidas, ao menos usando a métrica de cortes, porque o número de cortes de uma árvore parcialmente resolvida é menor que o número de cortes de uma árvore completamente resolvida, assim, as distâncias medidas entre pares em que ao menos uma das árvores é parcialmente resolvida tendem a ser consideravelmente menores que as distâncias obtidas entre pares de árvores completamente resolvidas.

Além disso, as distâncias observadas na Tabela 5.4 são extremamente grandes. Aproximadamente 57% das distâncias apresentadas nesta tabela são maiores ou iguais a 100, o que indica uma diferença de pelo menos 50 cortes entre as duas árvores, lembrando que o valor máximo para a distância entre duas árvores completamente resolvidas com 100 taxa é de 194, ou seja, uma diferença de 97 cortes.

Como se não bastasse, o isolamento da árvore original em relação ao conjunto é 106, o que significa que a árvore original difere em no mínimo 53 cortes de qualquer outra árvore do conjunto, o que correponde a mais de 50% da distância máxima. Por estes motivos, este conjunto não é um bom indicador da utilidade da árvore consenso como "construtora" de árvores filogenétcias.

5.3.2 Dados Artificiais Obtidos com o Modelo Covarion

As distâncias apresentadas na Tabela 5.5 foram obtidas comparando-se árvores para as quais as seqüências usadas na construção eram seqüências artificiais criadas pela simulação de um processo evolutivo usando o modelo de evolução conhecido como *covarion*, descrito

	ORIG	**FMEJ**	**FMEK**	**MMEJ**	**MMEK**	**MMET**	**MMP**	**MNJJ**	**MNJK**	**MNJT**	**PCO**	**PML**	**PMP**	**PNJJ**	**PNJK**	**PUPJ**	**PUPK**	**WNWJ**	**WNWK**	**EXP**	**ISOL**
ORIG	0	116	116	116	116	114	114	122	122	120	124	106	112	118	116	136	136	116	114	136	106
FMEJ	116	0	42	92	90	96	108	84	78	82	122	108	100	58	60	136	134	80	76	136	42
FMEK	116	42	0	88	86	94	110	76	78	80	124	106	102	58	58	138	136	88	84	138	42
MMEJ	116	92	88	0	40	42	112	78	80	86	130	112	108	82	82	140	138	80	82	140	40
MMEK	116	90	86	40	0	48	112	70	72	78	132	114	108	80	80	142	140	82	84	142	40
MMET	114	96	94	42	48	0	116	82	84	84	130	112	108	82	82	140	138	82	82	140	42
MMP	114	108	110	112	112	116	0	112	108	116	132	114	110	108	106	148	148	108	106	148	106
MNJJ	122	84	76	78	70	82	112	0	38	54	132	112	108	66	66	146	146	92	92	146	38
MNJK	122	78	78	80	72	84	108	38	0	52	132	112	108	66	66	146	146	90	90	146	38
MNJT	120	82	80	86	78	84	116	54	52	0	130	108	106	74	74	146	146	92	90	146	52
PCO	124	122	124	130	132	130	132	132	132	130	0	122	86	122	122	140	140	116	116	140	86
PML	106	108	106	112	114	112	114	112	112	108	122	0	104	110	108	138	138	102	98	138	98
PMP	112	100	102	108	108	108	110	108	108	106	86	104	0	98	98	134	134	92	94	134	86
PNJJ	118	58	58	82	80	82	108	66	66	74	122	110	98	0	30	134	134	72	68	134	30
PNJK	116	60	58	82	80	82	106	66	66	74	122	108	98	30	0	132	132	72	64	132	30
PUPJ	136	136	138	140	142	140	148	146	146	146	140	138	134	134	132	0	4	132	128	148	4
PUPK	136	134	136	138	140	138	148	146	146	146	140	138	134	134	132	4	0	130	126	148	4
WNWJ	116	80	88	80	82	82	108	92	90	92	116	102	92	72	72	132	130	0	28	132	28
WNWK	114	76	84	82	84	82	106	92	90	90	116	98	94	68	64	128	126	28	0	128	28

Excentricidade Mínima: 128 / Isolamento Mínimo: 4

Tabela 5.4: Resultados dos testes com árvores construídas a partir de seqüências artificiais criadas utilizando o modelo de evolução de dois parâmetros de Kimura associado a uma distribuição Γ de taxas de mutação por posição na seqüência e supondo a existência de um relógio molecular. A sigla "ORIG" indica a linha e a coluna onde estão apresentadas as distâncias relativas à árvore original. As demais linhas e colunas representam as distâncias em relação às árvores produzidas pelos construtores correspondentes na Tabela 5.1.

na Seção 4.3, em uma árvore cuja topologia se distanciava de uma árvore típica na qual se observa a hipótese do relógio molecular.

De um modo geral, as mesmas observações feitas para os dados na Tabela 5.3 podem ser feitas para este conjunto de distâncias. Note que aqui o desempenho da árvore consenso é melhor que no primeiro caso, pois a árvore consenso é a terceira árvore mais próxima da árvore original, perdendo apenas para as árvores criadas pelos construtores `PML` e `WNWK` e impatando com as árvores criadas pelos construtores `FMEJ` e `PMP`.

Um exame rápido mostra que todas as excentricidades estão próximas da diferença máxima entre duas árvores e todos os valores de excentricidade são elevados por causa da presença da árvore `PCO` na coleção. É espantoso observar que a menor das excentricidades corresponde a aproximadamente 87% da distância máxima entre duas árvores filogenéticas completamente resolvidas com 100 taxa.

Por outro lado, a principal observação feita na Seção 5.3.1 também é válida para este conjunto de dados. Embora com os valores de excentricidade um pouco distorcidos devido à presença de três árvores problemáticas, as árvores que mais se aproximam da árvore original têm excentricidades com valores que variam entre a excentricidade da árvore consenso e a excentricidade da árvore original.

Da mesma forma que a Tabela 5.5 se relaciona com a Tabela 5.3, a Tabela 5.6 se relaciona com a Tabela 5.4. Também aqui a única diferença em relação à tabela anterior é o fato de a topologia usada para a criação das seqüências se aproximar de uma topologia que sustenta a hipótese do relógio molecular.

As distâncias obtidas por comparações com a árvore consenso do grupo não puderam ser apresentadas novamente devido ao fato de a árvore consenso gerada pelo software `consense` não ser uma árvore filogenética completamente resolvida. Embora neste caso a porcentagem pares de árvores com distância acima de 100 ser bem menor, em torno de 45%, o número ainda é demasiadamente grande e a excentricidade da árvore original não deixa dúvidas de que este conjunto de árvores, de um modo geral, não é um bom conjunto.

5.3.3 Dados Reais: RNA Ribossômico Obtido da Menor Sub-Unidade

Os dados apresentados na Tabela 5.7 foram obtidos de árvores construídas com seqüências de RNA da sub-unidade menor de ribossomos de 72 espécies representativas do Projeto de Banco de Dados Ribossômico, mantido pelo Centro de Ecologia Microbiológica da Universidade Estadual de Michigan. A proximidade dos valores com os das Tabelas 5.3 e 5.5 não significa que as árvores têm qualidades semelhantes. É preciso lembrar que estas árvores foram construídas com 72 seqüências, e não com 100, como as demais. Assim, a

	ORIG	**CONS**	**FMEJ**	**FMEK**	**MMEJ**	**MMEK**	**MMET**	**MMP**	**MNJJ**	**MNJK**	**MNJT**	**PCO**	**PML**	**PMP**	**PNJJ**	**PNJK**	**PUPJ**	**PUPK**	**WNWJ**	**WNWK**	**EXC**	**ISOL**
ORIG	0	86	86	88	90	94	92	88	90	100	92	174	68	86	92	96	124	124	92	84	174	68
CONS	86	0	36	40	24	30	20	64	24	36	18	172	72	66	26	22	122	122	36	32	172	18
FMEJ	86	36	0	4	52	60	50	76	52	62	48	172	74	62	52	42	122	122	40	30	172	4
FMEK	88	40	4	0	56	64	54	80	54	60	50	172	74	62	54	42	124	124	44	34	172	4
MMEJ	90	24	52	56	0	26	18	80	26	50	36	176	78	76	24	32	120	120	38	34	176	18
MMEK	94	30	60	64	26	0	14	82	46	44	44	174	76	72	48	50	124	124	58	50	174	14
MMET	92	20	50	54	18	14	0	78	36	46	34	174	74	70	38	40	122	122	48	40	174	14
MMP	88	64	76	80	80	82	78	0	74	76	70	174	80	78	74	76	120	120	70	68	174	64
MNJJ	90	24	52	54	26	46	36	74	0	42	24	176	80	78	2	30	120	120	40	40	176	2
MNJK	100	36	62	60	50	44	46	76	42	0	18	172	82	78	44	38	124	124	50	52	172	18
MNJT	92	18	48	50	36	44	34	70	24	18	0	172	78	76	26	24	122	122	36	38	172	18
PCO	174	172	172	172	176	174	174	174	176	172	172	0	172	172	176	172	178	178	174	174	178	172
PML	68	72	74	74	78	76	74	80	80	82	78	172	0	64	80	76	132	132	78	68	172	64
PMP	86	66	62	62	76	72	70	78	78	78	76	172	64	0	78	74	128	128	72	66	172	62
PNJJ	92	26	52	54	24	48	38	74	2	44	26	176	80	78	0	28	120	120	40	40	176	2
PNJK	96	22	42	42	32	50	40	76	30	38	24	172	76	74	28	0	126	126	32	36	172	22
PUPJ	124	122	122	124	120	124	122	120	120	124	122	178	132	128	120	126	0	10	118	122	178	10
PUPK	124	122	122	124	120	124	122	120	120	124	122	178	132	128	120	126	10	0	118	122	178	10
WNWJ	92	36	40	44	38	58	48	70	40	50	36	174	78	72	40	32	118	118	0	22	174	22
WNWK	84	32	30	34	34	50	40	68	40	52	38	174	68	66	40	36	122	122	22	0	174	22

Excentricidade Mínima: 172 / Isolamento Mínimo: 2

Tabela 5.5: Resultados dos testes com árvores construídas a partir de seqüências artificiais criadas utilizando o conceito de covarions. As siglas "ORIG" e "CONS" indicam as linhas e colunas onde estão apresentadas as distâncias das árvores original e consenso, respectivamente. As demais linhas e colunas representam as distâncias em relação às árvores produzidas pelos construtores correspondentes na Tabela 5.1.

	ORIG	FMEJ	FMEK	MMEJ	MMEK	MMET	MMP	MNJJ	MNJK	MNJT	PCO	PML	PMP	PNJJ	PNJK	PUPJ	PUPK	WNWJ	WNWK	EXC	ISOL
ORIG	0	110	108	114	114	114	88	114	116	114	128	98	100	116	118	118	118	100	102	128	88
FMEJ	110	0	22	72	70	68	94	70	68	72	110	92	96	72	66	122	122	82	80	122	22
FMEK	108	22	0	78	76	74	94	74	70	76	112	94	96	74	64	122	122	82	76	122	22
MMEJ	114	72	78	0	30	30	98	74	74	70	112	94	98	74	70	128	128	78	84	128	30
MMEK	114	70	76	30	0	12	98	72	72	70	114	94	100	74	70	128	128	78	84	128	12
MMET	114	68	74	30	12	0	98	72	72	70	114	94	100	74	70	128	128	78	84	128	12
MMP	88	94	94	98	98	98	0	104	104	102	120	88	88	104	106	126	126	86	92	126	86
MNJJ	114	70	74	74	72	72	104	0	18	22	122	100	104	34	48	128	128	74	74	128	18
MNJK	116	68	70	74	72	72	104	18	0	26	122	100	104	34	46	128	128	74	74	128	18
MNJT	114	72	76	70	70	70	102	22	26	0	124	98	100	42	52	130	130	74	74	130	22
PCO	128	110	112	112	114	114	120	122	122	124	0	106	84	120	122	134	134	120	126	134	84
PML	98	92	94	94	94	94	88	100	100	98	106	0	84	98	100	122	122	92	90	122	84
PMP	100	96	96	98	100	100	88	104	104	100	84	84	0	100	104	118	118	94	96	118	84
PNJJ	116	72	74	74	74	74	104	34	34	42	120	98	100	0	40	114	114	70	70	120	34
PNJK	118	66	64	70	70	70	106	48	46	52	122	100	104	40	0	128	128	74	72	128	40
PUPJ	118	122	122	128	128	128	126	128	128	130	134	122	118	114	128	0	0	118	116	134	114
PUPK	118	122	122	128	128	128	126	128	128	130	134	122	118	114	128	0	0	118	116	134	114
WNWJ	100	82	82	78	78	78	86	74	74	74	120	92	94	70	74	118	118	0	32	120	32
WNWK	102	80	76	84	84	84	92	74	74	74	126	90	96	70	72	116	116	32	0	126	32

Excentricidade Mínima: 118 / Isolamento Mínimo: 12

Tabela 5.6: Resultados dos testes com árvores construídas a partir de seqüências artificiais criadas utilizando o conceito de covarions e supondo a existência de um relógio molecular. A sigla “ORIG” indica a linha e a coluna onde estão apresentadas as distâncias da árvore original. As demais linhas e colunas representam as distâncias em relação às árvores produzidas pelos construtores correspondentes na Tabela 5.1. Não foi possível obter um consenso completamente resolvido usando o mesmo software usado nos outros casos.

	ORIG	**FMEJ**	**FMEK**	**MMEJ**	**MMEK**	**MMET**	**MMP**	**MNJJ**	**MNJK**	**MNJT**	**PCO**	**PMP**	**PNJJ**	**PNJK**	**PUPJ**	**PUPK**	**WNWJ**	**WNWK**	**EXC**	**ISOL**
ORIG	0	88	90	110	110	106	106	106	104	106	130	94	94	90	98	96	96	92	130	88
FMEJ	88	0	24	82	82	82	94	86	82	84	124	68	38	44	64	66	42	54	124	24
FMEK	90	24	0	84	84	84	98	88	84	86	124	72	54	38	72	72	56	56	124	24
MMEJ	110	82	84	0	2	32	80	34	38	42	126	88	92	94	84	84	86	92	126	2
MMEK	110	82	84	2	0	34	80	36	40	44	126	88	92	94	84	84	86	92	126	2
MMET	106	82	84	32	34	0	84	50	48	44	124	88	90	94	86	86	92	90	124	32
MMP	106	94	98	80	80	84	0	72	76	78	126	100	92	98	94	94	98	100	126	72
MNJJ	106	86	88	34	36	50	72	0	16	20	126	86	92	96	86	86	92	94	126	16
MNJK	104	82	84	38	40	48	76	16	0	10	126	84	90	90	88	88	86	86	126	10
MNJT	106	84	86	42	44	44	78	20	10	0	126	84	90	92	90	90	88	88	126	10
PCO	130	124	124	126	126	124	126	126	126	126	0	120	126	126	126	126	126	126	130	120
PMP	94	68	72	88	88	88	100	86	84	84	120	0	84	80	80	82	76	80	120	68
PNJJ	94	38	54	92	92	90	92	92	90	90	126	84	0	32	70	70	54	50	126	32
PNJK	90	44	38	94	94	94	98	96	90	92	126	80	32	0	76	74	62	54	126	32
PUPJ	98	64	72	84	84	86	94	86	88	90	126	80	70	76	0	22	64	66	126	22
PUPK	96	66	72	84	84	86	94	86	88	90	126	82	70	74	22	0	66	64	126	22
WNWJ	96	42	56	86	86	92	98	92	86	88	126	76	54	62	64	66	0	34	126	34
WNWK	92	54	56	92	92	90	100	94	86	88	126	80	50	54	66	64	34	0	126	34

Excentricidade Mínima: 120 / Isolamento Mínimo: 2

Tabela 5.7: Resultados dos testes com árvores construídas a partir de seqüências reais de RNA ribossômico extraídas da sub-unidade menor. A sigla "ORIG" indica a linha e coluna onde estão apresentadas as distâncias da árvore original. As demais linhas e colunas representam as distâncias em relação às árvores produzidas pelos construtores correspondentes na Tabela 5.1. Não foi possível obter um consenso completamente resolvido usando o mesmo software usado nos outros casos.

distância máxima entre duas árvores passa a ser 138, e não mais 194.

Para o conjunto formado por seqüências reais, assim como para os conjuntos artificiais que simulavam a hipótese do relógio molecular, o método de consenso utilizado pelo software `consense` não foi capaz de gerar uma árvore consenso completamente resolvida. Isto pode ser explicado pelas distâncias apresentadas nas Tabelas 5.4, 5.6 e 5.7 e pela forma como o método de consenso implementado no software `consense` cria a árvore consenso. O software `consense` utiliza uma extensão do método da Regra da Maioria, apresentado na Seção 3.4, na qual, após a inclusão dos cortes presentes em mais de 50% das árvores, o software continua a incluir cortes compatíveis com os já presentes na árvore, em ordem decrescente de freqüência na coleção de árvores, enquanto ainda houver cortes a serem adicionados e a árvore consenso não for completamente resolvida. Note que quanto mais distintas forem as árvores, menos cortes aparecerão em mais de 50% delas e muito mais cortes serão deixados para a segunda fase do método, o que torna a chance de a árvore consenso ser completamente resolvida bem menor.

Fizemos uma tentativa de construir consensos completamente resolvidos eliminando árvores da coleção utilizada inicialmente para os três conjuntos em que tal árvore não foi conseguida na primeira tentativa. Eliminamos as árvores cuja distância de cortes em relação à árvore de referência era maior que um determinado limite. Para os dados artificiais, não houve limite capaz de reduzir o conjunto original de árvores filogenéticas a um conjunto cujo consenso gerado pelo software `consense` fosse completamente resolvido.

Para os dados reais, o limite para o qual uma árvore completamente resolvida foi encontrada foi 95. Selecionamos então o conjunto de árvores cuja distância à árvore de referência era menor que 95 e construímos um consenso com estas árvores, obtendo as distâncias mostradas na Tabela 5.8, que mostram um bom desempenho da árvore consenso.

5.4 Conclusões

A princípio os resultados podem parecer inconclusivos, devido ao fato de, para grande parte dos conjuntos, a árvore consenso não ser completamente resolvida e a distância das árvores consenso não poder ser comparada com as distâncias das árvores construídas a partir dos conjuntos de dados à árvore de referência de cada conjunto.

Por outro lado, sempre que isso pôde ser feito, nos conjuntos de árvores construídas a partir de dados artificiais que não levam em conta o relógio molecular e no conjunto reduzido de árvores construídas a partir de seqüências reais, a árvore consenso é uma das que mais se aproxima da árvore de referência do conjunto:

- No conjunto de dados simulados a partir do modelo K2P+Γ, a árvore consenso

	ORIG	CONS	FMEJ	FMEK	PMP	PNJJ	PNJK	WNWK	EXC	ISOL
ORIG	0	88	88	90	94	94	90	92	94	88
CONS	88	0	18	26	74	32	32	38	88	18
FMEJ	88	18	0	24	68	38	44	54	88	18
FMEK	90	26	24	0	72	54	38	56	90	24
PMP	94	74	68	72	0	84	80	80	94	68
PNJJ	94	32	38	54	84	0	32	50	94	32
PNJK	90	32	44	38	80	32	0	54	90	32
WNWK	92	38	54	56	80	50	54	0	92	38

Excentricidade Mínima: 88 / Isolamento Mínimo: 18

Tabela 5.8: Resultados dos testes com árvores construídas a partir de seqüências de RNA ribossômico extraídas da sub-unidade menor. Esta tabela foi construída a partir de um sub-conjunto da Tabela 5.7, considerando apenas árvores cuja distância de cortes em relação à árvore de referência não supera 95 cortes.

tinha a sexta menor distância da árvore referência, ficando atrás apenas de 5 das 18 árvores do conjunto;

- No conjunto de dados simulados a partir do modelo baseado em covarions, a árvore consenso tinha a terceira menor distância da árvore referência, ficando atrás apenas de 2 das 18 árvores do conjunto;

- No conjunto reduzido de seqüências reais, a árvore consenso tinha a menor distância de cortes em relação à árvore referência.

De uma forma geral, isso nos mostra que, num conjunto de árvores de boa qualidade, ou seja, de árvores próximas da árvore verdadeira, quando não sabemos qual das árvores de conjunto está mais próxima da árvore que procuramos, um consenso completamente resolvido construído com cortes mais freqüentes da coleção normalmente é uma opção mais segura do que simplesmente escolher uma das árvores da coleção como a melhor. Nos três casos em que uma árvore consenso completamente resolvida pode ser construída, escolher a árvore significava garantidamente estar escolhendo uma árvore melhor que no mínimo 72% das árvores do conjunto.

O fato do software `consense` não ter conseguido construir uma árvore consenso completamente resolvida para dois dos conjuntos de árvores construídas a partir de seqüências artificiais e para o conjunto maior de árvores construídas a partir de seqüências reais mostra que tais árvores dificilmente são obtidas se o método de consenso não se preocupa em procurar por árvores completamente resolvidas. Percebemos nestes casos a utilidade e a carência de um método de consenso capaz de unir os vários resultados obtidos em apenas uma árvore sem, no entanto, causar a perda de resolução observada normalmente nas árvores construídas pelos métodos de consenso mais comuns.

O próximo capítulo se dedica à apresentação de um método de consenso que produz árvores consenso completamente resolvidas a partir de uma coleção de árvores consenso também completamente resolvidas.

Capítulo 6

A Árvore Mais Provável

Este capítulo é dedicado à definição de uma espécie de árvore consenso completamente resolvida e sem raiz denominada *Árvore mais provável*, definida para coleções de árvores filogenéticas completamente resolvidas e sem raiz que possuam exatamente o mesmo conjunto de folhas.

Seja L um conjunto de unidades taxonômicas e $\mathcal{T}$ uma coleção não vazia qualquer de árvores filogenéticas pertencentes a $\mathcal{T}_U^*(L)$. Seja

$$\mathcal{S}(\mathcal{T}) = \bigcup_{T \in \mathcal{T}} \mathcal{S}(T)$$

o conjunto de todos os cortes encontrados na coleção $\mathcal{T}$ e seja

$$p(C, \mathcal{T}) = \frac{|\{T \in \mathcal{T} \mid C \in \mathcal{S}(T)\}|}{|\mathcal{T}|}$$

a freqüência relativa com que o corte C é encontrado na coleção de cortes $\mathcal{S}(\mathcal{T})$. Em outras palavras, $p(C, \mathcal{T})$ corresponde ao número de árvores em que C é encontrado dividido pelo número total de árvores presentes na coleção.

Para qualquer árvore filogenética T podemos definir a função:

$$p(T, \mathcal{T}) = \prod_{C \in \mathcal{S}(T)} p(C, \mathcal{T})$$

como sendo o *peso* da árvore T em relação à coleção $\mathcal{T}$. Como conseqüência da definição de peso de uma árvore, temos $p(T, \mathcal{T}) = 0$ quando $\mathcal{S}(T) \nsubseteq \mathcal{S}(\mathcal{T})$. Note que a função p equivaleria à probabilidade de se obter o sistema de cortes $\mathcal{S}$ tomando elementos de $\mathcal{S}(\mathcal{T})$ ao acaso, caso os eventos de escolha de um corte ocorressem de forma independente.

Sabemos que as escolhas de elementos de $\mathcal{S}(\mathcal{T})$ não podem ser feitas de maneira independente porque um sistema de cortes só equivale a uma árvore filogenética se seus cortes forem compatíveis dois a dois, o que não se aplica aos cortes presentes em $\mathcal{S}(\mathcal{T})$. Mesmo assim, abusaremos um pouco da linguagem chamando de *árvore mais provável* de um conjunto $\mathcal{T}$ uma árvore filogenética T^+ com as seguintes características:

- T^+ é completamente resolvida e sem raiz.
- $S(T^+)$ contém apenas cortes de $\mathcal{S}(\mathcal{T})$.
- Não existe árvore filogenética completamente resolvida e sem raiz T tal que $p(T^+, \mathcal{T}) < p(T, \mathcal{T})$.

É claro que para qualquer coleção $\mathcal{T}$ existe ao menos uma árvore mais provável, pois, uma vez que todas as árvores em $\mathcal{T}$ são completamente resolvidas, qualquer uma delas é formada apenas por elementos $\mathcal{S}(\mathcal{T})$ e para todas elas é possível calcular o valor de p. Se o conjunto de árvores formadas apenas por cortes de $\mathcal{S}(\mathcal{T})$ nunca é vazio, então sempre é possível escolher ao menos uma árvore cujo valor de p é máximo.

Assim como na Seção 2.3, aqui também é possível tirar proveito das relações entre sistemas de cortes e n-árvores para definir conceitos paralelos aos de freqüência relativa de um corte e peso de um sistema de cortes. Assim, seja

$$\mathcal{F}(\mathcal{T}) = \bigcup_{T \in \mathcal{T}} \mathcal{F}(T)$$

o conjunto de todos os subgrupos pequenos encontrados na coleção de árvores $\mathcal{T}$. Denotamos por $p(S, \mathcal{T})$ a freqüência relativa com que o subgrupo S é encontrado em $\mathcal{F}(\mathcal{T})$, definida formalmente como

$$p(S, \mathcal{T}) = \frac{|\{T \in \mathcal{T} \mid S \in \mathcal{F}(T)\}|}{|\mathcal{T}|},$$

e, para cada n-árvore Ψ podemos definir a função

$$p(\Psi, \mathcal{T}) = \prod_{S \in \Psi} p(S, \mathcal{T})$$

com sendo o *peso* da n-árvore Ψ em relação à coleção $\mathcal{T}$. Conforme veremos, há uma correspondência direta entre o peso de um sistema de cortes e o produto dos pesos das n-árvores maximais do conjunto dos subgrupos pequenos deste sistema. Antes, porém, apresentamos o lema a seguir, necessário para provar esta relação:

Lema 6.0.1 *Seja $\mathcal{T}$ uma coleção não vazia de árvores filogenéticas pertencentes a $\mathcal{T}^*_U(L)$ e sejam C um corte pertencente a $\mathcal{S}(\mathcal{T})$ e C_p um subgrupo pertencente a $\mathcal{F}(\mathcal{T})$ tais que C_p é o subgrupo pequeno de C. Então,*

$$p(C, \mathcal{T}) = p(C_p, \mathcal{T}).$$

Prova Pela definição do conjunto $\mathcal{F}(T)$ de uma árvore, sabemos que

$$C \in \mathcal{S}(T) \Leftrightarrow C_p \in \mathcal{F}(T),$$

o que significa que

$$\{T \in \mathcal{T} \mid C \in \mathcal{S}(T)\} = \{T \in \mathcal{T} \mid C_p \in \mathcal{F}(T)\}$$

e

$$\frac{|\{T \in \mathcal{T} \mid C \in \mathcal{S}(T)\}|}{|\mathcal{T}|} = \frac{|\{T \in \mathcal{T} \mid C_p \in \mathcal{F}(T)\}|}{|\mathcal{T}|},$$

o que é exatamente a mesma coisa que dizer que

$$p(C, \mathcal{T}) = p(C_p, \mathcal{T}).$$

□

Teorema 6.0.2 *Se $\mathcal{T}$ é uma coleção não vazia de árvores filogenéticas pertencentes a $\mathcal{T}_U^*(L)$ e T é uma árvore filogenética cujos cortes pertencem a $\mathcal{S}(\mathcal{T})$, então*

$$p(T, \mathcal{T}) = \prod_{\substack{\Psi \subset \mathcal{F}(T) \\ \Psi \text{ é } n\text{-árvore} \\ \Psi \text{ maximal}}} p(\Psi, \mathcal{T}).$$

Prova Este teorema é provado pela combinação dos Lemas 6.0.1 e 2.3.4. Lembrando que, por definição,

$$p(T, \mathcal{T}) = \prod_{C \in \mathcal{S}(T)} p(C, \mathcal{T}).$$

Pelo Lema 6.0.1, chegamos à conclusão que

$$p(T, \mathcal{T}) = \prod_{C_p \in \mathcal{F}(T)} p(C_p, \mathcal{T}).$$

Pelo Lema 2.3.4 podemos reescrever esta igualdade da seguinte maneira:

$$\begin{aligned} p(T, \mathcal{T}) &= \prod_{\substack{\Psi \subset \mathcal{F}(T) \\ \Psi \text{ é } n\text{-árvore} \\ \Psi \text{ maximal}}} \prod_{C_p \in \Psi} p(C_p, \mathcal{T}) \\ &= \prod_{\substack{\Psi \subset \mathcal{F}(T) \\ \Psi \text{ é } n\text{-árvore} \\ \Psi \text{ maximal}}} p(\Psi, \mathcal{T}). \end{aligned}$$

□

6.1 Um Algoritmo para Construir Árvores Mais Prováveis

Esta e as demais seções deste capítulo são destinadas à apresentação, prova de corretude e análise de complexidade de um algoritmo que constrói o conjunto das árvores mais prováveis para um conjunto de árvores filogenéticas completamente resolvidas.

Podemos usar o Teorema 2.3.5 para desenvolver uma estratégia de atacar este problema, pois não é difícil obter todos os subgrupos pequenos de todas as árvores completamente resolvidas. Com o conjunto de subgrupos pequenos em mãos, sabemos que, se encontrarmos um subconjunto deste conjunto que determine três n-árvores completamente resolvidas, teremos determinado o sistema de cortes de uma árvore filogenética completamente resolvida. Para determinar este conjunto de três n-árvores, o algoritmo descrito a seguir associa a cada subgrupo uma n-árvore completamente resolvida definida sobre o subgrupo em questão. O ato de associar ao subgrupo uma n-árvore completamente resolvida é chamado de *resolver* um subgrupo, a n-árvore associada ao subgrupo é conhecida como a *solução* do subgrupo e todo subgrupo que tem uma solução associada é chamado de subgrupo *resolvido*.

O conjunto de árvores mais prováveis é retornado pelo algoritmo na forma de uma estrutura de dados que chamaremos de *floresta de árvores de pesos iguais* ou simplesmente *floresta*, e que será representada pela letra F. Uma floresta possui um valor associado $F.ms$ igual ao peso das árvores armazenadas nela. As árvores são armazenadas na forma de triplas de n-árvores maximais. Sabemos pelo Teorema 2.3.5 que uma árvore qualquer T é completamente resolvida se e somente se $\mathcal{F}(T)$ contiver exatamente três n-árvores maximais completamente resolvidas, assim, qualquer árvore mais provável pode ser representada na forma de uma tripla de n-árvores completamente resolvidas. As árvores são armazenadas de forma que cada subgrupo presente na floresta seja representado apenas uma vez. Assim, há basicamente três tipos de sub-estruturas nesta estrutura de dados:

Subgrupo: A representação de um subgrupo pequeno da coleção de árvores filogenéticas completamente resolvidas $\mathcal{T}$. Um subgrupo consiste basicamente de quatro campos:

- **Elementos:** Os elementos do conjunto de espécies L que formam este subgrupo.
- **Freqüência Relativa:** Corresponde ao valor $p(S, \mathcal{T})$ definido na página 76. É representado pelo sufixo $.p$ adicionado ao nome do subgrupo.
- **Lista de Melhores Soluções:** Lista ligada de pares de subgrupos que correspondem às melhores soluções para este subgrupo. É indicada pelo sufixo $.lms$ adicionado ao nome do subgrupo.

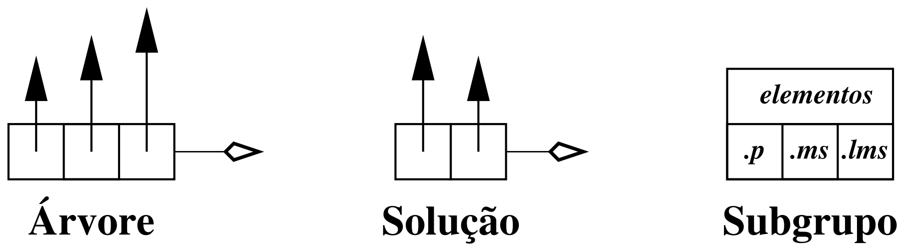

Figura 6.1: Representação gráfica usada na Figura 6.2 para os três tipos de sub-estruturas da estrutura de dados usada pelo algoritmo. As setas pretas são apontadores para subgrupos. Árvores e Soluções sempre são encontradas em listas ligadas e o apontador para o próximo elemento da lista é indicado pela seta de cor branca (a ausência de seta indica o fim da lista).

Peso da Melhor Solução: Corresponde ao peso das soluções armazenadas na lista de melhores soluções. Este valor é indicado pelo sufixo *.ms* adicionado ao nome do subgrupo.

Árvore: Consiste de uma tripla apontadores que apontam para os subgrupos nos quais as n-árvores maximais de uma árvore da floresta estão definidos.

Solução: Consiste num par de apontadores que apontam para dois subgrupos disjuntos A e B que, juntos, formam uma solução para o subgrupo $A \cup B$.

A Figura 6.2 exemplifica a estrutura de dados descrita anteriormente usando uma floresta composta por duas árvores filogenéticas completamente resolvidas. Quando conhecemos as árvores que compõem a floresta e temos uma representação gráfica como a Figura 6.2, para florestas com poucas árvores é relativamente fácil verificar que a estrutura realmente representa as árvores da floresta. Isso já não é tão fácil quando tudo o que temos é a estrutura e precisamos ter certeza de que as árvores mais prováveis estão representadas nela. Para isso precisaremos de dois conceitos, apresentados logo a seguir.

Dados dois subgrupos pequenos quaisquer S e U, ambos pertencentes a $\mathcal{F}(\mathcal{T})$, dizemos que U é *alcançável* a partir de S numa floresta F se um dos seguites casos ocorrer:

- $S = U$
- Existe um subgrupo V em ao menos um dos pares da lista $S.lms$ tal que U é alcançável a partir de V.

Numa definição mais informal, todo subgrupo é alcançável a partir de si mesmo e, se S e U são subgrupos diferentes, então U é alcançável a partir de S somente se existir um

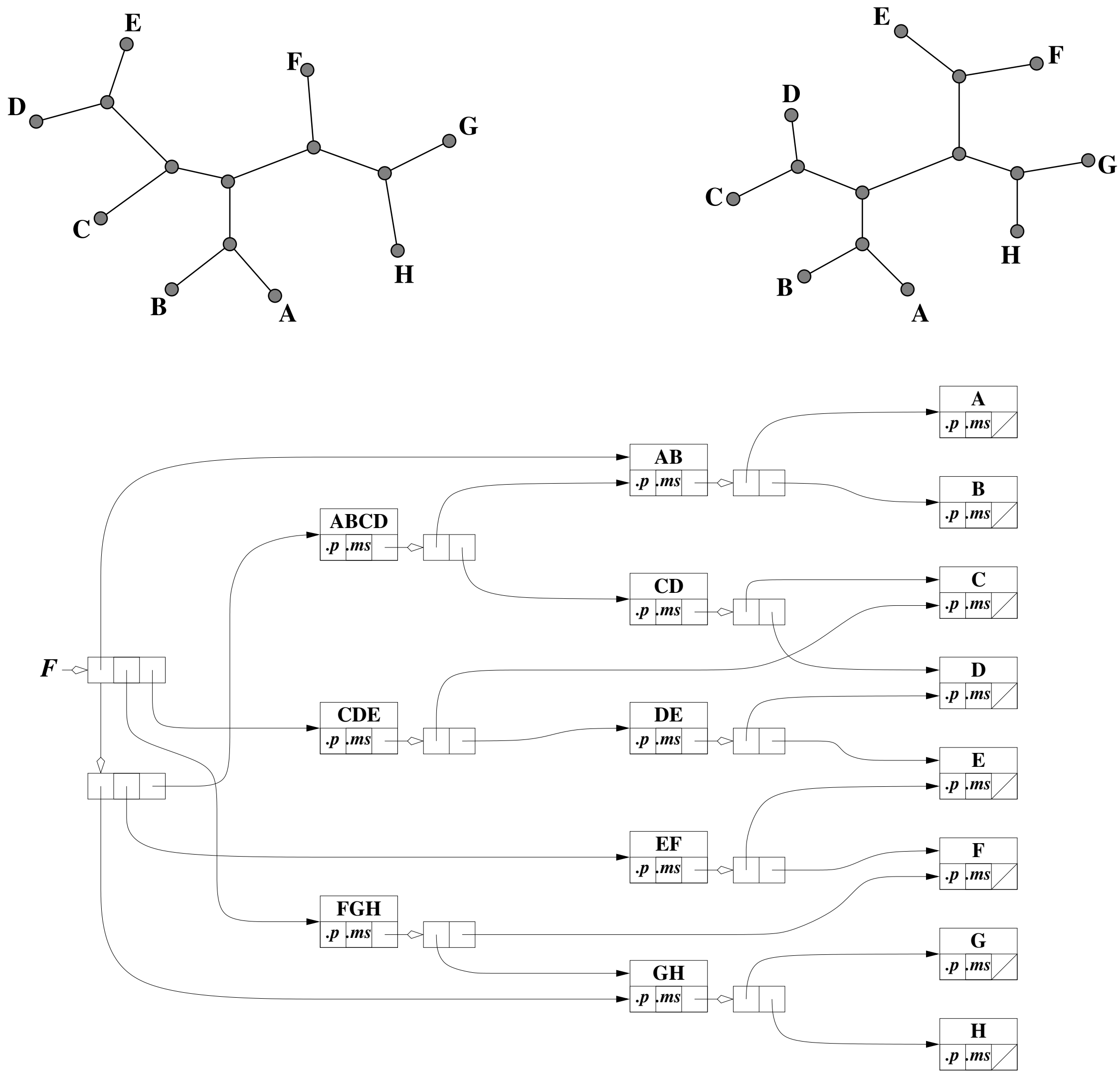

Figura 6.2: Uma floresta composta por duas árvores filogenéticas completamente resolvidas com conjunto de folhas $L = \{A, B, C, D, E, F, G, H\}$ e a representação da estrutura de dados usada para armazená-la. As sub-estruturas utilizadas são apresentadas em maiores detalhes na Figura 6.1. A enumeração $\varphi : L \mapsto \mathbb{N}_L$ é definida como a ordem ocupada pelo elemento de L quando L está ordenado lexicograficamente em ordem crescente, por isso o subgrupo $\{A, B, C, D\}$ é menor que o subgrupo $\{E, F, G, H\}$. A figura se destina meramente a exemplificar a utilização da estrutura de dados usada no algoritmo para representar uma floresta de árvores filogenéticas. Por este motivo, valores dos campos $.p$ e $.ms$ são omitidos, pois não há neste exemplo nenhuma preocupação com a garantia de que alguma coleção $\mathcal{T}$ fornecida para o algoritmo tenha como coleção de árvores mais prováveis exatamente esta floresta. Note que, apesar de duas árvores estarem representadas na estrutura, cada um dos subgrupos pequenos é representado uma única vez.

caminho de S a U num diagrama como o da Figura 6.2. Note que para que um subgrupo seja alcançável a partir de outro não basta apenas que ele esteja contido no outro, como mostra o caso dos subgrupos $\{C, D, E\}$ e $\{C, D\}$ na Figura 6.2, mas é preciso que, além de um deles estar contido no outro, ambos pertençam ao mesmo conjunto $\mathcal{F}(T)$ para ao menos um $T \in \mathcal{T}$.

Dizemos que uma árvore filogenética completamente resolvida T *está representada* numa floresta F se as duas condições abaixo forem verificadas:

- Há uma árvore em F cujos subgrupos são os subconjuntos de L sobre os quais as n-árvores maximais de $\mathcal{F}(T)$ estão definidas.
- Para qualquer subgrupo não trivial S em $\mathcal{F}(T)$, o par de grupos (A, B) tais que A e B pertencem a $\mathcal{F}(T)$, $A \cup B = S$ e $A \cap B = \emptyset$ está em $S.lms$

Intuitivamente, dizer que uma árvore filogenética completamente resolvida T está representada em F significa dizer que há uma maneira de escolher um elemento em cada lista ligada de forma que, eliminando todos os elementos não escolhidos, fiquemos com uma árvore binária equivalente a T.

Subgrupos podem ter diversas soluções diferentes e nem todas necessariamente levam a uma árvore filogenética completamente resolvida que maximiza $p(T, \mathcal{T})$. Desta forma, não basta apenas encontrar uma solução para cada subgrupo. É preciso também que essa solução seja ótima, no sentido de que a solução Ψ associada a este subgrupo maximiza $p(\Psi, \mathcal{T})$, caso contrário esta solução jamais faria parte de uma árvore mais provável. Chamaremos $p(\Psi, \mathcal{T})$ de *participação* de uma n-árvore no peso de uma árvore, uma vez que, dada uma n-árvore Ψ formada pelos subgrupos pequenos de cortes definidos por arestas de uma árvore filogenética T, podemos reescrever a função peso de T como sendo:

$$p(T, \mathcal{T}) = \prod_{S \in \mathcal{S}(T) \text{ e } S_p \in \Psi} p(S, \mathcal{T}) \times \prod_{S \in \mathcal{S}(T) \text{ e } S_p \notin \Psi} p(S, \mathcal{T}).$$

Em todos os procedimentos, A, B e C correspondem a subgrupos de L. Além disso, se A é um subgrupo de L, então $A.p$ denota a freqüência relativa $p(A, \mathcal{T})$; o atributo $A.ms$ denota a participação da n-subárvore definida sobre A no valor de $p(T, \mathcal{T})$, caso a melhor solução encontrada até o momento para A seja utilizada na construção da árvore mais provável; e $A.lms$ denota um apontador para o primeiro elemento de uma lista ligada que guarda todas as soluções encontradas até o momento para A cuja participação é máxima. O algoritmo é apresentado abaixo em pseudocódigo e discutido logo a seguir.

```
ARVORE-MAIS-PROVAVEL(𝒯)
Pequenos ← PEQUENOS(𝒯)
F.lms ← ∅
```

```
F.ms ← −∞
for each A in Pequenos, tomados em ordem crescente
    do for each B in {B ∈ Pequenos | B < A}
           do if (A ∩ B = ∅)
                 then C ← A ∪ B
                      if (|C| ≤ |L| /2)
                        then if (C ∈ Pequenos)
                                then Atualiza-Subgrupo(C, A, B)
                                else Descarte o par (A, B)
                        else C′ ← L \ C
                             if (C′ < B e C′ ∈ Pequenos)
                                then Atualiza-Floresta(A, B, C′, F)
                                else Descarte a tripla (A, B, C′)
                 else Descarte o par (A, B)
return F
```

O algoritmo começa preparando o conjunto de árvores fornecido de modo a obter a união dos conjuntos de subgrupos pequenos destas árvores. O vetor de subgrupos retornado pela função Pequenos() contém uma única cópia de cada subgrupo pequeno presente em $\mathcal{T}$ e é ordenado de acordo com a relação definida na Seção 2.3, já com o valor da freqüência relativa associado e com a lista e o valor de melhores soluções inicializados. A princípio, apenas subgrupos triviais são resolvidos, pois suas soluções são n-árvores unitárias contendo o único elemento do subgrupo, enquanto o valor desta solução é a freqüência relativa do corte que tem neste subgrupo o seu subgrupo pequeno. Como cortes triviais estão presentes em todas as árvores, a freqüência relativa de subgrupos triviais é 1, assim como o peso de sua única e, portanto, melhor solução, que consiste na n-árvore trivial composta unicamente pelo próprio subgrupo. Os demais subgrupos têm a lista de melhores soluções inicializadas com $\emptyset$ e o peso de sua melhor solução com o valor $-\infty$, o que garante que nenhuma solução inicial será descartada. A inicialização de subgrupos é tratada com maiores detalhes na Seção 6.1.1.

Com o vetor de subgrupos preparado, o algoritmo começa a verificar os pares de subgrupos de maneira ordenada, verificando se o par de subgrupos analisado a cada iteração é candidato a solução de um dos subgrupos pequenos ou se ele pode, juntamente com um terceiro subgrupo pertencente a *Pequenos*, formar um conjunto de três n-árvores completamente resolvidas, ou seja, um novo candidato a árvore mais provável.

Nem todo par de subgrupos pequenos é interessante na construção de candidatos a árvore mais provável. Pares que não são compatíveis, por exemplo, devem ser descartados, pois jamais poderiam ser encontrados juntos em uma n-árvore, enquanto pares compatíveis em que um dos subgrupos está contido no outro também não são interessantes, pois não são capazes de, juntos, criar uma solução diferente das já encontradas pelo algoritmo. O único tipo de par que realmente interessa é o par em que os subgrupos A e B, além de compatíveis, são também disjuntos, pois quando isto acontece, o par A, B pode vir a ser usado em um dos casos abaixo:

Solução Para Um Terceiro Subgrupo: Se $C = A \cup B$ tiver cardinalidade menor ou igual a $|L|/2$, então C pode ser o subgrupo pequeno do corte $\{C, L \setminus C\}$ e, no caso de ser realmente o subgrupo pequeno deste corte, pode ainda estar no vetor de subgrupos. Note que o conjunto formado pela união das soluções de A e B com o subgrupo C é claramente uma solução para C.

Descoberta de Uma Nova Candidata a Árvore Mais Provável: Se $C = A \cup B$ tiver cardinalidade maior que a $|L|/2$, então $C' = L \setminus C$ é o subgrupo pequeno do corte $\{C, L \setminus C\}$ e também pode estar no vetor de subgrupos. Além disso, qualquer conjunto da forma $A.lms[i] \cup B.lms[j] \cup C'.lms[k]$, onde i, j e k são índices válidos, é uma união de n-árvores disjuntas e, se as três n-árvores forem completamente resolvidas, determina o conjunto de cortes de uma árvore completamente resolvida.

O desafio do algoritmo é garantir que todos os subgrupos estejam resolvidos e que somente as triplas que representam as árvores mais prováveis estejam no conjunto F ao final dos loops aninhados nas linhas 4 a 16, o que faz com que o algoritmo retorne uma estrutura com todas as árvores mais prováveis do conjunto. Os Teoremas 6.1.1 e 6.1.5, apresentados a seguir, provam que ambas as condições são verificadas.

A prova dos teoremas a seguir referenciam os subgrupos A, B e C em iterações diferentes, representando, portanto, trios distintos de subgrupos do vetor *Pequenos*. Para facilitar o entendimento, o nome do subgrupo acompanhado por um índice i denota o subgrupo correpondente na i-ésima iteração do loop da linha 4.

Teorema 6.1.1 *A cada execução da linha 4 do algoritmo, o subgrupo A está resolvido e $A.ms$ é máximo.*

Prova Provaremos este teorema usando indução na cardinalidade de A.

Base: Quando $|A| = 1$ o teorema se verifica, pois como dito anteriormente, subgrupos de cardinalidade um têm solução trivial, que é o próprio subgrupo, e o valor desta solução, igual a um, é único, logo, é máximo.

Hipótese de Indução: A hipótese é verdadeira para qualquer subgrupo com cardinalidade menor que um número inteiro a.

Passo: Seja X um subgrupo com cardinalidade igual a a e sejam Y e Z, com $Y > Z$, dois subgrupos tais que $Y \cup Z = X$, $Y \cap Z = \emptyset$ e $Y.ms \times Z.ms$ seja máximo. Note que, como X é o subgrupo pequeno de algum corte de uma árvore filogenética completamente resolvida T usada para a criação do conjunto de subgrupos, existe ao menos um par de subgrupos Y, Z no conjunto de subgrupos tal que $Y \cup Z = X$ e $Y \cap Z = \emptyset$, uma vez que o conjunto dos subgrupos pequenos de T determina um conjunto de três n-árvores independentes e completamente resolvidas, de acordo com o Teorema 2.3.5.

Denotaremos por A_i o valor da variável A na i-ésima iteração. Como os subgrupos são tomados em ordem crescente na linha 4 do algoritmo, temos três iterações distintas i, j e k onde $A_i = Z$, $A_j = Y$ e $A_k = X$ onde $i < j < k$. Assim, na iteração j, tanto o subgrupo $Y = A_j$ quanto o subgrupo Z estão resolvidos e o peso de suas soluções é máximo, por hipótese de indução, já que eles têm cardinalidade menor que a. Também na iteração j os subgrupos menores que Y são tomados um a um no loop da linha 5, de modo que há uma iteração do loop da linha 5 na qual o par formado pelos subgrupos Y e Z é analisado e, da forma como Y e Z foram escolhidos, nesta iteração as condições para a execução da linha 10 do algoritmo são satisfeitas e o procedimento Atualiza() é executado com parâmetros X, Y e Z, lembrando que $X = Y \cup Z$ e $|X| \leq |L|/2$, já que $X \in Pequenos$.

Como o valor de $Y.ms \times Z.ms$ é máximo, o peso desta solução, claramente igual a $X.p \times Y.ms \times Z.ms$, é máximo e será armazenado pelo procedimento na lista de melhores soluções de X, $X.lms$, causando também, caso necessário, a atualização do valor de $X.ms$. Neste momento, claramente anterior à iteração k, $X.ms$ é máximo e $X.lms$ contém pelo menos uma solução para X.

□

Corolário 6.1.2 *A cada execução da linha 4 do algoritmo, o subgrupo A está resolvido e $A.lms$ contém todos os pares (B, C) de subgrupos pertencentes a $\mathcal{F}(\mathcal{T})$ tais que $A.p \times B.ms \times C.ms$ é máximo.*

Prova Mostramos na prova do Teorema 6.1.1 que ao menos um par como o citado no enunciado do corolário é inserido na lista de melhores soluções de A. Sabemos, no entanto, que a solução de um subgrupo só pode ser feita com a combinação de subgrupos menores que o próprio subgrupo e todos os pares formados exclusivamente por subgrupos menores que A são analisados antes que o próprio subgrupo A. Assim, todo par (B, C) de subgrupos pertencentes a $\mathcal{F}(\mathcal{T})$ tal que $A.p \times B.ms \times C.ms$ é máximo é analisado antes de A e inserido em sua lista de melhores soluções.

□

Devido ao fato de as árvores mais prováveis serem formadas única e exclusivamente por cortes de $\mathcal{S}(\mathcal{T})$, sabemos que todos os subgrupos pequenos de árvores filogenéticas mais prováveis estão resolvidos ao final do loop da linha 4, tal como sugere o corolário acima. Não sabemos, no entanto, se é possível recuperar todas as árvores mais prováveis da estrutura de dados retornada pelo algoritmo.

Lema 6.1.3 *Se T é uma árvore mais provável de $\mathcal{T}$ então, logo antes da execução da linha 17 do algoritmo, há uma árvore em F cujos subgrupos são os subconjuntos de L sobre os quais as n-árvores maximais de $\mathcal{F}(T)$ estão definidas.*

Prova O fato de T ser uma árvore mais provável de $\mathcal{T}$ nos indica que $\mathcal{F}(T)$ possui 3 n-árvores maximais (Teorema 2.3.5) e $\mathcal{F}(T) \subset \mathcal{F}(\mathcal{T})$. Sejam X, Y e Z os três subgrupos nos quais as n-árvores maximais de $\mathcal{F}(\mathcal{T})$ estão definidas. Podemos supor ainda, sem perda de generalidade que $X > Y > Z$. Como $\{X, Y, Z\} \subset \mathcal{F}(T)$, há uma iteração do loop da linha 5 em que o subgrupo X corresponde ao subgrupo A, o subgrupo Y corresponde ao B e o subgrupo Z ao C'. Neste momento, como $|X \cup Y| > |L|/2$, o teste da linha 13 é executado e, como $Y > Z$ e $Z \in \mathcal{F}(\mathcal{T})$, o procedimento Atualiza-Floresta() é executado.

Como T é uma das árvores mais prováveis de $\mathcal{T}$, a expressão

$$X.ms * Y.ms * Z.ms \geq F.ms$$

é verdadeira, assim, a linha 2 ou a linha 5 do procedimento é executada (ver página 88), dependendo das árvores já armazenadas em F, o que de qualquer forma leva à inclusão da árvore (X, Y, Z) em F. Lembrando que não há árvore T' tal que $p(T', \mathcal{T}) > p(T, \mathcal{T})$ por definição, a tripla (X, Y, Z) não é excluída de F em nenhuma iteração posterior.

□

Lema 6.1.4 *Se T é uma árvore mais provável de $\mathcal{T}$, então, logo antes da execução da linha 17 do algoritmo, para cada subgrupo não trivial A em $\mathcal{F}(T)$ há um par de subgrupos (B, C) em $A.lms$ tal que tanto B quanto C pertencem a $\mathcal{F}(T)$.*

Prova Seja $\Psi_A = \mathcal{F}(T)[A]$ a n-subárvore induzida por A em $\mathcal{F}(T)$ e sejam Ψ_B e Ψ_C as n-subárvores definidas da mesma forma para B e C, respectivamente. Note que, como Ψ_A é completamente resolvida, pelo Teorema 2.2.3, B e C certamente existem e são tais que $B \cup C = A$ e $B \cap C = \emptyset$. Além disso, note que o fato de T ser uma árvore mais provável de $\mathcal{T}$ garante que dentre todas as n-árvores completamente resolvidas definidas sobre A, Ψ_A é uma das que tem o maior peso, ou seja, $p(\Psi_A, \mathcal{T})$ é máximo entre as n-árvores completamente resolvidas definidas sobre A. Analogamente, chegamos à conclusão de que $p(\Psi_B, \mathcal{T})$ e $p(\Psi_C, \mathcal{T})$ também são maximais. Assim, quando o par (B, C) é analisado pelo algoritmo, a linha 10 é executada, porque $A = B \cup C$ pertence a *Pequenos* e, como $A.p * B.ms * C.ms$ é maximo o procedimento Atualiza-Subgrupo() certamente inclui (B, C) em $A.lms$.

□

Teorema 6.1.5 *Imediatamente antes da execução da linha 17, qualquer que seja a árvore mais provável T da coleção $\mathcal{T}$, T está representada na lista F.*

Prova Pelo Lema 6.1.3, há uma árvore em *F.lms* cujos subgrupos são os subconjuntos de L sobre os quais as n-árvores maximais de $\mathcal{F}(T)$ estão definidas. Assim, resta saber se a segunda característica necessária para que T esteja representada em F é verificada, o que é comprovado pelo Lema 6.1.4.

□

6.1.1 Pré-processamento da Coleção de Árvores Filogenéticas

A função Pequenos() recebe um conjunto de árvores filogenéticas completamente resolvidas sem raiz $\mathcal{T}$ e retorna um vetor contendo todos os subgrupos pequenos dos cortes encontrados nesta coleção. O vetor retornado se encontra ordenado em ordem crescente e os subgrupos já estão inicializados conforme descrito na página 82.

```
Pequenos(𝒯)
 1  Pequenos ← ∅
 2  for each T in 𝒯
 3      do for each S in ℱ(T)
 4              do if (S ∈ Pequenos)
 5                      then i ← Índice de S em Pequenos
 6                           Pequenos[i].c ← Pequenos[i].c + 1
 7                      else S.c ← 1
 8                           Inclua S em Pequenos mantendo-o ordenado.
 9  for each subgrupo A ∈ Pequenos
10      do if (|A| = 1)
11            then A.p ← 1
12                 A.ms ← 1
13                 A.lms ← {A}
14            else A.p ← A.c/ |𝒯|
15                 A.ms ← −∞
16                 A.lms ← ∅
17  return Pequenos
```

A coleção $\mathcal{T}$ é representada através de uma lista de árvores filogenéticas, onde cada árvore é uma lista de cortes ou, equivalentemente, de seus subgrupos pequenos. Cada subgrupo S é representado através de um par composto por um inteiro, que determina o tamanho do subgrupo, e uma string $s_{|L|-1}s_{|L|-2}\dots s_2 s_1 s_0$ de $|L|$ bits, onde $s_i = 1$ se e somente se $\varphi^{-1}(i) \in S$, para $0 \leq i \leq |L|-1$, onde φ^{-1} é a função inversa da enumeração dos elementos de L definida na página 10. Desta forma, a cardinalidade do subgrupo pode ser determinada em tempo constante e testes de compatibilidade de subgrupos também tornam-se bem mais simples, supondo que $|L|$ cabe na palavra do computador, pois dados subgrupos S e R e suas respectivas strings s e r, $S \cap R = \emptyset$ se e somente se $s \wedge r = 0$, onde $\wedge$ indica "e" lógico bit a bit; $S \subset R$ se e somente se $s \vee r = r$ e $R \subset S$ se e somente se $r \vee s = s$, onde $\vee$ indica "ou" lógico bit a bit. Mesmo quando a cardinalidade do conjunto L é maior que a palavra do computador, estas operações ainda podem ser feitas em $O(|L|)$.

Nas linhas 2 a 8, os subgrupos pequenos das árvores da coleção $\mathcal{T}$ são inseridos um a um no conjunto de subgrupos *Pequenos*. Nesta inserção, subgrupos repetidos são descartados e cada um dos subgrupos mantém um contador, por exemplo $S.c$, responsável por contabilizar o número de árvores em que ele foi encontrado, facilitando atribuições futuras, como a que ocorre na linha 14. Durante todo o tempo em que os subgrupos são inseridos, o vetor se mantém ordenado de acordo com a relação estabelecida na Seção 2.3 e os subgrupos são inicializados da maneira descrita na página 82, de modo que o vetor retornado na linha 17 contém uma única cópia de cada subgrupo, devidamente inicializada, e está ordenado, tal como o algoritmo propriamente dito requer. Considerações sobre o tempo de execução deste trecho do algoritmo podem ser encontradas na Seção 6.2.1.

6.1.2 Resolução de um subgrupo

Uma n-subárvore pertence ao conjunto de subgrupos pequenos de uma árvore completamente resolvida somente se for uma n-subárvore completamente resolvida, pois o Teorema 2.3.5 determina que um conjunto de n-árvores disjuntas só pode representar o conjunto de subgrupos pequenos do sistema de cortes de uma árvore filogenética completamente resolvida se for composto por três n-árvores completamente resolvidas.

Além disso, pelo Teorema 2.2.3, sabemos que uma n-subárvore só é completamente resolvida se todo subgrupo não unitário nesta n-subárvore tiver um corte em que ambas as partições pertençam à n-subárvore. Desta forma, nos referiremos a um subgrupo qualquer C de L como *resolvido* quando pudermos associar a este subgrupo um par A, B de subgrupos resolvidos e chamaremos o par A, B de *solução* de C. Consideramos os subgrupos triviais unitários resolvidos por definição e a sua solução é o próprio subgrupo.

O procedimento Atualiza-Subgrupo() tem a função de garantir que somente soluções melhores ou iguais às soluções previamente encontradas serão armazenadas para o subgrupo em questão. Os parâmetros A, B e C correspondem aos subgrupos relacionados de tal forma que o par A, B forma uma nova solução para C. O primeiro teste feito neste procedimento verifica se a resolução de C com os subgrupos A e B é uma solução para C melhor do que as encontradas até então e, em caso afirmativo, descarta a lista guardada em $C.lms$, criando uma nova lista contendo apenas o par A, B, além de atualizar o valor de $C.ms$. Caso contrário, ainda há a possibilidade de a nova solução ser tão boa quanto as melhores soluções encontradas até o momento e, neste caso, o par A, B passa a encabeçar a lista de soluções de C.

Não é difícil perceber que a execução de Atualiza-Subgrupo() é feita em tempo constante, pois nos piores casos, ou seja, nos casos em que a lista de soluções deve ser alterada, o procedimento executa uma comparação e duas atribuições ou duas comparações e a inserção de um elemento no início de uma lista ligada.

```
Atualiza-Subgrupo(C, A, B)
1  if (C.p * A.ms * B.ms > C.ms)
2     then C.lms ← {(A, B)}
3          C.ms ← C.p * A.ms * B.ms
4     else if (C.p * A.ms * B.ms = C.ms)
5             then C.lms ← C.lms ∪ {(A, B)}
```

6.1.3 Seleção das Árvores Mais Prováveis

À medida em que o algoritmo encontra trios de subgrupos cujas soluções possam compor o conjunto $\mathcal{F}(T)$ de uma é uma árvore mais provável T, o procedimento Atualiza-Floresta(), apresentado a seguir, se encarrega de atualizar a floresta de árvores mais prováveis.

```
Atualiza-Floresta(A, B, C', F)
1  if (A.ms * B.ms * C'.ms > F.ms)
2     then F.lms ← {(A, B, C')}
3          F.ms ← A.ms * B.ms * C'.ms
4     else if (A.ms * B.ms * C'.ms = F.ms)
5             then F.lms ← F.lms ∪ {(A, B, C')}
```

Os procedimentos Atualiza-Subgrupo() e Atualiza-Floresta() são essencialmente iguais, pois assim como para cada subgrupo só interessam as soluções de maior peso, para a floresta de árvores mais prováveis só interessam as árvores de maior peso, já que por definição nenhuma das outras poderia ser uma árvore mais provável.

Assim, o teste feito na linha 1 do procedimento tenta identificar o caso em que uma árvore com peso maior que as árvores na floresta é encontrada. Neste caso, todas as demais árvores são descartadas, permanecendo apenas a nova árvore na floresta. Além disso, o valor do peso das árvores, $F.ms$, deve ser atualizado, o que acontece na linha 3. No caso em que a nova árvore não é melhor (ou mais pesada) que as árvores presentes na floresta, ainda há a possibilidade de seu peso ser igual ao das outras árvores. Neste caso, a nova árvore deve ser incluída na floresta, o que é feito na linha 5.

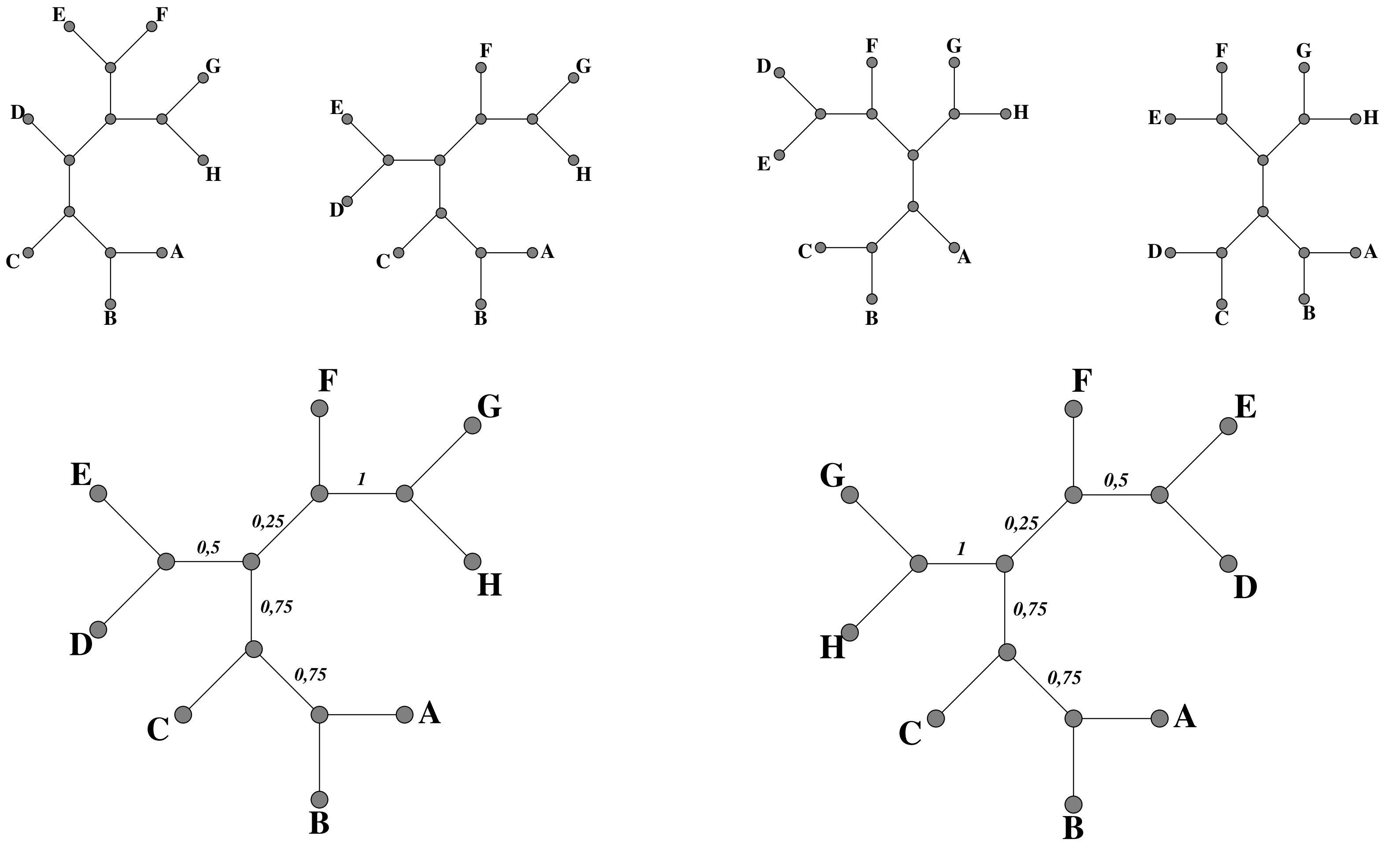

Figura 6.3: A coleção de árvores filogenéticas mostrada na parte superior da figura possui duas Árvores Mais Prováveis, mostradas em destaque na parte inferior. Note que a Árvore Mais Provável da esquerda também é encontrada da coleção, enquanto a da direita não. O peso de ambas as árvores é $0,0703125$, enquanto as freqüências relativas dos cortes não triviais são indicadas nas Árvores Mais Prováveis.

É importante notar que qualquer árvore com peso inferior ao das árvores presentes na floresta é simplesmente ignorada, de modo que, a partir do momento em que a primeira árvore mais provável é inserida na floresta, a mesma passa a conter apenas árvores mais prováveis. A Figura 6.3 mostra uma coleção de árvores filogenéticas completamente resolvidas e as duas árvores mais prováveis desta coleção.

6.2 Considerações Sobre a Complexidade do Algoritmo

É de se esperar que a complexidade do algoritmo gire em torno de duas variáveis: o número t de árvores presentes na coleção de árvores fornecidas como entrada e o número l de folhas destas árvores. De fato, o teorema a seguir mostra que o tamanho do vetor de subgrupos pequenos do algoritmo depende diretamente destas duas variáveis.

Teorema 6.2.1 *O número de subgrupos pequenos distintos em uma coleção $\mathcal{T}$ composta por t árvores filogenéticas completamente resolvidas com l folhas cada uma é $\Omega(l)$ e $O(tl)$.*

Prova Pelo fato de todas as árvores filogenéticas determinarem todos os cortes trivias possíveis de L, quaiquer duas árvores em $\mathcal{T}$ têm no mínimo l subgrupos pequenos em comum. Logo, o que determina o tamanho do conjunto de subgrupos distintos é o número de subgrupos distintos que podemos encontrar entre os $t \times (l - 3)$ subgrupos pequenos restantes. Neste caso, temos dois extremos:

Todas as Árvores são Exatamente Iguais: Neste caso temos somente $l - 3$ subgrupos distintos não triviais, num total de $2l - 3$ subgrupos pequenos para toda a coleção, não importando o tamanho dela. Não é difícil ver que este é o menor tamanho possível para o conjunto de subgrupos de uma coleção de árvores com l folhas. Assim, qualquer que seja a coleção de árvores $\mathcal{T}$, o número de subgrupos pequenos distintos em $\mathcal{T}$ é maior ou igual a $2l - 3$, o que significa que o número de subgrupos distintos numa coleção de árvores filogenéticas completamente resolvidas com l folhas é $\Omega(l)$.

Todas as Árvore são Completamente Diferentes: Neste caso, nenhum dos subgrupos pequenos não-triviais é igual a qualquer outro e o número de subgrupos distintos é igual à soma dos l subgrupos pequenos triviais com os $t \times (l - 3)$ subgrupos pequenos não triviais, num total de $tl - 3t + l$ subgrupos e sabemos que nenhuma coleção $\mathcal{T}$ de árvores filogenéticas completamente resolvidas pode ter um número de subgrupos maior que este. Assim, o número de subgrupos de uma coleção qualquer de t árvores filogenéticas todas com l folhas é $O(tl)$.

Linha	**Repetições**	**Custo**	**Total**
1	$\Theta(1)$	$\Theta(1)$	$\Theta(1)$
2	$\Theta(t)$	$\Theta(1)$	$\Theta(t)$
3	$\Theta(2lt - 3t)$	$\Theta(1)$	$\Theta(2lt - 3t)$
4	$\Theta(2lt - 3t)$	$O(l \lg p)$	$O(2l^2t \lg p - 3lt \lg p)$
$5 - 6$	$\Theta(2lt - 3t - p)$	$\Theta(1)$	$\Theta(2lt - 3t - p)$
7	$\Theta(p)$	$\Theta(1)$	$\Theta(p)$
8	$\Theta(p)$	$O(p)$	$O(p^2)$
9	$\Theta(p)$	$\Theta(1)$	$\Theta(p)$
10	$\Theta(p)$	$\Theta(1)$	$\Theta(p)$
$11 - 13$	$\Theta(l)$	$\Theta(1)$	$\Theta(l)$
$14 - 16$	$\Theta(p - l)$	$\Theta(1)$	$\Theta(p - l)$
17	$\Theta(1)$	$\Theta(1)$	$\Theta(1)$

Tabela 6.1: Estimativa do tempo de execução do procedimento PEQUENOS(), descrito na página 86. Para cada linha é apresentado um limitante superior para o número de vezes que esta linha é repetida (coluna *Repetições*) e o custo de cada repetição (coluna *Custo*). A estimativa para o tempo de execução é dada na coluna *Total*. Os limites são apresentados em termos do número de folhas de cada árvore, l, o total de árvores na coleção, t, e o número de subgrupos pequenos distintos encontrados na coleção, p.

□

Ao olhar para a descrição do algoritmo na página 81 fica claro que a sua complexidade depende da complexidade de dois trechos distintos: a preparação dos dados de entrada na linha 1 e a análise de todos os pares de subgrupos possíveis nas linhas 4 a 16. A complexidade de cada um destes trechos é estudada nas seções seguintes.

Em vários casos a complexidade de trechos do algoritmo depende do logaritmo de um dos parâmetros. Sempre que isto ocorre, o logaritmo aparece devido a uma busca binária, portanto está na base 2 e será denotado por lg.

6.2.1 Preparação da Coleção de Árvores

Na preparação da coleção de árvores, as árvores filogenéticas da coleção $\mathcal{T}$ são tomadas uma a uma e seus subgrupos pequenos são inseridos na lista *Pequenos* de maneira que esta lista, ao final do procedimento PEQUENOS(), contenha todos os subgrupos pequenos de todas as árvores da coleção ordenados em ordem crescente de acordo com a relação entre subgrupos apresentada na Seção 2.3. Sempre que um subgrupo é tomado, o algoritmo procura por um outro subgrupo igual a ele na lista de subgrupos pequenos. Se o subgrupo igual é encontrado, o contador de ocorrências do subgrupo na lista de pequenos é incrementado, caso contrário, o novo subgrupo é inserido na lista na posição correta e seu contador de ocorrências é inicializado.

A Tabela 6.1 apresenta limitantes superiores para o tempo de execução do procedimento de preparação do conjunto de árvores. Para cada linha do procedimento são dados limites superiores para o número de vezes que a linha é executada, para o custo de uma execução da linha e a para parcela do tempo total de execução do algoritmo relativa àquela linha.

Os parâmetros usados na análise do algoritmo são os mesmos apresentados no início da Seção 6.2, com a inclusão do tamanho do conjunto de subgrupos pequenos, $p = |Pequenos|$, para o qual sabemos que valem os seguintes limitantes:

$$2l - 3 \leq p \leq l + lt - 3t,$$

onde o limitante inferior é dado pelo caso em que todas as árvores em $\mathcal{T}$ são iguais e o limitante superior é dado pelo caso em que todas elas são diferentes, ou seja, têm subgrupos pequenos não triviais todos distintos.

Pela Tabela 6.1 fica claro que o tempo de execução do algoritmo será dominado pelo tempo de execução das linhas 4 e 8. Vemos então que o tempo de execução da rotina de preparação da coleção de árvores é

$$O(l^2 t \lg p + p^2),$$

o que para coleções de árvores em que p é mínimo equivale a $O(l^2 t \lg l)$, enquanto para coleções em que p é máximo equivale a $O(l^2 t \lg lt + l^2 t^2)$.

6.2.2 Construção da Árvore Mais Provável

Faremos a análise do tempo de execução do algoritmo Arvore-Mais-Provavel() de uma maneira um pouco diferente. Não analisaremos linha por linha, como no caso anterior, mas veremos o que acontece com cada par (A, B) analisado pelo algoritmo. A Tabela 6.2 apresenta todas as situações verificadas pelo algoritmo e as ações tomadas em cada caso. Note que em dois casos o tempo de execução permanece $O(l)$. Todos os demais, que envolvem uma busca no vetor de subgrupos *Pequenos*, têm complexidade $O(l \lg p)$.

É fácil perceber que o número total de pares analisado pelo algoritmo é dado por

$$\frac{p^2 - p}{2} = \Theta(p^2),$$

além disso, sabemos pela Tabela 6.2 a análise de qualquer par é feita em tempo $O(l \lg p)$, assim chegamos à conclusão que a complexidade da solução dos subgrupos no vetor de

$A \cap B = \emptyset$	$\lvert C \rvert \leq \lvert L \rvert /2$	$C \in Pequenos$	$C' < B$	$C' \in Pequenos$	**Ação**	**Tempo**
Não					descarta (A, B)	$\Theta(l)$
Sim	Não		Não		descarta (A, B)	$\Theta(l)$
Sim	Não		Sim	Não	descarta (A, B)	$O(l \lg p)$
Sim	Não		Sim	Sim	Atualiza-Floresta()	$O(l \lg p)$
Sim	Sim	Não			descarta (A, B)	$O(l \lg p)$
Sim	Sim	Sim			Atualiza-Subgrupo()	$O(l \lg p)$

Tabela 6.2: Casos nos quais pares de subgrupos podem se enquadrar no algoritmo, a ação executada para cada caso e o tempo gasto com cada ação durante toda a execução do algoritmo.

subgrupos pequenos é

$$O(p^2 l \lg p).$$

Ao todo, considerando a preparação da coleção de árvores filogenéticas, a complexidade do algoritmo que calcula a floresta de árvores mais prováveis para uma coleção de árvores filogenéticas completamente resolvida é

$$O(l^2 t \lg p + lp^2 \lg p).$$

Considerando novamente os dois extremos para o valor de p, chegamos à conclusão de que, para uma coleção de árvores iguais, o algoritmo é executado em $O(l^2 t \lg l + l^3 \lg l))$, enquanto no pior de todos os casos, o tempo gasto na resolução dos subgrupos pequenos predomina sobre o tempo de preparação da coleção de árvores, de forma que a complexidade do algoritmo acaba sendo $O(l^3 t^2 \lg lt)$.

6.3 O Algoritmo na Prática

Como o nosso objetivo neste projeto não se resume a propor uma definição de consenso completamente resolvido, mas também a comprovação de que consensos entre árvores filogenéticas podem servir para dar origem a árvores filogenéticas confiáveis, não basta para nós simplesmente propor um algoritmo para a determinação do conjunto de árvores mais prováveis de uma coleção de árvores filogenétifcas completamente resolvidas. Precisamos também implementar o algoritmo e verificar o seu desempenho e a qualidade das árvores filogenéticas encontradas por ele em relação ao conjunto de árvores filogenéticas criadas por métodos tradicionais de reconstrução de árvores. Por este motivo, o algoritmo foi implementado usando a linguagem `JAVA` e vários testes foram realizados com o intuito de obter medidas de desempenho do algoritmo. Nas próximas seções, veremos a medida deste desempenho em relação a duas dimensões: o tempo de processamento e a qualidade da saída.

Em relação ao tempo de processamento, foram usados dois conjuntos de árvores filogenéticas bastante distintos para a criação dos casos de teste. O primeiro deles é composto por 1.000 árvores filogenéticas criadas pela equipe de Gascuel [12]. Estas árvores foram criadas com o auxílio de softwares de simulação de evolução, utilizando parâmetros distintos para cada uma delas, o que faz com que a chance de duas árvores desta coleção serem muito semelhantes seja bem pequena. O segundo conjunto é composto pelas 945 árvores diferentes criadas para o segundo conjunto de seqüências artificiais descrito no Capítulo 5, usando o método *Máxima Parcimônia*, sendo, portanto, composto por árvores bastante semelhantes. Chamaremos estes dois conjuntos de conjunto de *árvores distintas* e conjunto de *árvores semelhantes*, respectivamente.

Cada rodada de testes consistia na execução de dois passos, sendo o primeiro a criação de 32 coleções de árvores filogenéticas, escolhendo-se aleatoreamente entre as árvores de um dos conjuntos originais de tal forma que a i-ésima coleção tenha exatamente $16 \times i$ árvores; e o segundo passo consiste em encontrar a floresta de árvores mais prováveis para cada uma das coleções criadas, tomando o cuidado de medir certas estatísticas durante a execução do algoritmo. Lembramos que tanto as árvores de entrada quanto as geradas pelo algoritmo são completamente resolvidas.

Em relação à qualidade da saída, os conjuntos de dados utilizados foram os mesmos do Capítulo 5. As conclusões tomadas com base nas estatísticas medidas são apresentadas no restante desta seção.

6.3.1 O Tamanho da Coleção de Subgrupos Pequenos

O tamanho do conjunto de subgrupos pequenos de cada caso de teste foi medido com o intuito de dar uma idéia da variação do valor de p conforme a proximidade entre as árvores da coleção. Esta preocupação justifica-se devido à influência de p no tempo de execução. As Tabelas 6.3 e 6.4 mostram as médias dos tamanhos de p medidos em cinco rodadas diferentes para os conjuntos de árvores distintas e semelhantes, respectivamente. As Figuras 6.4 e 6.5 mostram a mesma informação graficamente.

A primeira diferença notória entre os dados das duas tabelas é a diferença nos valores assumidos por p. Enquanto na Tabela 6.4 o maior valor assumido por p é menor que 900 e o maior valor é apenas 1,85 vezes o menor valor, na Tabela 6.3, o menor valor assumido por p já é maior que 1.500 e o maior valor é 23,7 vezes o menor valor. Outro fato interessante é que em ambos os casos, a porcentagem que o valor de p observado representa do maior valor de p possível tende a diminuir conforme o tamanho da coleção cresce. Além disso, a queda de porcentagem do valor máximo observada em conjuntos de árvores semelhantes conforme o tamanho da coleção de árvores filogenéticas aumenta é bem mais acentuada que a queda do mesmo valor observada em conjuntos de árvores

Árvores Distintas			
$\lvert\mathcal{T}\rvert$	p **médio**	p **máximo**	%
16	1.627,6	1.652	99%
32	3.111,0	3.204	97%
48	4.553,4	4.756	96%
64	5.970,4	6.308	95%
80	7.339,2	7.860	93%
96	8.678,6	9.412	92%
112	10.008,6	10.964	91%
128	11.287,2	12.516	90%
144	12.549,6	14.068	89%
160	13.793,8	15.620	88%
176	15.054,2	17.172	88%
192	16.245,2	18.724	87%
208	17.438,4	20.276	86%
224	18.647,0	21.828	85%
240	19.803,2	23.380	85%
256	20.986,4	24.932	84%
272	22.131,4	26.484	84%
288	23.290,2	28.036	83%
304	24.367,6	29.588	82%
320	25.519,2	31.140	82%
336	26.640,0	32.692	81%
352	27.735,4	34.244	81%
368	28.844,2	35.796	81%
384	29.951,6	37.348	80%
400	31.061,8	38.900	80%
416	32.186,4	40.452	80%
432	33.229,4	42.004	79%
448	34.326,8	43.556	79%
464	35.401,8	45.108	78%
480	36.422,6	46.660	78%
496	37.531,2	48.212	78%
512	38.580,8	49.764	78%

Tabela 6.3: Valor médio de p para conjuntos de árvores pouco semelhantes. Para cada tamanho de coleção de árvore, $|\mathcal{T}|$, foram criadas cinco coleções e calculada a média entre o tamanho do conjunto de subgrupos pequenos das 5 coleções, chamada de "p médio". A coluna "p máximo" indica o maior valor que p poderia assumir e a coluna "%" a porcentagem que o valor de p observado médio representa do máximo.

Árvores Semelhantes			
$\lvert\mathcal{T}\rvert$	p **médio**	p **máximo**	%
16	452,0	1.652	27%
32	523,2	3.204	16%
48	572,2	4.756	12%
64	597,2	6.308	9%
80	610,0	7.860	8%
96	636,0	9.412	7%
112	657,4	10.964	6%
128	653,8	12.516	5%
144	675,8	14.068	5%
160	683,8	15.620	4%
176	693,6	17.172	4%
192	693,8	18.724	4%
208	709,6	20.276	3%
224	722,4	21.828	3%
240	732,0	23.380	3%
256	736,6	24.932	3%
272	739,0	26.484	3%
288	753,0	28.036	3%
304	760,2	29.588	3%
320	764,8	31.140	2%
336	776,6	32.692	2%
352	779,4	34.244	2%
368	790,8	35.796	2%
384	792,4	37.348	2%
400	804,0	38.900	2%
416	806,4	40.452	2%
432	808,0	42.004	2%
448	818,4	43.556	2%
464	827,2	45.108	2%
480	827,0	46.660	2%
496	837,0	48.212	2%
512	837,4	49.764	2%

Tabela 6.4: Valor médio de p para conjuntos de árvores muito semelhantes. Para cada tamanho de coleção de árvore, $|\mathcal{T}|$, foram criadas cinco coleções e calculada a média entre o tamanho do conjunto de subgrupos pequenos das 5 coleções, chamada de "p médio". A coluna "p máximo" indica o maior valor que p poderia assumir e a coluna "%" a porcentagem que o valor de p observado médio representa do máximo.

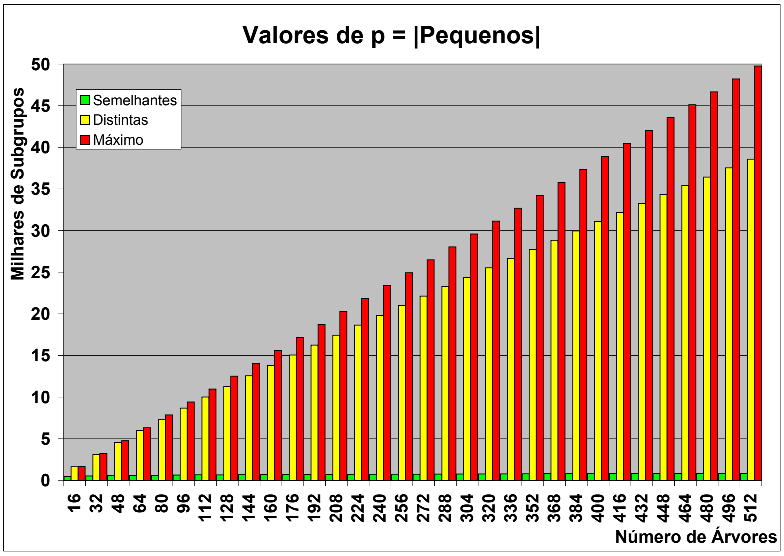

Figura 6.4: Tamanho médio do conjunto de subgrupos pequenos nas coleções usadas nos testes. As barras mais à direita de cada grupo, em vermelho, correspondem ao valor máximo que o conjunto de subgrupos pequenos poderia atingir, ou seja, $l + lt - 3t$. As barras centrais, em amarelo, correspondem ao tamanho médio do conjunto de subgrupos pequenos para conjuntos de árvores distintas e as colunas barras mais à esquerda, de cor verde, correspondem ao tamanho médio do conjunto de subgrupos pequenos para conjuntos de árvores semelhantes.

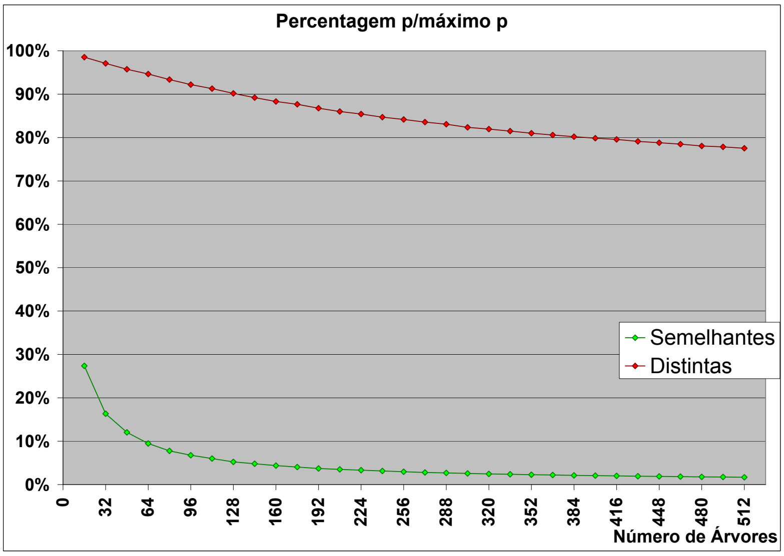

Figura 6.5: Porcentagem do tamanho observado do conjunto de subgrupos pequenos em relação ao valor máximo possível. Os pontos vermelhos correspondem aos conjuntos de subgrupos pequenos das coleções de árvores distintas, enquanto os pontos em verde representam os conjuntos equivalentes para árvores semelhantes.

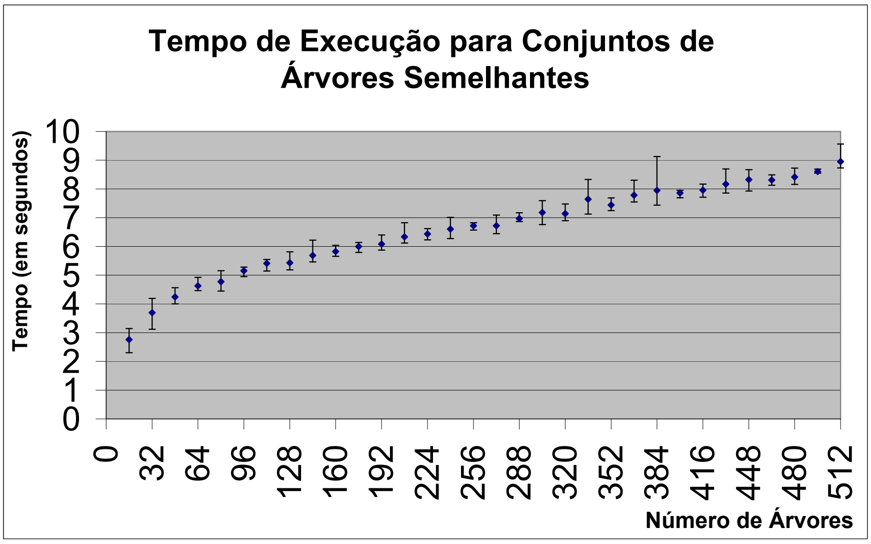

Figura 6.6: Tempo de execução do algoritmo para coleções de árvores semelhantes. No exemplo, apenas o número t de árvores varia, sendo o número l de folhas uma constante. Note que os tempos são da ordem de alguns segundos para até 500 árvores. Cada ponto representa a média dos tempos observado. Barras inferiores e superiores indicam, respectivamente, o menor e o maior tempo observados.

distintas, como sugere o gráfico da Figura 6.5.

Voltamos à questão dos dois limites superiores para o tempo de execução do algoritmo e lembramos também que este algoritmo foi projetado para trabalhar com conjuntos de árvores semelhantes na esperança de se conseguir uma árvore mais confiável. Na situação para a qual o algoritmo foi projetado, temos uma coleção de árvores filogenéticas mais próximas das coleções usadas para a coleta dos dados da Tabela 6.4 e nesta tabela nós observamos um comportamento de p mais próximo do limite inferior.

6.3.2 O Tempo de Execução

Como vimos na Seção 6.2.2, a complexidade do algoritmo é de $O(l^3t^2 \lg lt)$ ou então $O(l^2t \lg l + l^3 \lg l)$. O leitor pode estar se perguntando a esta altura porque apresentamos dois limites superiores para o tempo de execução, sendo que um deles não vale para todas as instâncias do problema. Os gráficos das Figuras 6.6 e 6.7 sugerem que, dependendo do tipo de entrada, ambas as fórmulas têm utilidade.

Figura 6.7: Tempo de execução do algoritmo para coleções de árvores distintas. No exemplo, apenas o número t de árvores varia, sendo o número l de folhas uma constante. Note que os tempos são da ordem de horas. Barras inferiores e superiores indicam, respectivamente, o menor e o maior tempo observados.

Lembrando que as árvores distintas têm um comportamento mais próximo do pior caso para o tamanho de p, enquanto as árvores semelhantes têm um comportamento mais próximo do melhor caso e que, normalmente, esperamos que as coleção de árvores produzidas por bons métodos de reconstrução estejam mais próximas das coleções de árvores semelhantes do que das coleções de árvores distintas, podemos afirmar que, apesar do limite assintótico superior $O(l^3t^2 \lg lt)$ ser o limitante correto para o tempo de execução do algoritmo, na prática, podemos esperar um comportamento da forma $O(l^2t \lg l + l^3 \lg l)$. Outro ponto a destacar é que para coleções de árvores distintas, o algoritmo leva cerca de 3.000 vezes mais tempo para encontrar as árvores mais prováveis do que para coleções de árvores semelhantes.

6.3.3 A Qualidade da Árvore Mais Provável

Para finalizar a avaliação da Árvore Mais Provável, apresentamos uma comparação dos resultados obtidos repetindo os casos de teste executados no Capítulo 5, desta vez substituindo o software `consense`, do pacote `PHYLIP`, pela nossa implementação do algoritmo que calcula a floresta de árvores mais prováveis. A Tabela 6.5 apresenta as distâncias das árvores consenso às demais árvores do conjunto de dados, tornando possível comparar a qualidade das árvores consenso.

A princípio os dados não mostram muita vantagem da Árvore Mais Provável em relação ao consenso anterior. As distâncias parecem ser as mesmas, ou se revezar, sendo hora menor para o consenso antigo, hora menor para a Árvore Mais Provável. No entanto, se prestarmos um pouco de atenção, veremos dois detalhes interessantes: (1) para o conjunto de dados artificiais número 1, as árvores obtidas pelos dois métodos eram exatamente iguais; isto explica a igualdade entre as duas colunas; (2) para o conjunto de dados número 3, embora as excentricidades das duas árvores sejam as mesmas, o isolamento da Árvore Mais Provável é menor, o que é um indício de que ela está mais próxima do conjunto do que a árvore consenso do `consense`. Isso é comprovado também pelo fato de a média das distâncias das árvores dos conjuntos para a Árvore Mais Provável ser sempre menor ou igual à mesma média para o consenso do software `consense`. Junte-se a isso o fato de que, no conjunto de dados reais, o novo método foi capaz de produzir uma saída para o conjunto de dados completo, ao contrário do `consense`.

A Tabela 6.6 compara os consensos de uma outra maneira, levando em conta a sua proximidade à árvore original em relação às árvores da coleção usada para construi-la. Notamos neste caso que, quando a comparação é possível, a Árvore Mais Provável pode ser menos eficiente que o consenso determinado pelo software `consense`, no sentido em que ela se distancia um pouco mais da árvore original, o que é mostrado pelos conjunto de dados artificiais número 3. Podemos notar, no entanto, que sua qualidade de modo geral é alta, o que é indicado pelo fato de ela normalmente estar mais próxima da árvore original que mais de 65% das árvores da coleção usada para criá-la. A única exceção é o conjunto de dados número 2, onde a Árvore Mais Provável só está mais próxima da árvore original que 33% das árvores da coleção.

Mesmo com alguma desvantagem em relação à proximidade da árvore original, a característica fundamental das Árvores Mais Prováveis, de serem árvores completamente resolvidas, acaba pondo-a em vantagem em relação às outras árvores consenso quando o objetivo é unir vários resultados diferentes para reconstruir uma filogenia, pois, além de evitar a perda de resolução através da criação de politomias, ela ainda apresentou a vantagem de estar mais centrada em relação à coleção de árvores filogenéticas do que o consenso obtido anteriormente. Claro que os testes ainda são poucos e pode haver métodos capazes de criar árvores consenso completamente resolvidas e que possam ser comparados com a Árvore Mais Provável no futuro, mas os poucos resultados que já possuímos são bastante animadores.

	CD1		CD2	CD3		CD4	REAIS		
	C	**M**	**M**	**C**	**M**	**M**	**C**	**M**	**M***
ORIG	**48**	**48**	118	**86**	88	108	**88**	88	**88**
FMEJ	**26**	**26**	66	**36**	38	64	**18**	50	22
FMEK	**26**	**26**	72	**40**	42	70	**26**	56	30
MMEJ	**30**	**30**	60	24	**20**	46	n/u	50	n/u
MMEK	**34**	**34**	62	**30**	34	40	n/u	50	n/u
MMET	**34**	**34**	66	**20**	24	40	n/u	60	n/u
MMP	**46**	**46**	112	**64**	72	96	n/u	76	n/u
MNJJ	**28**	**28**	74	24	**22**	56	n/u	52	n/u
MNJK	**18**	**18**	72	**36**	**36**	56	n/u	54	n/u
MNJT	**26**	**26**	78	**18**	**18**	56	n/u	58	n/u
PCO	**144**	**144**	118	**172**	**172**	112	n/u	126	n/u
PML	**44**	**44**	108	**72**	**72**	88	—	—	—
PMK	—	—	—	—	—	—	—	—	—
PMP	**46**	**46**	94	**66**	**66**	94	**74**	64	**74**
PNJJ	**16**	**16**	50	26	**24**	46	32	62	**28**
PNJK	**18**	**18**	50	22	**16**	56	32	62	**28**
PUPJ	**100**	**100**	106	122	**120**	102	n/u	50	n/u
PUPK	**102**	**102**	104	122	**120**	102	n/u	50	n/u
WNWJ	**22**	**22**	54	36	**26**	58	**38**	54	**38**
WNWK	**22**	**22**	54	32	**26**	62	n/u	58	n/u
MÉDIA	**43,68**	**43,68**	77,78	53,44	**52,67**	69,11	**36,67**	60,71	**36,67**
EX	**144**	**144**	118	**172**	**172**	112	**88**	126	**88**
IS	**16**	**16**	50	18	**16**	40	**18**	50	22

Tabela 6.5: Comparação entre as árvores consenso obtidas no Capítulo 5 e a Árvore Mais Provável para o mesmo conjunto de dados. As colunas "CDX" apresentam as distâncias de cortes para árvores do conjunto de dados artificiais X. A coluna "REAIS" apresenta as distâncias de cortes para árvores do conjunto de dados reais. Na segunda linha, a letra "C" indica dados do consenso obtido pelo software `consense` do pacote `PHYLIP`, enquanto a letra "M" indica dados da Árvore Mais Provável. No caso dos dados reais, a coluna "M*" refere-se ao conjunto reduzido usado na Tabela 5.8 (linhas sem "n/u"). Nos conjuntos CD2 e CD4 o software `consense` forneceu um consenso não completamente resolvido. Nos demais conjuntos, incluindo o de dados reais, os menores valores são mostrados em negrito. Os números das linhas correspondem aos métodos listados na Tabela 5.1, enquanto a linha indicada pela sigla "ORIG" indica as distâncias em relação à árvore original. Traços horizontais (—) indicam que o método de construção correspondente não foi capaz de construir uma árvore filogenética em tempo hábil. A sigla "n/u" indica que a árvore correpondente não foi usada no conjunto reduzido. A linha "MÉDIA" apresenta a média das distâncias do consenso às demais árvores, desconsiderando a árvore original. A linha "EX" apresenta as excentricidades das árvores consenso, enquanto a linha "IS" apresenta os isolamentos das mesmas. As definições de isolamento e excentricidade são apresentadas na página 62.

	CD1		CD2	CD3		CD4	REAIS		
	C	**M**	**M**	**C**	**M**	**M**	**C**	**M**	**M***
PERDE	5	5	11	2	4	5	0	0	0
EMPATA	0	0	1	2	2	1	1	1	1
GANHA	13	13	6	14	12	12	5	16	5
%	72%	72%	33%	78%	67%	67%	83%	94%	83%

Tabela 6.6: A linha "PERDE" apresenta o número de árvores na coleção que estão mais próximas da árvore original que o consenso, ou seja, o número de árvores para as quais o consenso perde. A partir desta definição, o significado das linhas "EMPATA" e "GANHA" são intuitivamente entendidos. Finalmente a linha "%" indica a porcentagem do número total de árvores da qual a árvore consenso "ganhou". Como na Tabela 6.5, aqui também as colunas "CDX" apresentam os valores para árvores do conjunto de dados artificiais X e a coluna "REAIS" apresenta os valores para o conjunto de dados reais. Na segunda linha, a letra "C" indica dados do consenso obtido pelo software `consense` do pacote `PHYLIP`, enquanto a letra "M" indica dados da Árvore Mais Provável. Para o conjunto de dados reais, temos novamente a coluna "M*", contendo dados da Árvore Mais Provável construída com base no mesmo conjunto de árvores utilizado para a construção do consenso usando o software `consense`. Conjuntos de dados que não têm uma coluna "C" são conjuntos de dados para os quais o software `consense` forneceu um consenso, porém não completamente resolvido.

Capítulo 7

Conclusão

Embora esta dissertação tenha dois objetivos maiores, que são a verificação da utilidade de consensos entre árvores filogenéticas como meio de criação, e não mera comparação, de árvores filogenéticas e a proposta de um consenso completamente resolvido entre árvores filogenéticas completamente resolvidas, cada capítulo aborda um tema em particular, de modo que as principais conclusões deste trabalho já foram apresentadas ao final de cada capítulo. Este capítulo se dedica somente a resumir as principais conclusões tomadas e as contribuições à área apresentadas ao longo desta dissertação.

No Capítulo 2, apresentamos as duas formas de representação mais comuns de árvores filogenéticas como subconjuntos de um conjunto de folhas. Apesar de ambos, os sistemas de cortes e as n-árvores, serem utilizados há algum tempo em estudos de árvore filogenética, relações entre as duas formas de representação, tais como as apresentadas na Seção 2.3, foram pouco ou nada exploradas até o momento. Apresentamos neste capítulo uma nova prova para a fórmula da distância de cortes, uma caracterização para n-árvores completamente resolvidas (Teorema 2.2.3) e uma caracterização para árvores filogenéticas completamente resolvidas baseada nos subgrupos pequenos dos cortes determinados por suas arestas no conjunto de folhas (Teorema 2.3.5). Também no Capítulo 2, mostramos que basta uma enumeração dos elementos de um conjunto qualquer L para que seja possível estabelecer uma relação de ordem também entre subgrupos definidos sobre o conjunto L (Teorema 2.3.2).

A justificativa para a utilidade desta dissertação é deixada para o Capítulo 5, onde são apresentadas avaliações da qualidade de árvores consenso completamente resolvidas em comparação com as árvores utilizadas para construi-las, construídas através de métodos tradicionais de reconstrução de árvores filogenéticas. Fica claro, através dos resultados apresentados neste capítulo, que o consenso completamente resolvido gerado pelo software `consense` do pacote `PHYLIP` não é pior que as árvores utilizadas para construí-lo, como trabalhos como o de Sharkey e Leathers [26] sugerem. Pelo contrário, árvores consenso completamente resolvidas normalmente tendem a estar entre as árvores que mais se aproximam da árvore original.

Finalmente, no Capítulo 6, apresentamos a definição de um consenso completamente resolvido de uma coleção de árvores filogenéticas completamente resolvidas, bem como um algoritmo capaz de calcular todas as árvores que satisfazem a definição para uma coleção de árvores filogenéticas completamente resolvidas. Vimos que a complexidade do algoritmo depende basicamente do número de subgrupos distintos que podem ser encontrados na coleção de árvores e que, de acordo com os testes apresentados na Seção 6.3, o crescimento no número destes subgrupos tende a ser linear no tamanho da coleção de árvores, podendo se aproximar de um crescimento logarítmico quando a coleção é formada por árvores bastante semelhantes.

7.1 Trabalhos Futuros

Embora apresentem resultados animadores, os testes com árvores filogenéticas construídas e o consenso entre elas ainda apresenta alguns problemas cujas soluções dariam mais credibilidade a seus resultados. Um deles é o fato de apenas um dos conjuntos de testes ter sido construído com base em seqüências reais. Testes com dados artificiais são comumente usados para testar métodos de construção de árvores filogenéticas principalmente por ser o único tipo de dado que fornece a árvore original para comparação. Por outro lado, por mais perfeitas que sejam, seqüências artificiais não simulam todos os casos ocorrido na natureza. Seria interessante que mais testes fossem feitos utilizando dados reais.

Além disso, o fato de termos usado apenas métodos baseados em seqüências de DNA pode distorcer um pouco os resultados. Acreditamos que a inclusão de outras árvores, construídas usando métodos não baseados em seqüências de DNA poderiam dar uma visão melhor do poder das árvores consenso.

A árvore mais provável, por fim, pode ser estudada mais a fundo. Algumas variações são bem fáceis de definir, como a árvore mais provável com pesos, onde cada árvore da coleção teria um peso, representando a confiança depositada no fato de ela estar correta ou não, e a construção da árvore mais provável levaria em conta este peso. Outra variação poderia ser criada tentando modelar o fato de que escolhas de subgrupos não são eventos independentes.

Para finalizar, a árvore mais provável poderia ainda ser comparada com os outros consensos já definidos para árvores filogenéticas. Seria interessante saber, por exemplo, qual a relação entre a árvore mais provável e as árvores obtidas pela regra da maioria ou a mediana em relação à métrica determinada pela distância por cortes entre árvores.

Apêndice A

Notação Utilizada

$\emptyset$ - Conjunto vazio.

$\wedge$ - Operador "e" lógico bit a bit.

$\vee$ - Operador "ou" lógico bit a bit.

α - Relação entre elementos de $\Theta(L)$; Probabilidade de ocorrência um evento de mutação num intervalo de tempo determinado no modelo de Jukes-Cantor; Probabilidade de um evento de transição num intervalo de tempo determinado no modelo de dois parâmetros de Kimura.

β - Probabilidade de um evento de transversão num intervalo de tempo determinado no modelo de dois parâmetros de Kimura.

$d(v)$ - Grau do vértice v.

$E(G)$ - Conjunto de arestas de um grafo G qualquer.

η - Função de altura (definida para subgrupos).

η_0 - Função de altura canônica.

$\eta_\sharp$ - Função de altura de cardinalidade.

$\mathcal{F}(T)$ - Conjunto dos subgrupos pequenos dos cortes em $\mathcal{S}_T$.

$\mathcal{F}(\mathcal{T})$ - Conjunto de todos os subgrupos pequenos de árvores filogenéticas encontradas numa mesma coleção $\mathcal{T}$ de árvores filogenéticas completamente resolvidas.

φ - Função $\varphi : L \to \mathbb{N}_L$ que enumera os elementos de L.

$\varphi(R)$ - Conjunto dos valores $\varphi(r)$ para $r \in R$ e $R \subseteq L$.

$G_{\Theta(L)}$ - Grafo em que os vértices são os elementos de $\Theta(L)$ e há arestas ligando quaiquer dois vértices relacionados por α.

L - Um conjunto qualquer de espécies.

$\mathbb{N}$ - Conjunto dos números naturais.

$\mathbb{N}_L$ - Conjunto dos $|L|$ primeiros números naturais.

$\mathcal{N}_T$ - Conjunto dos subgrupos relativos de todos os vértices de uma árvore filogenética qualquer $T \in \mathcal{T}_R(L)$.

$p(C, \mathcal{T})$**, onde** C **é um corte** - Freqüência relativa do corte C na coleção de árvores filogenéticas completamente resolvidas $\mathcal{T}$. É definida formalmente como $\frac{|\{T \in \mathcal{T} \ | \ C \in \mathcal{S}(T)\}|}{|\mathcal{T}|}$.

$p(T, \mathcal{T})$**, onde** T **é uma árvore filogenética completamente resolvida** - Peso da árvore filogenética completamente resolvida T em relação à coleção de árvores filogenéticas completamente resolvidas $\mathcal{T}$. É definida formalmente como $\prod_{C \in \mathcal{S}(T)} p(C, \mathcal{T})$.

$p(S, \mathcal{T})$**, onde** S **é um subgrupo** - Freqüência relativa do subgrupo S na coleção de árvores filogenéticas completamente resolvidas $\mathcal{T}$. É definida formalmente como $\frac{|\{T \in \mathcal{T} \ | \ S \in \mathcal{F}(T)\}|}{|\mathcal{T}|}$.

$p(\Psi, \mathcal{T})$**, onde** Ψ **é uma** n**-árvore** - Peso da n-árvore Ψ em relação à coleção de árvores filogenéticas completamente resolvidas $\mathcal{T}$. É definida formalmente como $\prod_{S \in \Psi} p(S, \mathcal{T})$.

Ψ - Uma n-árvore qualquer.

$\Psi[S]$ - A n-subárvore induzida pelo subgrupo S em Ψ. Esta notação também é usada no caso em que o conjunto de subgrupos Ψ não é uma n-árvore.

$\rho(T, U)$ - Comprimento do menor caminho entre $\mathcal{S}_L(U)$ e $\mathcal{S}_L(T)$ no grafo $G_{\Theta(L)}$.

$\mathcal{S}_L(T)$ - Sistema de cortes definido pelas arestas da árvore filogenética T sobre o conjunto L. Também denotado por $\mathcal{S}(T)$, para fins de simplificação.

$\mathcal{S}(T)$ - Notação simplificada do conjunto $\mathcal{S}_L(T)$ quando T é uma árvore filogenética e L é um conjunto bem definido no contexto.

$\mathcal{S}(\mathcal{T})$ - Conjunto de todos os cortes definidos pelas arestas de árvores filogenéticas encontradas numa mesma coleção $\mathcal{T}$ de árvores filogenéticas completamente resolvidas.

$\mathcal{S}^*(L)$ - Sistema de cortes formado por todos os cortes triviais do conjunto L.

$\mathcal{S}(X)$ - O conjunto de todos os cortes possíveis definidos sobre o conjunto X.

S_g - Subgrupo grande de um corte S.

S_p - Subgrupo pequeno de um corte S.

S_v - O subgrupo relativo a um vértice v qualquer de uma árvore filogenética qualquer $T \in \mathcal{T}_R(L)$.

T - Normalmente, uma árvore filogenética pertencente a $\mathcal{T}(L)$.

$\mathcal{T}$ - Uma coleção qualquer de árvores filogenéticas completamente resolvidas.

$\mathcal{T}(L)$ - Conjunto de todas as árvores filogenéticas cujas folhas são (rotuladas por) elementos de L.

$\Theta(L)$ - Conjunto de todos os sistemas de cortes definidos sobre L que contém $\mathcal{S}^*(L)$ e cujos cortes são dois a dois compatíveis.

$\mathcal{T}^*(L)$ - Maior subconjunto de $\mathcal{T}(L)$ que contém apenas árvores filogenéticas completamente resolvidas.

$\mathcal{T}_R(L)$ - Maior subconjunto de $\mathcal{T}(L)$ que contém apenas árvores filogenéticas com raiz.

$\mathcal{T}_R^*(L)$ - Maior subconjunto de $\mathcal{T}(L)$ que contém apenas árvores filogenéticas completamente resolvidas com raiz.

$\mathcal{T}_U(L)$ - Maior subconjunto de $\mathcal{T}(L)$ que contém apenas árvores filogenéticas sem raiz.

$\mathcal{T}_U^*(L)$ - Maior subconjunto de $\mathcal{T}(L)$ que contém apenas árvores filogenéticas completamente resolvidas sem raiz.

$V(G)$ - Conjunto dos vértices de um grafo G qualquer.

Apêndice B

Lista de Espécies Presentes nos Testes com Seqüências de rRNA

1. ***Acanthamoeba castellanii*** (amoeba)
2. ***Agrocybe aegerita str. SM47*** (gill mushroom; basidiomycete fungus)
3. ***Ailuropoda melanoleuca*** (giant panda)
4. ***Albinaria coerulea*** (land snail)
5. ***Alligator mississippiensis*** (American alligator)
6. ***Allomyces macrogynus str. his1 3-35*** (35oC) (ATCC 46923) (fungus)
7. ***Anas platyrhynchos str. Tsai ya*** (Peking duck)
8. ***Anopheles gambiae str. G3*** (mosquito)
9. ***Apis mellifera subsp. ligustica*** (western honey bee)
10. ***Arbacia lixula*** (black urchin)
11. ***Artemia franciscana*** (brine shrimp)
12. ***Ascaris suum*** (nematode)
13. ***Asterina pectinifera*** (starfish)
14. ***Balaenoptera physalus*** (fin whale or common rorqual)
15. ***Bos taurus*** (ox; cow)

16. ***Caenorhabditis elegans*** (nematode)
17. ***Canis familiaris*** (dog)
18. ***Chlamydomonas eugametos*** (unicellular green alga)
19. ***Chlamydomonas reinhardtii str. CW-15*** (unicellular green alga)
20. ***Chondrus crispus*** (red alga (=carragheen ?))
21. ***Chorthippus parallelus str. ESC*** (meadow grasshopper)
22. ***Coturnix coturnix*** (Japanese quail)
23. ***Crocodylus acutus*** (crocodile)
24. ***Crossostoma lacustre*** (freshwater loach (=fish))
25. ***Crypturellus undulatus*** (tinamou bird)
26. ***Cyprinus carpio*** (carp)
27. ***Daphnia pulex*** (water flea)
28. ***Dictyostelium discoideum str. AX3*** (cellular slime mold)
29. ***Didelphis virginiana*** (Virginian opossum)
30. ***Drosophila melanogaster*** (fruit fly)
31. ***Drosophila yakuba str. 2317.6 Ivory Coast*** (fruit fly)
32. ***Emericella nidulans*** (ascocarp fungus)
33. ***Eptesicus fuscus*** (brown bat)
34. ***Fulica atra*** (Eurasian coot)
35. ***Gallus gallus*** (chicken)
36. ***hlorarachnion sp.*** (CCMP 621) (amoeboflagellate)
37. ***Homo sapiens*** (human)
38. ***Katharina tunicata*** (black chiton)
39. ***Latimeria chalumnae*** (coelacanth)

40. ***Locusta migratoria*** (migratory locust)

41. ***Lumbricus terrestris*** (common earthworm)

42. ***Metridium senile*** (sea anemone)

43. ***Mus musculus*** (common or house mouse)

44. ***Mytilus edulis*** (blue mussel)

45. ***Neoceratodus foresteri*** (Australian lungfish)

46. ***Neurospora crassa str. 74-OR23-1A*** (FGSC 2489) (red bread mold; ascomycete fungus)

47. ***Odocoileus virginianus*** (white-tailed deer)

48. ***Oncorhynchus mykiss*** (rainbow trout)

49. ***Ornithorhynchus anatinus*** (duckbill platypus)

50. ***Panthera leo*** (lion)

51. ***Paracentrotus lividus*** (sea urchin)

52. ***Paramecium tetraurelia*** (ciliate)

53. ***Petromyzon marinus*** (sea lamprey)

54. ***Phoca vitulina*** (harbor seal)

55. ***Physarum polycephalum str. M3C*** (acellular slime mold)

56. ***Prototheca wickerhamii*** (SAG 263-11) (colorless unicellular alga)

57. ***Pylaiella littoralis*** (brown alga)

58. ***Rana catesbeiana*** (bull frog)

59. ***Rhea americana*** (greater rhea)

60. ***Saccharomyces cerevisiae*** (brewer's yeast; baker's yeast)

61. ***Salmo salar*** (Atlantic salmon)

62. ***Sceloporus undulatus*** (North American desert lizard; fence lizard)

63. ***Schizosaccharomyces japonicus*** (fission yeast)

64. ***Schizosaccharomyces pombe*** (fission yeast)

65. ***Sphenodon punctatus*** (tuatara (=reptile))

66. ***Stenella coeruleoalba*** (striped dolphin)

67. ***Strongylocentrotus purpuratus*** (sea urchin)

68. ***Tetrahymena pyriformis str. ST*** (ciliate)

69. ***Trachemys scripta*** (red-eared slider turtle; slider turtle; dime-store turtle)

70. ***Trypanosoma brucei*** (African trypanosome)

71. ***Zalophus californianus*** (California sea lion)

72. ***Zea mays*** (maize)

Apêndice C

Pacotes Utilizados nos Testes

Este apêndice apresenta uma pequena descrição dos pacotes e softwares utilizados nos testes dos Capítulos 5 e 6.

C.1 FastMe

O software `fastMe` [7] foi desenvolvido por Richard Desper e Olivier Gascuel e baseia-se no princípio de evolução mínima para reconstruir árvores filogenéticas. O artigo que o descreve foi publicado em 2002.

O código fonte, um executável para MS-DOS e arquivos de teste estão disponíveis gratuitamente no site `http://www.lirmm.fr/~w3ifa/MAAS/FastME/FastME.html`. Também estão disponíveis no site versões "draft" de artigos publicados a respeito do software.

C.2 Mega

O pacote `Mega` [15] foi desenvolvido por Sudhir Kumar, Koichiro Tamura e Masatoshi Nei. Trata-se de uma ferramenta integrada para alinhamento automático e manual de seqüências, reconstrução de árvores filogenéticas, busca em bancos de dados baseados em web, estimativas de taxas de evolução e testes de hipóteses evolucionárias.

A primeira versão do software foi criada em 1993. A versão usada no projeto, cuja primeira distribuição data de 14 de julho de 2003, foi a versão 3 para Windows, distribuída através do site `http://www.megasoftware.net/`. No site há também versões para MS-DOS, Mac OS 9 e Mac OS X. O pacote é distribuído gratuitamente para fins educacionais e de pesquisa.

C.3 PHYLIP

O pacote `PHYLIP` [9] está na versão 3.6, que data de 2 de dezembro de 2004. Ele foi desenvolvido por Joe Felsenstein e sua primeira versão apareceu em outubro de 1980. Ele é distribuído através do site `http://evolution.gs.washington.edu/phylip.html` e é gratuito. No site é possível encontrar tanto o código fonte quanto executáveis para Windows, Mac OS 8 ou 9, e Mac OS X, sendo que o código fonte, em `C`, pode ser facilmente compilado em sistemas UNIX e Linux.

Neste pacote é possível encontrar programas que usam os mais variados métodos para reconstruir árvores filogenéticas, como maximização de parcimônia, métodos baseados em distâncias e maximização de verossimilhança, entre outros. Os programas também aceitam vários tipos de dados de entrada, incluindo seqüências de DNA e RNA, seqüências de proteínas, sítios de restrição, caracteres discretos 0/1, freqüências de genes, caracteres contínuos e matrizes de distâncias.

C.4 Weighbor

O software `weighbor` [3] implementa uma versão ponderada de um método bastante conhecido de reconstrução de árvores filogenéticas chamado Neighbor-Joining. Ele foi desenvolvido por William J. Bruno, Nicholas D. Socci e Aaron L. Halpern no ano 2000 e a versão utilizada neste projeto foi a versão 1.2.1. O software está disponível gratuitamente no site `http://www.t10.lanl.gov/billb/weighbor/`.

Bibliografia

[1] Arne Anderberg and Anders Tehler. Consensus trees, a necessity in taxonomic practice. *Cladistics*, 6:399–402, 1990.

[2] Kare Bremer. Combinable component consensus. *Cladistics*, 6:369–372, March 1990.

[3] William J. Bruno, Nicholas D. Socci, and Aaron L. Halpern. Weighted neighbor joining: A likelihood-based approach to distance-based phylogeny reconstruction. *Molecular Biology and Evolution*, 17(1):189–197, 2000.

[4] David Bryant. A classification of consensus methods for phylogenetics. In M. Janowitz, F. J. Lapointe, F. McMorris, B. Mirkin, and F. Roberts, editors, *Bioconsensus*, pages 163–184. DIMACS-AMS, 2003.

[5] J. R. Cole, B. Chai, T. L. Marsh, R. J. Ferris, Q. Wang, S. A. Kulam, S. Chandra, D. M. McGarrell, T. M. Schmidt, G. M. Garrity, and J. M. Tiedje. The Ribosomal Database Project (RDP-II): previewing a new autoaligner that allows regular updates and the new prokaryotic taxonomy. *Nucleic Acids Research*, 31(1):442–443, Jan 2003.

[6] Charles Darwin. *The Origin of Species*. Oxford World´s Classics. Oxford University Press, 1998. Edited with an Introduction and Notes by Gillian Beer, ISBN: 0-19-283438-X.

[7] Richard Desper and Olivier Gascuel. Fast and accurate phylogeny reconstruction algorithms based on the minimum-evolution principle. *Journal of Computational Biology*, 19(5):687–705, 2002.

[8] Andreas W. M. Dress. Recent results and new problems in phylogenetic combinatorics. Technical report, University of Bielefeld, December 2001.

[9] Joe Felsenstein. PHYLIP - phylogeny inference package (version 3.2). *Cladistics*, 5:164–166, 1989.

[10] Walter M. Fitch. Cladistic and other methods: Problems, pitfalls, and potentials. In T. Duncan and T. F. Stuessy, editors, *Cladistics: Perspectives on the Reconstruction of Evolutionary History*, chapter 12, pages 221–252. Columbia University Press, 1984.

[11] Naveen Garg, Rohit Khandekar, and Kunal Talwar. Covering a laminar family (extended abstract). http://citeseer.ist.psu.edu/480128.html, 2000. Indian Institute of Technology, Delhi, New Delhi, India.

[12] Olivier Gascuel. Methods and algorithms in bioinformatics. http://www.lirmm.fr/∼w3ifa/MAAS/US-MAAS.html, 2003.

[13] Dan Graur and Wen-Hsiung Li. *Fundamentals of Molecular Evolution*. Sinauer Associates, Inc., second edition, 1999.

[14] Ian J. Kitching, Peter L. Forey, Christopher J. Humpries, and David M. Williams. *Cladistics: The Theory and Practice of Parsimony Analysis*. Number 11 in The Systematics Association Publication. Oxford University Press, second edition, 1998. ISBN: 0-19-850139-0 (Hbk), 0-19-850138-2 (Pbk).

[15] Sudhir Kumar, Koichiro Tamura, and Masatoshi Nei. MEGA3: Integrated software for molecular evolutionary genetics analysis and sequence alignment. *Briefings in Bioinformatics*, 5:150–163, 2004.

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

[17] Masatoshi Nei. *Molecular Population Genetics and Evolution*, volume 40 of *Frontiers of Biology*. North-Holland Publishing Company, 1975.

[18] Roderic D. M. Page and Edward C. Holmes. *Molecular Evolution: A Phylogenetic Approach*. Blackwell Science Ltd, 1998.

[19] David Penny, Bennet J. McComish, Michael A. Charleston, and Michael D. Hendy. Mathematical elegance with biochemical realism: The covarion model of molecular evolution. *Journal of Molecular Evolution*, 53:711–723, 2001.

[20] Cynthia Philips and Tandy J. Warnow. The asymmetric median tree – a new model for building consensus tree. *Discrete Applied Mathematics*, 71:311–335, 1996.

[21] R. C. Powers. Intersection rules for consensus n-trees. *Applied Mathematics Letters*, 8(4):51–55, 1995.

[22] Donald L. J. Quicke. *Principles and Techniques of Contemporary Taxonomy*. Blackie Academic & Professional, first edition, 1993.

[23] Andrew Rambaut and Nicholas C. Grassly. Seq-Gen: an application for the Monte Carlo simulation of DNA evolution along phylogenetic trees. *CABIOS*, 13(3):235–238, 1997.

[24] D. F. Robinson and L. R. Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53:131–147, 1981.

[25] João Carlos Setubal and João Meidanis. *Introduction to Computational Molecular Biology*. PWS Publishing Company, 1997. ISBN: 0-534-95262-3.

[26] Michael J. Sharkey and Jason W. Leathers. Majority does not rule: The trouble with majority-rule consensus trees. *Cladistics*, 17:282–284, 2001.

[27] Koichiro Tamura and Masatoshi Nei. Estimation of the number of nucleotide substitutions in the control region of mitochondrial dna in humans and chimpanzees. *Molecular Biology and Evolution*, 10(3):512–526, 1993.

[28] Michael S. Waterman. *Introduction to Computational Biology*. Chapman & Hall, 1995.

[29] David L. Wheeler, Colombe Chappey, Alex E. Lash, Detlef D. Leipe, Thomas L. Madden, Gregory D. Schuler, Tatiana A. Tatusova, and Barbara A. Rapp. Database resources of the National Center for Biotechnology Information. *Nucleic Acid Research*, 28(1):10–14, 2000.