

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



Biblioteca de Programas para Análise de Genética Populacional

Marco Alexandre do Nascimento Alves

Mestrado Integrado em Engenharia Informática e Computação

Orientadora: Luísa Maria de Sousa Mesquita Pereira (Doutora)

Co-orientador: Rui Carlos Camacho de Sousa Ferreira da Silva (Professor)

Julho de 2011

Biblioteca de Programas para Análise de Genética Populacional

Marco Alexandre do Nascimento Alves

Mestrado Integrado em Engenharia Informática e Computação

Aprovado em provas públicas pelo Júri:

Presidente: António Manuel Correia Pereira (Doutor)

Vogal Externo: Miguel Francisco Almeida Pereira Rocha (Doutor)

Orientadora: Luísa Maria de Sousa Mesquita Pereira (Doutora)

14 de Julho de 2011

Resumo

A Análise de Fundador é uma metodologia de análise em Genética Populacional que tem como objectivo calcular os tempos de migração das populações humanas ao longo da Pré-História e História. Utiliza para esse efeito a informação genética de indivíduos de diferentes áreas geográficas, traduzida por árvores e redes filogenéticas, sendo o tempo de separação das populações calculado através da acumulação de diferenças genéticas (mutações).

Apesar de o artigo que descreve esta metodologia ser bastante citado, esta metodologia não voltou a ser muito aplicada devido ao facto de obrigar a um trabalhoso processo de geração de dados que, se for realizado manualmente, demora em média alguns meses. Pretende-se com este projecto criar uma aplicação de *software* que seja capaz de automatizar a aplicação desta metodologia, de forma a tornar a sua utilização mais rápida, mais fácil e mais fiável.

Pretende-se com esta dissertação documentar o trabalho, concepção e implementação da aplicação da metodologia da Análise de Fundador. É apresentado o tema geral da tese, bem como o estado da arte do trabalho na área e os resultados experimentais obtidos com a aplicação criada.

Abstract

Founder Analysis is a methodology for population genetics analysis, which aims to estimate migration times of human populations along history and prehistory. It uses genetic information of individuals (mutations) from different geographical areas, represented by phylogenetic trees and networks. In this way, the separation time between populations is calculated through the accumulation of genetic differences (mutations).

Despite the original paper that describes this methodology being highly cited, the Founder Analysis has not been applied because of the hard work required, if applied manually, which takes over a few months. In this project, we will create a software application that will automatize the application of this methodology, in order to render its application easier, quicker and more reliable.

This dissertation aims to document the work, development and conception of this software application of Founder Analysis. We will present the thesis theme, as well as the state of the art of related work and the experimental results.

Agradecimentos

Em primeiro lugar, gostaria de agradecer à Doutora Luísa Pereira, orientadora e responsável pelo acompanhamento na instituição e ao Professor Rui Camacho, orientador na FEUP, por todo o apoio e dedicação que me deram ao longo do período de realização da dissertação. Apesar de não ter sido meu orientador oficial, não quero deixar de realçar e agradecer toda a ajuda e orientação prestada pelo Pedro Soares, pós-doc do Grupo de Diversidade Genética e perito na análise de fundador.

Gostaria, igualmente, de agradecer ao IPATIMUP por esta oportunidade, em particular a todos os restantes elementos do Grupo de Diversidade Genética que me providenciaram, para além do apoio na compreensão de novos conceitos da biologia, um agradável ambiente de trabalho, tornando a realização deste projecto ainda mais cativante do que o expectável.

Por último, mas não menos importante, não me posso esquecer de toda a minha família, em particular dos meus pais e do meu irmão, que sempre me apoiaram e sempre fizeram tudo o que estaria ao seu alcance para contribuir para o meu crescimento e ainda a minha namorada que com o seu apoio, afecto e paciência nas piores horas, contribuiu para um melhor desempenho da minha parte.

A todos... Um MUITO OBRIGADO!

Marco Alves

Conteúdo

1	Introdução	1
1.1	Projecto	2
1.2	Motivação.....	3
1.3	Estrutura da Dissertação.....	3
2	Conceitos Teóricos da Área.....	5
2.1	DNA.....	5
2.2	DNA Mitocondrial.....	7
2.3	Alinhamento de Sequências.....	9
2.4	Análise Filogenética	10
2.5	Análise de Fundador.....	12
2.6	Resumo.....	14
3	Revisão Bibliográfica	17
3.1	Emparelhamento de Sequências.....	17
3.1.1	Análise da Matriz de Pontos.....	19
3.1.2	Algoritmos de Programação Dinâmica.....	20
3.1.3	Dicionário de Palavras ou K-Tuplos	21
3.2	Grafos.....	21
3.3	Árvores.....	26
3.4	Algoritmos Utilizados na Área	28
3.5	Estado da Arte da Análise de Fundador	30
3.5.1	Informatização da Análise de Fundador	31
3.6	Resumo.....	32
4	Redes e Árvores Filogenéticas	33
4.1	Implementação das Árvores e Redes.....	33
4.2	Construção de Árvores a Partir de Redes.....	35
4.2.1	Algoritmo de Prim	36
4.2.2	Algoritmo de Dijkstra.....	37
4.2.3	Melhoria aos Algoritmos	37
4.3	Algoritmo de Ranking de Árvores	38

Conteúdo

4.4 Resultados Experimentais	39
4.4.1 Caso de Teste 1.....	39
4.4.2 Caso de Teste 2.....	42
4.4.3 Caso de Teste 3.....	46
4.4.4 Caso de Teste 4.....	49
4.4.5 Caso de Teste 5.....	52
4.5 Resumo.....	55
5 Estatísticas.....	59
5.1 Implementação das Estatísticas.....	59
5.2 Resultados Experimentais	60
5.2.1 Caso de Teste 1.....	61
5.2.2 Caso de Teste 2.....	65
5.2.3 Caso de Teste 3.....	69
5.2.4 Caso de Teste 4.....	73
5.2.5 Caso de Teste 5.....	77
5.3 Resumo.....	81
6 Aplicação Desenvolvida.....	83
6.1 Casos de Utilização	83
6.2 Interface da Aplicação	85
6.2.1 Módulo de Geração de Árvores.....	85
6.2.2 Módulo de Escolha de Arestas	86
6.2.3 Módulo de Limpeza de <i>Median Vectors</i>	86
6.2.4 Módulo de estatísticas	87
6.3 Informação Adicional	88
7 Conclusões e Trabalho Futuro	89
7.1 Satisfação dos Objectivos.....	89
7.2 Trabalho Futuro.....	91
Referências.....	93
A Utilização do Network.....	97
A.1 Descrição da Utilização do Network 4.6.....	97
A.2 Formato RDF.....	102
A.3 Formato OUT.....	104
A.4 Formato FDI.....	108
B Detalhes da Aplicação	113
B.1 Guia de Utilização.....	113
B.2 Diagrama de Classes.....	117
B.3 Ficheiro Auxiliar com Frequências das Mutações	118

Conteúdo

B.4	Ficheiro Auxiliar com Informação das Populações.....	120
-----	--	-----

Lista de Figuras

Figura 2.1.1: Estrutura em dupla hélice da molécula de DNA com ligações entre bases azotadas	6
Figura 2.2.1: Ilustração representativa da mitocôndria.	7
Figura 2.2.2: Esquema representativo da molécula de mtDNA humano, com as localizações dos vários genes na região codificante e a região de controlo (<i>D-loop</i>).	8
Figura 2.4.1: Possíveis representações filogenéticas.	10
Figura 2.5.1: Representação de uma rede filogenética com o <i>software</i> Network 4.6.	12
Figura 2.5.2: Diversidade das populações de origem e de destino.	13
Figura 2.5.3: Potenciais fundadores obtidos através do cruzamento das populações de origem e destino.	14
Figura 3.1.1: Comparação de duas sequências.	17
Figura 3.1.2: Alinhamento Global.	19
Figura 3.1.3: Alinhamento Local.	19
Figura 3.1.4: Matriz de pontos com várias diagonais e alguns pontos dispersos.	20
Figura 3.2.1: Grafo não-dirigido.	21
Figura 3.2.2: Grafo dirigido.	22
Figura 3.2.3: Problema das Sete Pontes de Königsberg – Euler, 1735.	22
Figura 3.2.4: Problema das Sete Pontes de Königsberg traduzido para grafo.	22
Figura 3.2.5: Exemplo de listas de adjacências.	25
Figura 3.3.1: Árvore com 6 nós e 5 arestas.	27
Figura 3.3.2: Árvore com raiz.	27
Figura 3.3.3: Exemplo de Árvore.	28
Figura 3.5.1: Diferentes critérios de cálculo de fundadores.	30
Figura 4.1.1: Diferenças na aplicação dos algoritmos de Prim e Dijkstra.	34
Figura 4.2.1: Tipo de ciclos (reticulações) presentes nas redes filogenéticas, diferentes tamanhos de nós e diferentes custos das arestas.	36
Figura 4.2.2: Remoção de nós com caminho único.	38
Figura 4.4.1: Caso de teste 1 – Rede filogenética.	39
Figura 4.4.2: Caso de teste 1 – Primeira árvore filogenética obtida.	40

Lista de Figuras

Figura 4.4.3: Caso de teste 1 – Segunda árvore filogenética obtida.	41
Figura 4.4.4: Caso de teste 1 – Terceira árvore filogenética obtida.	42
Figura 4.4.5: Caso de teste 2 – Rede filogenética.	43
Figura 4.4.6: Caso de teste 2 – Árvore filogenética obtida com o algoritmo de Prim.	44
Figura 4.4.7: Caso de teste 2 – Árvore filogenética obtida com o algoritmo de Dijkstra.	45
Figura 4.4.8: Caso de teste 2 – Solução proposta pelo especialista da área.	45
Figura 4.4.9: Caso de teste 3 – Rede filogenética.	46
Figura 4.4.10: Caso de teste 3 – Árvore filogenética obtida com o algoritmo de Prim.	47
Figura 4.4.11: Caso de teste 3 – Árvore filogenética obtida com o algoritmo de Dijkstra.	48
Figura 4.4.12: Caso de teste 3 – Solução proposta pelo especialista da área.	48
Figura 4.4.13: Caso de teste 4 – Rede filogenética.	49
Figura 4.4.14: Caso de teste 4 – Árvore filogenética obtida com o algoritmo de Prim.	50
Figura 4.4.15: Caso de teste 4 – Árvore filogenética obtida com o algoritmo de Dijkstra.	51
Figura 4.4.16: Caso de teste 4 – Solução proposta pelo especialista da área.	51
Figura 4.4.17: Caso de teste 5 – Rede filogenética.	52
Figura 4.4.18: Caso de teste 5 – Árvore filogenética obtida com o algoritmo de Prim.	53
Figura 4.4.19: Caso de teste 5 – Árvore filogenética obtida com o algoritmo de Dijkstra.	54
Figura 4.4.20: Caso de teste 5 – Solução proposta pelo especialista da área.	55
Figura 5.2.1: Caso de teste 1 – Árvore filogenética.	61
Figura 5.2.2: Caso de teste 1 – Parâmetros de entrada para f1.	62
Figura 5.2.3: Caso de teste 1 – Ficheiro Excel resultante para f1.	63
Figura 5.2.4: Caso de teste 1 – Árvore filogenética com fundadores para f1.	63
Figura 5.2.5: Caso de teste 1 – Parâmetros de entrada para f2.	64
Figura 5.2.6: Caso de teste 1 – Ficheiro Excel resultante para f2.	64
Figura 5.2.7: Caso de teste 1 – Árvore filogenética com fundadores para f2.	65
Figura 5.2.8: Caso de teste 2 – Árvore filogenética.	66
Figura 5.2.9: Caso de teste 2 – Parâmetros de entrada para f1.	67
Figura 5.2.10: Caso de teste 2 – Ficheiro Excel resultante para f1.	67
Figura 5.2.11: Caso de teste 2 – Árvore filogenética com fundadores para f1.	68
Figura 5.2.12: Caso de teste 2 – Parâmetros de entrada para f2.	68
Figura 5.2.13: Caso de teste 2 – Ficheiro Excel resultante para f2.	69
Figura 5.2.14: Caso de teste 2 – Árvore filogenética com fundadores para f2.	69
Figura 5.2.15: Caso de teste 3 – Árvore filogenética.	70
Figura 5.2.16: Caso de teste 3 – Parâmetros de entrada para f1.	71
Figura 5.2.17: Caso de teste 3 – Ficheiro Excel resultante para f1.	71

Lista de Figuras

Figura 5.2.18: Caso de teste 3 – Árvore filogenética com fundadores para f1.	72
Figura 5.2.19: Caso de teste 3 – Parâmetros de entrada para f2.	72
Figura 5.2.20: Caso de teste 3 – Ficheiro Excel resultante para f2.	73
Figura 5.2.21: Caso de teste 3 – Árvore filogenética com fundadores para f2.	73
Figura 5.2.22: Caso de teste 4 – Árvore filogenética.	74
Figura 5.2.23: Caso de teste 4 – Parâmetros de entrada para f1.	75
Figura 5.2.24: Caso de teste 4 – Ficheiro Excel resultante para f1.	75
Figura 5.2.25: Caso de teste 4 – Árvore filogenética com fundadores para f1.	75
Figura 5.2.26: Caso de teste 4 – Parâmetros de entrada para f2.	76
Figura 5.2.27: Caso de teste 4 – Ficheiro Excel resultante para f2.	76
Figura 5.2.28: Caso de teste 4 – Árvore filogenética com fundadores para f2.	77
Figura 5.2.29: Caso de teste 5 – Árvore filogenética.	77
Figura 5.2.30: Caso de teste 5 – Parâmetros de entrada para f1.	78
Figura 5.2.31: Caso de teste 5 – Ficheiro Excel resultante para f1.	79
Figura 5.2.32: Caso de teste 5 – Árvore filogenética com fundadores para f1.	79
Figura 5.2.33: Caso de teste 5 – Parâmetros de entrada para f2.	80
Figura 5.2.34: Caso de teste 5 – Ficheiro Excel resultante para f2.	80
Figura 5.2.35: Caso de teste 5 – Árvore filogenética com fundadores para f2.	81
Figura 6.1.1: Diagrama de casos de utilização da aplicação	84
Figura 6.2.1: Menu inicial da aplicação	85
Figura 6.2.2: Módulo de geração de árvores	85
Figura 6.2.3: Módulo de escolha de arestas	86
Figura 6.2.4: Módulo de estatísticas	87
Figura 7.1.1: Distribuição da probabilidade de migração para as Ilhas do Japão baseado na estimativa das idades de 214 fundadores utilizando rho.	90
Figura A.1: Utilização do Network 4.6 (1).	97
Figura A.2: Utilização do Network 4.6 (2).	98
Figura A.3: Utilização do Network 4.6 (3).	98
Figura A.4: Utilização do Network 4.6 (4).	98
Figura A.5: Utilização do Network 4.6 (5).	99
Figura A.6: Utilização do Network 4.6 (6).	99
Figura A.7: Utilização do Network 4.6 (7).	99
Figura A.8: Rede filogenética gerada.	100
Figura A.9: <i>Work flow</i> do Network 4.6.	101
Figura B.1: Menu inicial da aplicação.	113
Figura B.2: Selecção do ficheiro que contém a rede.	114
Figura B.3: Menu de parâmetros da ferramenta de geração de árvores.	114
Figura B.4: Escolha das arestas da árvore de forma manual.	115
Figura B.5: Selecção do ficheiro que contém a árvore.	116

Lista de Figuras

Figura B.6: Menu de parâmetros da ferramenta de cálculo de estatísticas.	116
--	-----

Lista de Tabelas

Tabela 4.4.1: Caso de teste 1 – Características da rede filogenética original.	40
Tabela 4.4.2: Caso de teste 2 – Características da rede filogenética original.	43
Tabela 4.4.3: Caso de teste 2 – Resumo dos resultados obtidos com os dois algoritmos.	46
Tabela 4.4.4: Caso de teste 3 – Características da rede filogenética original.	47
Tabela 4.4.5: Caso de teste 3 – Resumo dos resultados obtidos com os dois algoritmos.	49
Tabela 4.4.6: Caso de teste 4 – Características da rede filogenética original.	50
Tabela 4.4.7: Caso de teste 4 – Resumo dos resultados obtidos com os dois algoritmos.	52
Tabela 4.4.8: Caso de teste 5 – Características da rede filogenética original.	53
Tabela 4.4.9: Caso de teste 5 – Resumo dos resultados obtidos com os dois algoritmos.	55
Tabela 4.5.1: Características da rede filogenética de cada caso de tese.	56
Tabela 4.5.2: Resumo dos resultados gerais.	56
Tabela 5.2.1: Caso de teste 1 – Características da árvore filogenética.	62
Tabela 5.2.2: Caso de teste 2 – Características da árvore filogenética.	66
Tabela 5.2.3: Caso de teste 3 – Características da árvore filogenética.	70
Tabela 5.2.4: Caso de teste 4 – Características da árvore filogenética.	74
Tabela 5.2.5: Caso de teste 5 – Características da árvore filogenética.	78
Tabela 7.1.1: Cumprimento dos objectivos do projecto.	89

Abreviaturas e Símbolos

CR	Região de Controlo do DNA mitocondrial (<i>Control Region</i>)
CRS	Sequência Referência de Cambridge (<i>Cambridge Reference Sequence</i>)
DNA	Ácido DesoxirriboNucleico (<i>DeoxyriboNucleic Acid</i>)
D-loop	Região de Controlo do DNA mitocondrial
HVS-I	Região Hipervariável 1 do DNA mitocondrial
HVS-II	Região Hipervariável 2 do DNA mitocondrial
Java SE 6	Java Standard Edition – Versão 6
mtDNA	DNA Mitocondrial
RNA	Ácido RiboNucleico (<i>RiboNucleic Acid</i>)
rRNA	RNA Ribossómico

Capítulo 1

Introdução

O tema desta dissertação enquadra-se no trabalho do grupo de Diversidade Genética do Instituto de Patologia e Imunologia Molecular da Universidade do Porto (IPATIMUP). No âmbito das actividades realizadas por este grupo, é utilizado um método denominado Análise de Fundador, que consiste em calcular o tempo de migração das populações humanas tanto na Pré-História como na História, através da determinação das relações entre a informação genética de indivíduos de diferentes áreas geográficas e o cálculo do tempo de separação das populações, realizado a partir da acumulação de diferenças genéticas (mutações) durante um período de tempo.

Assim, foi proposto pelo grupo de Diversidade Genética do IPATIMUP um projecto que consiste no desenvolvimento de ferramentas informáticas que apoiem os métodos de análise de genética populacional necessários para realizar a Análise de Fundador. O principal objectivo deste trabalho é por isso a implementação computacional e automatização parcial deste método, no mínimo parcial, para que no futuro seja possível efectuar a análise num período de tempo bastante mais curto e de forma fiável.

O IPATIMUP é uma associação privada, sem fins lucrativos e de Utilidade Pública, criada em 1989 sob a égide da Universidade do Porto. Tem uma articulação funcional com o IBMC¹ e o INEB² e protocolos de colaboração com numerosas instituições europeias, do Brasil, E.U.A., Canadá, China e PALOP's³. Estas colaborações justificam, por exemplo, que cerca de 60% dos artigos científicos publicados anualmente por investigadores do IPATIMUP tenham co-autores de instituições estrangeiras [CiViv11]. A vocação fundamental do IPATIMUP consiste na investigação de translação e pós-graduação em Oncobiologia, Medicina Molecular e Genética. Constituem também actividades fundamentais a divulgação científica com ênfase na prevenção

¹ IBMC – Instituto de Biologia Molecular e Celular;

² INEB – Instituto de Engenharia Biomédica;

³ PALOP – Países Africanos de Língua Oficial Portuguesa.

do cancro e de outras doenças, e a prestação de serviços à comunidade nos seguintes domínios: diagnóstico anatomopatológico e selecção terapêutica de doenças cancerosas e pré-cancerosas, diagnóstico molecular, aconselhamento genético de cancro familiar e ambiental, genética populacional e forense [CiViv11].

1.1 Projecto

O método da Análise de Fundador necessita de um laborioso processo de colecta, armazenamento e análise de dados, o que faz com que não tenha sido muito aplicado desde o ano 2000, quando foi publicado [RMH00]. Actualmente, o processo utiliza diferentes aplicações de *software*, sendo necessário copiar dados entre aplicações para poder calcular as estatísticas e todos os dados intermédios necessários. Adicionalmente existe ainda uma grande quantidade de dados a ser gerida, o que faz com que os resultados sejam bastante susceptíveis a erros, bem como morosos de obter. Qualquer alteração que se pretenda efectuar numa árvore filogenética, como por exemplo, inserção de novas sequências, implica que sejam novamente calculadas todas as estatísticas manualmente.

Os objectivos gerais deste projecto consistem na automatização do processo de construção de árvores filogenéticas e da Análise de Fundador.

A proposta de trabalho para alcançar os objectivos subdivide-se em três fases distintas:

- **1ª Fase** – Automatizar tanto quanto possível a construção de árvores a partir de grafos, produzidos pelo *software* Network.
 - o Construir diferentes árvores filogenéticas alternativas
 - o Desenvolver um algoritmo de *ranking* das árvores
- **2ª Fase** – Automatizar o processo de cálculo de estatísticas para o método da Análise de Fundador
 - o Definição e programação de estatísticas relevantes
 - o Melhorias na Análise de Fundador
- **3ª Fase** – Serão estudadas melhorias do método, juntamente com um dos seus criadores, o Dr. Vincent Macaulay da Universidade de Glasgow, ou o Dr. Martin Richards da Universidade de Leeds.

1.2 Motivação

A principal motivação para realizar este projecto centra-se na importância que este método tem para estudo da diversidade genética, sendo possível com este projecto otimizar a forma como o método é aplicado, através da redução do tempo despendido para o executar, e torná-lo mais útil para os profissionais da área, através da adição de novas estatísticas.

Sendo automatizada a aplicação da Análise de Fundador, será possível realizar mais estudos nesta área e de forma mais fiável.

Será poupado tempo ao realizar a Análise de Fundador, pois pretende-se otimizar o tempo despendido na construção de árvores filogenéticas e tornar essa construção menos susceptível a erros. Será também poupado tempo no cálculo de estatísticas, que demora imenso tempo no processo manual, uma vez que é preciso efectuar vários cálculos e copiar dados entre aplicações, estando todo este processo bastante susceptível a erros.

A título de exemplo, o grupo de Diversidade Genética elaborou recentemente um estudo que envolveu um investigador sénior durante cerca de 2 anos. Os dados utilizados neste estudo estão a ser usados para testar o *software* desenvolvido. Nesta fase final do trabalho, 1/3 dos dados usados demoraram cerca de 1 semana a serem analisados, quando esse mesmo volume de dados demorava anteriormente cerca de 1 ano a ser analisado.

1.3 Estrutura da Dissertação

Para além da introdução, esta dissertação contém mais 5 capítulos. No Capítulo 2, são apresentados os conceitos da área da Genética Populacional, tais como a definição de DNA e DNA mitocondrial, de Emparelhamento de Sequências, Análise Filogenética e Análise de Fundador, conceitos necessários para se poder perceber de forma clara o tema desta dissertação. No Capítulo 3, é descrito o estado da arte e são apresentadas aplicações de *software* já existentes que satisfazem alguns dos requisitos da área. No Capítulo 4, é descrito todo o processo de implementação da parte do projecto correspondente à construção de árvores filogenéticas a partir de redes filogenéticas previamente existentes. No Capítulo 5 é descrito o processo de desenvolvimento da parte do projecto relativa ao cálculo de estatísticas e melhorias da Análise de Fundador. No Capítulo 6 será apresentada a aplicação desenvolvida.

Por último, no Capítulo 7, apresentar-se-ão as conclusões do trabalho que foi realizado e serão descritas as perspectivas de trabalho futuro.

Introdução

Capítulo 2

Conceitos Teóricos da Área

2.1 DNA

O DNA, sigla internacional para **Ácido DesoxirriboNucleico** (ADN em português), é a molécula da vida, que contém informação relativa à estrutura, desenvolvimento e manutenção da vida dos seres vivos. Molecularmente, o DNA consiste num polímero formado por elementos designados por nucleótidos, que são constituídos por um açúcar (pentose) do tipo desoxirribose, um grupo fosfato e uma base azotada. As bases azotadas são: Adenina (A), Timina (T), Guanina (G) e Citosina (C). Em termos de estrutura química, a Timina e a Citosina consistem em anéis azotados simples, designados por pirimidinas, e a Adenina e a Guanina em anéis azotados duplos, designados por purinas. Estas quatro letras formam um alfabeto, que permite codificar a informação para desenvolver as características de todos os seres vivos. A referência completa de letras que codifica todas as características de um ser vivo é conhecida como genoma. No caso dos humanos, o genoma está organizado em 23 pares de cromossomas localizados no núcleo. As partes de texto que codificam informação genética são conhecidas por genes e a sequência das suas bases codifica a informação genética para uma determinada característica do ser vivo, sendo esta informação transcrita para RNA (**Ácido RiboNucleico**) e traduzida para proteínas. Mas grande parte do texto (cerca de 97%) tem significado biológico ainda desconhecido, possuindo uma elevada proporção de sequências repetidas várias vezes seguidas.

A molécula de DNA é constituída por duas cadeias de nucleótidos organizadas em dupla hélice, unidas por pontes de hidrogénio. Assim, as quatro bases azotadas encontram-se sempre emparelhadas de forma complementar, ou seja, a Adenina liga-se sempre à Timina através de duas pontes de hidrogénio e a Citosina liga-se sempre à Guanina através de três pontes de hidrogénio [Pit01], como se pode observar na Figura 2.1.1.

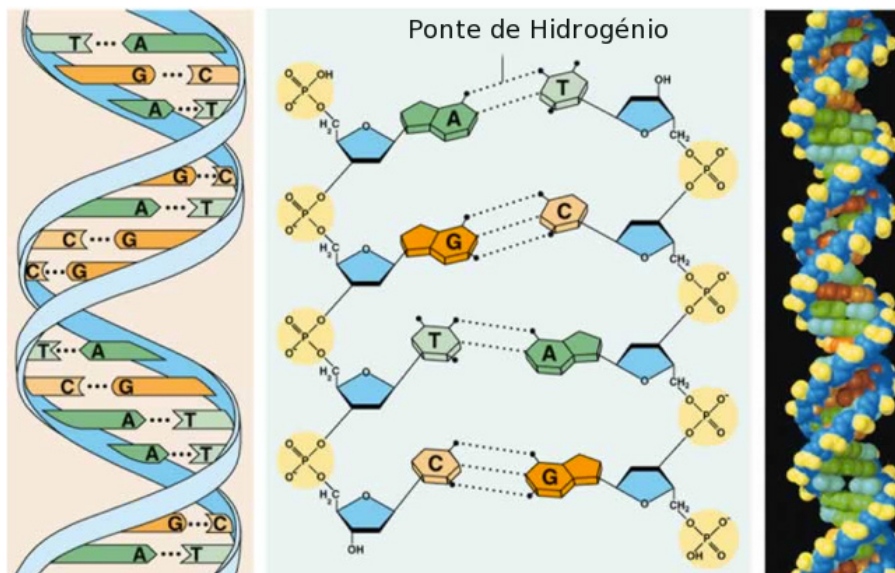


Figura 2.1.1: Estrutura em dupla hélice da molécula de DNA com ligações entre bases azotadas⁴

No DNA ocorrem por vezes mudanças na sua sequência de letras, gerando polimorfismos. Quando estes polimorfismos acontecem numa região codificante e levam à manifestação de uma doença, recebem a designação de mutações, mas há uma grande flexibilidade na utilização destes termos. Os polimorfismos podem ser espontâneos ou induzidos, por exemplo, pelo efeito de radiações ou de agentes químicos tóxicos, e podem ser classificados como:

- **Inserção** (ganho de bases)
- **Deleção** (perda de bases)
- **Substituição de bases**
 - o **Transição** – substituição de uma base purina por outra purina, ou de uma pirimidina por outra pirimidina
 - o **Transversão** – substituição de uma base pirimidina por uma purina ou vice-versa

⁴ Pitocchelli, J. Saint Anselm College. 2001. Lecture Notes for Genetics. Disponível em: <http://www.anselm.edu/homepage/jpitolch/genbio/doublehelix.JPG> Última consulta em 10 de Janeiro de 2011.

Dado que é mais difícil substituir bases de um tipo por outro, em termos estruturais, as transições são mais frequentes que as transversões.

As mutações podem ocorrer a nível somático, só numa determinada célula somática do indivíduo (todas as células que não as sexuais ou gâmetas), estando restritas a essa célula e às células que resultam da sua divisão – por exemplo, o cancro resulta de mutações que ocorreram em certos genes numa dada célula somática. Se as mutações ocorrem nos gâmetas, estas podem ser transmitidas à próxima geração, originando diversidade genética na população e espécie. Neste trabalho só vamos analisar as mutações que são transmitidas ao longo das gerações.

Como a mutação é um acontecimento regular, a taxa de mutação é directamente proporcional ao tamanho do genoma e ao tempo geracional da espécie. Foi já possível avaliar as taxas de mutação para os genomas de diversas espécies e, por exemplo, na espécie humana, *Homo sapiens*, a taxa de mutação pontual no DNA nuclear é de sensivelmente 10^{-9} por base por ano.

2.2 DNA Mitochondrial

As células eucarióticas apresentam uma estrutura organizada em compartimentos rodeados por membrana. Para além do núcleo contendo o genoma, existem vários compartimentos citoplasmáticos denominados organelos, que desempenham diversas funções biológicas. Um organelo essencial é a mitocôndria, que é responsável pela produção de energia celular a partir de glicose e oxigénio, através da respiração oxidativa, de forma a manter o funcionamento da célula. Na Figura 2.2.1 é apresentada uma ilustração da estrutura da mitocôndria.

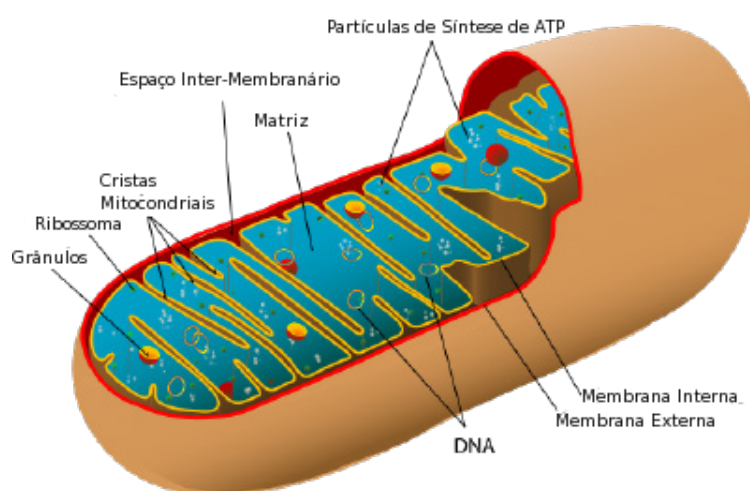


Figura 2.2.1: Ilustração representativa da mitocôndria. ⁵

⁵ Mariana Ruiz Villarreal. Mitochondrion. Wikimedia Commons. Disponível em: <http://biology.about.com/od/cellanatomy/ss/mitochondria.htm> Última consulta em 10 Janeiro de 2011.

Uma característica especial que distingue as mitocôndrias dos outros organelos presentes em células eucarióticas animais é a presença de um genoma próprio, o DNA mitocondrial (abreviadamente mtDNA), e que tem várias cópias por mitocôndria. O mtDNA humano está organizado também em cadeia dupla mas, ao contrário do nuclear, é circular e não linear, veja-se na Figura 2.2.2. Possui cerca de 16569 pares de bases, organizados em duas regiões: uma região codificante composta por genes que codificam 13 proteínas da cadeia respiratória, 2 RNAs ribossômicos (rRNA) e 22 RNAs de transferência (tRNA) [And81, AKC99]; uma região de controlo ou D-loop com 1122 pares de bases, onde se localizam os sinais de controlo para a replicação do DNA e da transcrição para RNA e duas regiões que apresentam uma elevada diversidade genética, recebendo o nome de regiões hipervariáveis I e II (HVS-I e HVS-II).

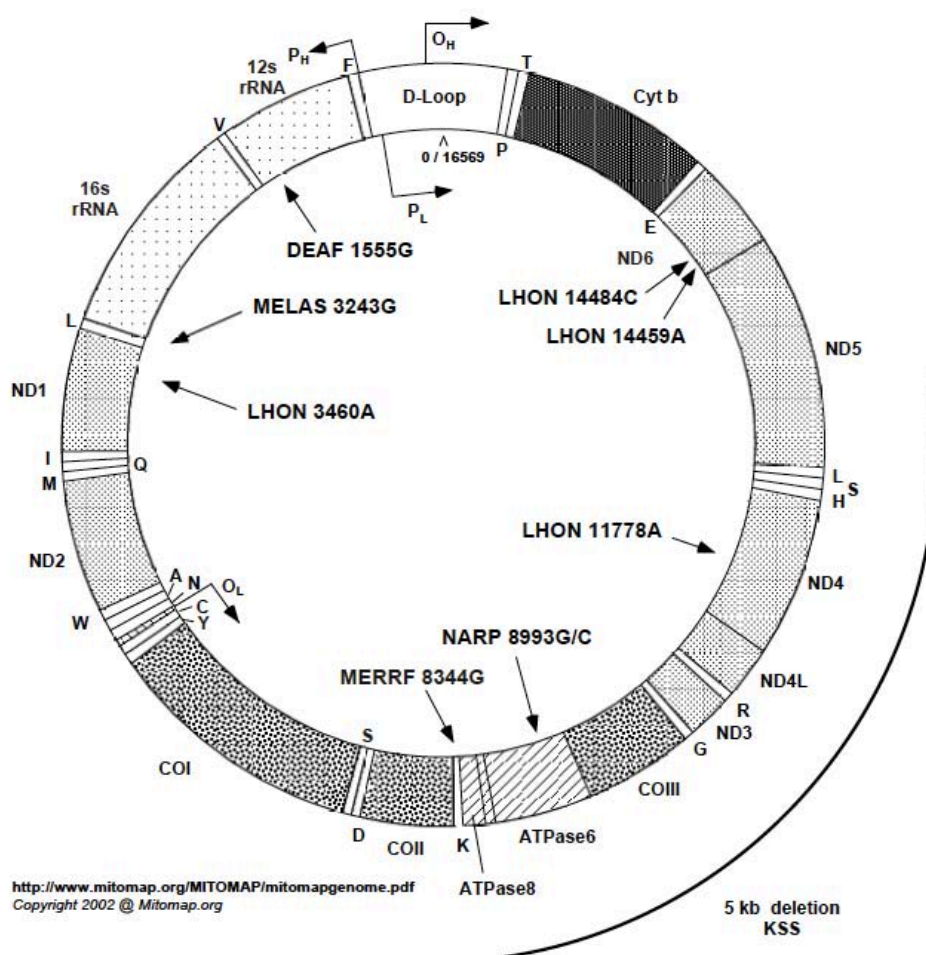


Figura 2.2.2: Esquema representativo da molécula de mtDNA humano, com as localizações dos vários genes na região codificante e a região de controlo (*D-loop*).⁶

⁶ Mitochondrial DNA Map. 2002. Mitomap. Disponível em: <http://www.mitomap.org/MITOMAP/mitomapgenome.pdf> Última consulta em 10 Janeiro de 2011.

A teoria endossimbiótica defende que as mitocôndrias descendem de bactérias que invadiram outras células, originando as células eucarióticas. Esta teoria baseia-se nos factos de que a estrutura do mtDNA é circular como os genomas bacterianos e não linear como no DNA nuclear das células eucarióticas, e de que o código genético para sintetizar as proteínas mitocondriais apresenta também algumas diferenças relativamente ao código genético nuclear dos eucariontes.

O mtDNA é transmitido entre gerações somente pela mãe [GBC80], ao contrário do DNA nuclear que tem contribuição de ambos os progenitores. Após a fertilização do óvulo pelo espermatozóide, as mitocôndrias deste são destruídas, eliminando o mtDNA paterno no novo indivíduo. Assim, no mtDNA não ocorre o fenómeno de recombinação, o qual efectua a troca de material genético entre os cromossomas nucleares paterno e materno. Assim sendo, quando se estuda o mtDNA caracterizam-se linhagens maternas.

A taxa de mutação do mtDNA é superior ao do DNA nuclear, aproximadamente 10 vezes maior na região codificante (10^{-8}) e 100 vezes maior na região controlo (10^{-7}) [IKP00]. As razões para a taxa de mutação ser maior no mtDNA são:

- O mtDNA não está envolvido por proteínas, como o DNA nuclear
- O mtDNA não tem mecanismos de reparação de mutações tão eficientes como o DNA nuclear
- Aquando da produção de energia, na mitocôndria são produzidos radicais livres, que podem danificar as moléculas de mtDNA.

2.3 Alinhamento de Sequências

A identificação de todas as posições nucleotídicas variáveis no mtDNA num conjunto de indivíduos é feita por comparação a uma referência, que na espécie humana corresponde à primeira sequência publicada e designada por CRS (*Cambridge Reference Sequence*), que foi revista e corrigida em 1999 [AKC99]. Para tal, tem que se efectuar o alinhamento das várias sequências em relação à CRS.

O alinhamento de sequências consiste no processo de comparar duas ou mais sequências, procurando séries de caracteres ou um padrão de caracteres que tenha a mesma ordem nas 2 sequências. O alinhamento é realizado antes da fase da construção das árvores filogenéticas, tema abordado neste projecto. A razão principal para utilizar o alinhamento de sequências é a possibilidade de descobrir informação funcional e evolutiva nas sequências biológicas. No caso de serem detectadas sequências similares, estas podem possuir a mesma função e caso duas sequências de diferentes organismos sejam similares, poderá ter existido uma sequência ancestral comum, sendo nesse caso as sequências definidas como homólogas [Sch11]. No

âmbito da evolução, as sequências de DNA sofrem mutações, que podem ser inserções, deleções ou substituições, tal como referido anteriormente na Secção 2.1.

2.4 Análise Filogenética

Ao longo do tempo, o mtDNA acumulou diversas variações genéticas, sendo parte delas passadas às gerações seguintes. Cada indivíduo é caracterizado por um conjunto de modificações genéticas, que são os polimorfismos. Este conjunto é conhecido como o haplótipo mitocondrial. O conjunto dos haplótipos dos indivíduos que partilham o mesmo ascendente materno constitui um *clade* ou grupo monofilético, conhecido também como haplogrupo [LW00, HEF02]. Através da análise filogenética é possível obter o número e a sequência temporal de todas as mutações ocorridas numa população ou espécie [BFS95].

Numa árvore filogenética, em cada ramo, estão representados todos os caracteres ou, no caso de se estudar DNA, as posições nucleotídicas distintas entre dois nodos, um antepassado e o seu descendente. Um carácter pode ser considerado homoplásico ou recorrente quando evoluiu independentemente mais do que uma vez, não tendo uma origem única. No caso do mtDNA, esta característica é muito importante, uma vez que a elevada taxa de mutação do mtDNA faz com que existam muitas posições recorrentes, incutindo alguma incerteza na reconstrução da filogenia. Por este motivo, tornou-se vulgar a aplicação de redes filogenéticas (*networks*) ao estudo do mtDNA em vez de árvores filogenéticas.

Na Figura 2.4.1 são apresentadas as diferenças entre estas duas representações filogenéticas possíveis.

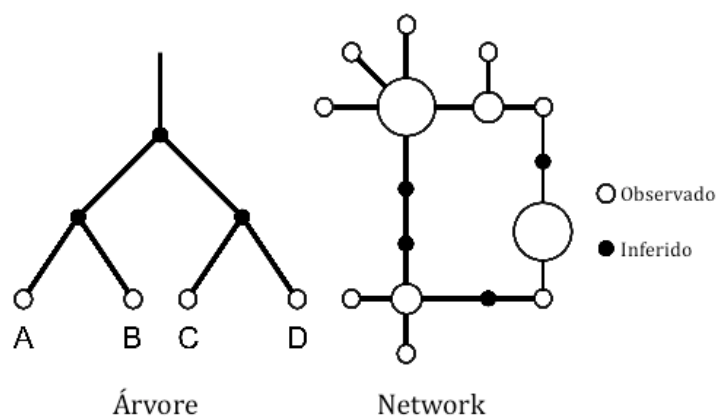


Figura 2.4.1: Possíveis representações filogenéticas. ⁷

⁷ Kidd, D. EvoViz. 2007. Networks. Disponível em: <https://www.nescent.org/sites/evoviz/Network> Última consulta em 16 de Janeiro de 2011.

Conceitos Teóricos da Área

As principais diferenças são:

- **Árvore Filogenética**
 - o Representa apenas um caminho evolutivo possível
 - o É uma rede filogenética sem reticulações (ciclos)
 - o É a forma mais adequada para modelar sistemas sem reticulações, ou seja, sem ciclos
- **Rede Filogenética (*network*)**
 - o Uma rede filogenética não possui ligações entre todas as entidades e possui pelo menos uma reticulação
 - o É a superclasse das representações, contendo todas as árvores possíveis

Nesta área é utilizado um *software*, criado pela Fluxus Technology Ltd., denominado Network (versão actual 4.6)⁸ que tem o objectivo de gerar redes filogenéticas a partir de dados genéticos. Estes dados de entrada terão de já ter sido submetidos a um *software* de alinhamento de sequências, antes de serem lidos pelo Network. Possui ainda a capacidade de fornecer estimativas de idade para qualquer ancestral presente na rede filogenética, desde que o utilizador identifique o ancestral e todos os seus descendentes. Para a geração das redes filogenéticas, este *software* utiliza dois algoritmos apresentados na Secção 3.4. O resultado obtido a partir deste programa (*output*) é necessário para este projecto, na medida que este é a base para realizar a Análise de Fundador.

Assim, sequenciando o mtDNA de vários indivíduos, é possível construir uma árvore filogenética representativa da divergência das sequências de mtDNA, registando a sua história evolutiva. Zuckerkandl e Pauling, em 1965, sugeriram a hipótese da existência do relógio molecular da evolução [ZP65], sendo esta hipótese revista em 1986 [Aya86]. Na sua revisão concluiu-se que a evolução é suficientemente regular para ser aplicada na maior parte das situações como um relógio, sendo apenas obrigatório ter atenção à incerteza relativa às propriedades do relógio molecular. Aquando do desenvolvimento destes conceitos, não existiam ainda muitas sequências de DNA descritas, por forma a poder testar a sua veracidade. Na revisão de 1986 [Aya86] foi utilizada como primeira molécula para validação, a proteína mitocondrial *citocromo c*, que demonstrou comportar-se de forma muito semelhante a um relógio molecular [MGF99].

⁸ *Software* disponível em: <http://www.fluxus-engineering.com/sharenet.htm>

Como existe maior número de mutações no mtDNA e como não há recombinação no seu genoma, este é o genoma mais utilizado em análises filogenéticas. Quanto maior o número de mutações, maior é a diversidade entre os indivíduos e associado à ausência de recombinação, mais fácil será a reconstrução da história evolutiva da molécula de mtDNA.

2.5 Análise de Fundador

A Análise de Fundador é um método de análise em Genética Populacional que visa calcular o tempo de migração das populações humanas ao longo da história, com base na análise filogenética da sua diversidade. Utiliza uma hipotética população de origem e uma hipotética população de destino. Será necessário identificar todos os potenciais fundadores e estimar o tempo desde a chegada ao destino. São requisitos fundamentais a identificação das sequências de fundador na população de origem, bem como a determinação dos *clades* derivados destes fundadores na população de destino. Estas idades estimadas para cada *clade* na população de destino, são comparadas com tempos de migração inferidos por outros campos de pesquisa como a arqueologia, ecologia, climatologia, geologia, linguística, entre outros. A idade de um *clade* residente numa região poderá determinar a idade de colonização desta mesma região. É necessário existir uma representação filogenética das sequências de DNA das populações, recorrendo-se, para tal, a redes filogenéticas, como a representada na Figura 2.5.1.

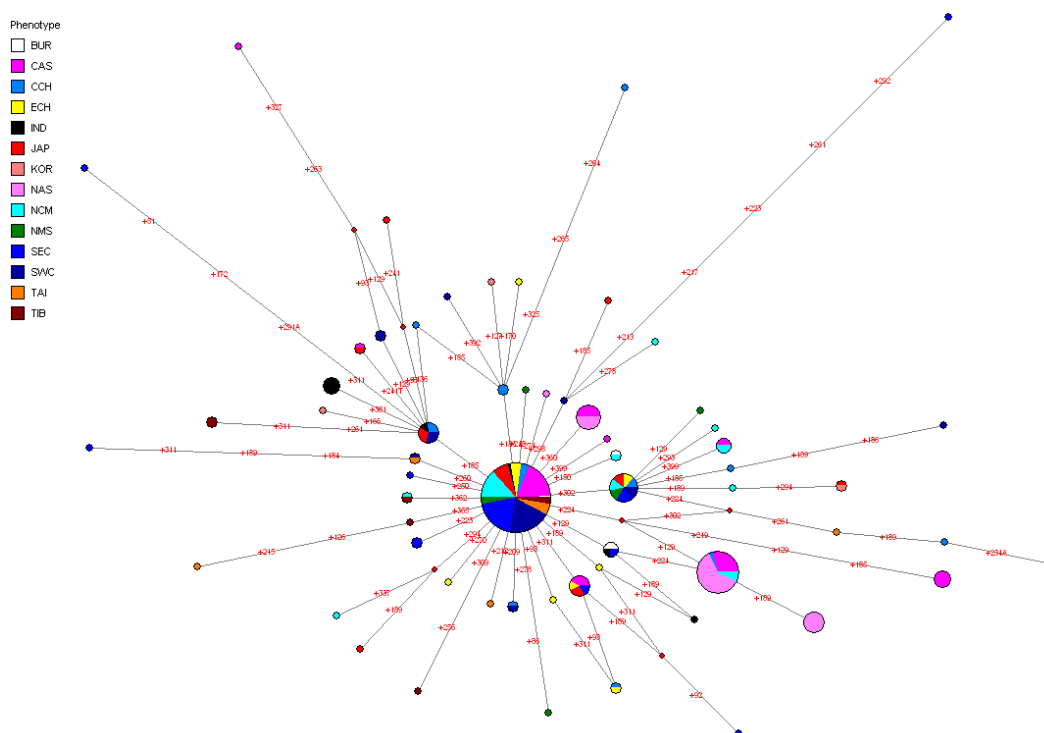


Figura 2.5.1: Representação de uma rede filogenética com o *software* Network 4.6.

O primeiro passo importante é a definição das populações de origem e destino, de modo a saber quais os conjuntos que irão ser incluídos na origem, quais os que irão ser atribuídos à população de destino e quais os conjuntos que não irão desempenhar qualquer papel fundamental na análise.

A seguir, para que seja possível detectar os fundadores será necessária a reconstrução filogenética, realizada através do algoritmo *reduced-median network*, que está implementado no *software* Network 4.6, utilizado actualmente. O algoritmo *median-joining network* não poderá ser utilizado para este caso, uma vez que produz redes filogenéticas menos complexas [Soa09].

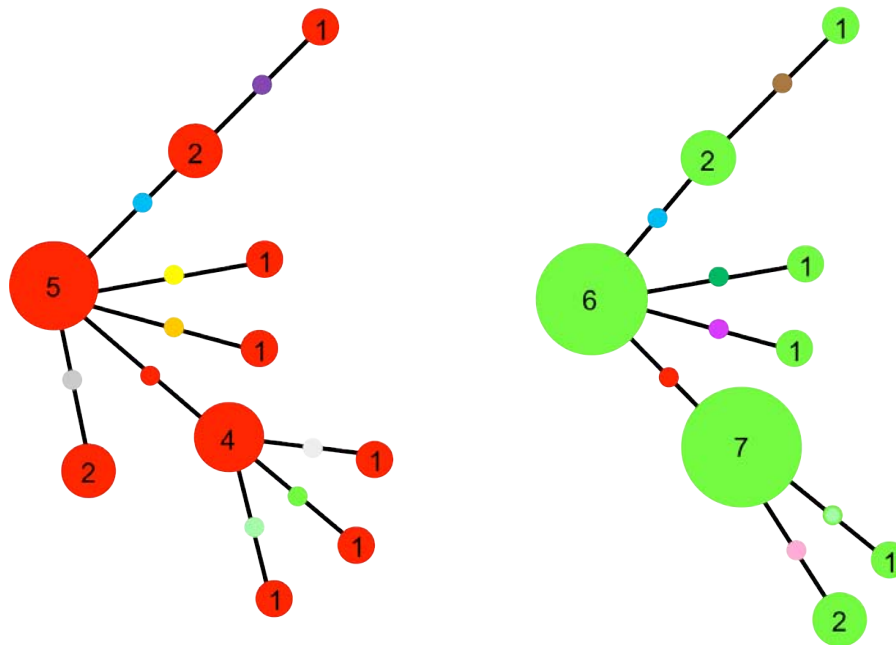


Figura 2.5.2: Diversidade das populações de origem e de destino.

Os potenciais fundadores são aqueles cujos tipos de sequências são idênticos na população de destino e de origem.

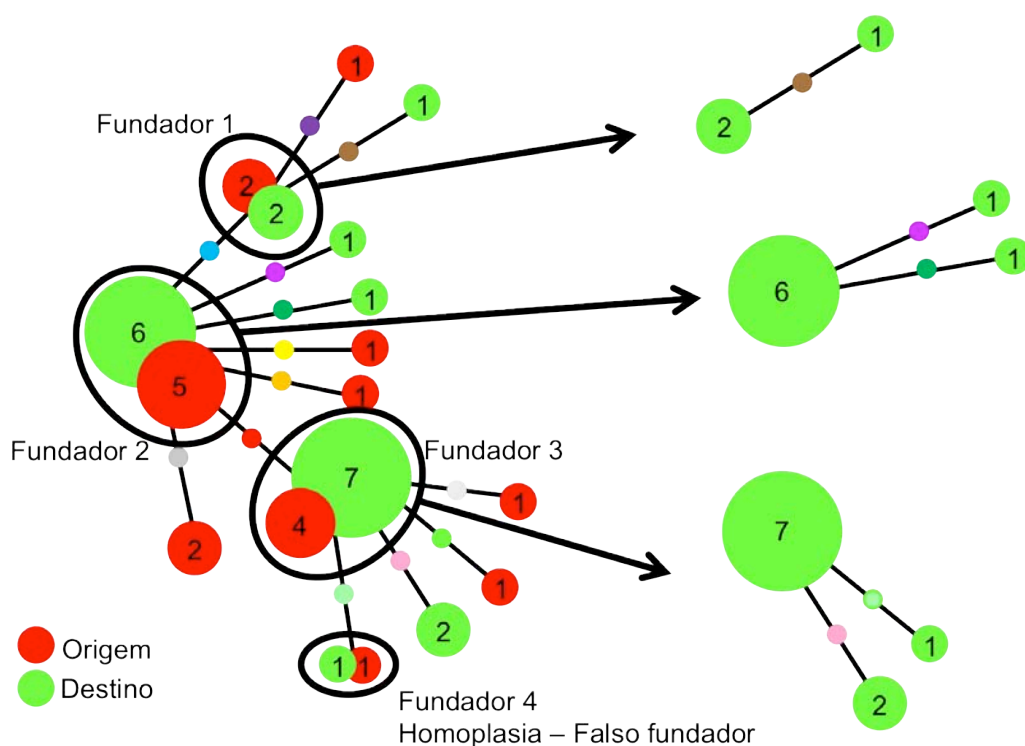


Figura 2.5.3: Potenciais fundadores obtidos através do cruzamento das populações de origem e destino.

Assim, na Figura 2.5.2 são representadas as árvores filogenéticas da diversidade da população de origem e da população de destino, que demonstram a diversidade genética de cada uma. Através da sobreposição das duas árvores, conseguem-se obter os tipos de sequências idênticas, que consistem nos potenciais fundadores, tal como é possível observar na Figura 2.5.3. Contudo, dos 4 hipotéticos fundadores alguns poderão ser falsos fundadores. Na descrição original da Análise de Fundador os autores determinaram que hipotéticos fundadores, onde não existe diversidade nesse *clade*, na população de origem, têm uma probabilidade alta de serem falsos fundadores gerados por homoplasia ou migração recente no sentido inverso. No caso do exemplo acima só existem 3 fundadores, uma vez que o potencial fundador 4 é um “falso fundador”, possivelmente gerado por homoplasia, ou seja, houve uma evolução idêntica mais do que uma vez. Daqui se conclui que na Análise de Fundador o fenómeno da homoplasia pode gerar algum ruído de fundo na interpretação dos fundadores.

2.6 Resumo

Com este capítulo pretendeu-se apresentar ao leitor os conceitos-base da área da biologia necessários para ser possível compreender o contexto deste projecto. Todos estes conceitos, apresentados cronologicamente, constituem tudo aquilo que ocorre antes da Análise de

Conceitos Teóricos da Área

Fundador, ou seja, será necessário obter as sequências de mtDNA, sendo feita inicialmente uma diferenciação entre DNA e mtDNA, e a partir dessas sequências, que têm de ser alinhadas por um software computacional para esse efeito, é possível construir redes filogenéticas. A partir deste ponto começa a Análise de Fundador, explicada resumidamente na Secção [2.5](#). No próximo capítulo serão apresentados estes temas com maior detalhe e exposto o estado da arte de cada um destes.

Conceitos Teóricos da Área

Capítulo 3

Revisão Bibliográfica

3.1 Emparelhamento de Sequências

Na Secção 2.3 introduziu-se o conceito de emparelhamento de sequências. Para clarificar a forma como este procedimento é feito, apresenta-se agora o estado da arte acerca deste tema.

Os elementos a ter em conta num alinhamento de sequências são as similaridades (*matches*) e diferenças (*mismatches*) entre os caracteres das sequências e os espaçamentos (*gaps*) que podem ser provocados por deleções ou inserções, tal como se pode observar na Figura 3.1.1.

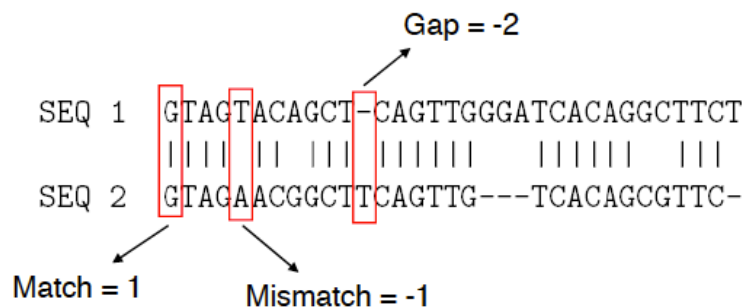


Figura 3.1.1: Comparação de duas sequências. ⁹

Com estes dados é possível calcular uma pontuação (*score*), que é dada pela equação (3.1.1), a seguir apresentada [Car08].

⁹ Carazzolle, M. 2008. Métodos de Alinhamento de Sequências Biológicas. Disponível em <http://www.lge.ibi.unicamp.br/lgeextensao2008/extsup/alinhamentos.pdf> Última consulta em 20 de Janeiro de 2011.

$$Score = \#Matches - \#Mismatches - 2 \times \#Gaps \quad (3.1.1)$$

Existem três tipos de alinhamento [Car08]:

- Alinhamento Global (Figura 3.1.2)
 - o Este tipo de alinhamento é efectuado sobre todo o comprimento da sequência, de modo a incluir o máximo de nucleótidos possível
 - o Aplica-se em sequências que sejam similares e com igual comprimento
 - o O alinhamento maximiza as regiões de semelhança e minimiza os espaçamentos
 - o Aplicações de *software* que o utilizam: CLUSTAL
- Alinhamento Local (Figura 3.1.3)
 - o É efectuado sobre partes da sequência que possuam a maior densidade de semelhanças
 - o Aplica-se em sequências similares em parte do seu comprimento, assim como em sequências diferentes em comprimento e sequências que partilhem uma região conservada
 - o É útil no caso de não existir conhecimento acerca das semelhanças entre as duas sequências
 - o Aplicações de *software* que o utilizam: BLAST, EMBOSS e FASTA
- Alinhamento Semi-global
 - o É utilizado para encontrar sobreposições de fragmentos de sequência
 - o Aplicações de *software* que o utilizam: PHRAP e CAP3

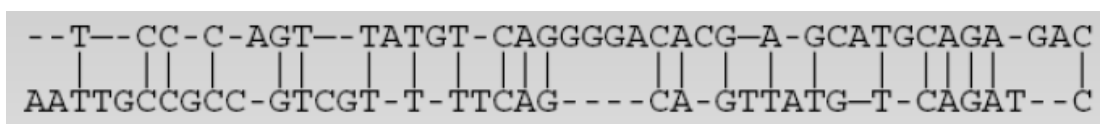


Figura 3.1.2: Alinhamento Global. ¹⁰



Figura 3.1.3: Alinhamento Local. ¹⁰

Dentro do alinhamento de sequências existe o alinhamento múltiplo de sequências, onde é efectuado o alinhamento de três ou mais sequências. Neste alinhamento pode ser identificada uma grande quantidade de nucleótidos que definem a estrutura funcional de famílias de indivíduos. No entanto é um problema difícil de resolver algorítmicamente.

Para além do alinhamento múltiplo, existe o alinhamento de pares de sequências, também conhecido como emparelhamento de sequências. Para este tipo de alinhamento existem diferentes algoritmos de emparelhamento, que se apresentam de seguida.

3.1.1 Análise da Matriz de Pontos

Este método consiste em representar toda e qualquer possibilidade de alinhamento de sequências como diagonais numa matriz. Com este método são comparadas duas sequências para avaliar um possível alinhamento dos caracteres da sequência. Desta forma é possível revelar inserções, deleções e repetições em sequências de DNA.

Começa por comparar pares de sequências sendo que a primeira sequência é listada sobre a segunda sequência. São utilizados pontos para representar caracteres similares, demonstrando as possíveis combinações dos caracteres da primeira sequência com os caracteres da segunda sequência. Quando se verifica uma diagonal formada por pontos, esta representa uma região de sequências similares, enquanto pontos aleatoriamente isolados representam combinações aleatórias. Na Figura 3.1.4 está representado um exemplo de uma matriz de pontos.

¹⁰ Schimek, M. Referate Bioinformatik I: Alignment of Pairs of Sequences. Disponível em: http://user.medunigraz.at/michael.schimek/Alignment_of_pairs_of_sequences.pdf Última consulta em 20 de Janeiro de 2011.

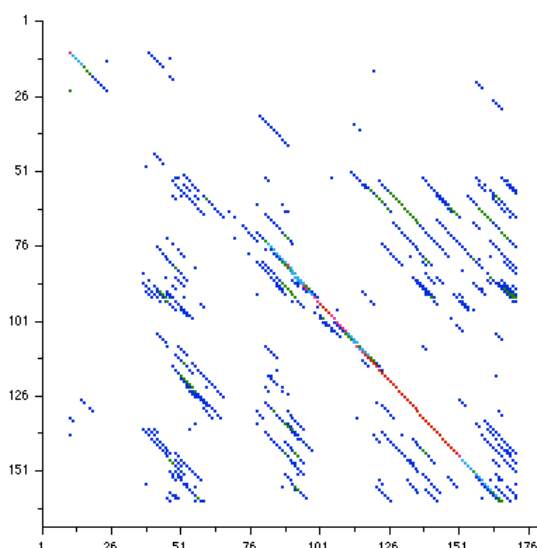


Figura 3.1.4: Matriz de pontos com várias diagonais e alguns pontos dispersos.¹¹

Assim, com a matriz de pontos é possível encontrar repetições dentro das sequências, através da análise da sequência com ela própria. Caso haja linhas diagonais de pontos, esta possui repetições. Poderão ser detectadas também repetições das mesmas sequências de caracteres diversas vezes, nesse caso estas são detectadas pela presença de linhas verticais ou horizontais na matriz resultante do cruzamento da sequência com ela própria [Sch11].

3.1.2 Algoritmos de Programação Dinâmica

Consiste num método computacional para alinhamento de duas sequências. Com a sua aplicação é possível obter a solução óptima de alinhamento, através da comparação de todos os pares de caracteres existentes nas duas sequências. São incluídos quer os caracteres similares como os que são diferentes, obtendo-se desta forma o maior número possível de combinações. Quanto menos similares forem as sequências, maior a dificuldade do alinhamento através deste algoritmo.

Para o alinhamento global é utilizado o Algoritmo de Needleman-Wunsch, no caso do alinhamento local, é utilizado o Algoritmo de Smith-Waterman. A programação dinâmica inicia o processamento no final das duas sequências e tenta combinar todos os pares de caracteres possíveis entre as sequências. Utilizando o sistema de pontuação baseado nas similaridades, diferenças e espaçamentos, anteriormente referido obtêm-se pontuações para cada diferente

¹¹ Schimek, M. Referate Bioinformatik I: Alignment of Pairs of Sequences. Disponível em: http://user.medunigraz.at/michael.schimek/Alignment_of_pairs_of_sequences.pdf Última consulta em 20 de Janeiro de 2011.

solução possível de alinhamento, sendo que a solução óptima será aquela que possua maior pontuação.

O grande defeito dos algoritmos baseados em programação dinâmica é que a sua execução pode ser muito lenta. Supondo que o alinhamento de 2 sequências de 100 pares de bases demora 1 segundo a ser executado, então o alinhamento de 10 milhões de pares de bases levará 377 anos [Car08, Sch11].

3.1.3 Dicionário de Palavras ou K-Tuplos

Este método sobrepõe-se aos algoritmos de programação dinâmica uma vez que é muito mais rápido, apesar de não garantir que é encontrada a solução óptima. Resumidamente, o que este método faz é identificar palavras curtas dentro de uma das sequências, realizando depois uma pesquisa por essas mesmas palavras nas sequências restantes.

Este método foi obtido através de algumas modificações realizadas aos métodos referidos anteriormente, tendo-se criado assim os dois programas de alinhamento local BLAST e FASTA [Car08, Sch11].

3.2 Grafos

Um grafo consiste num conjunto de nós ou vértices ligados por segmentos de recta ou arestas. É definido pela equação (3.2.1)

$$G = (V, A), A \subseteq V \times V \quad (3.2.1)$$

V representa o conjunto de vértices ou nós do grafo e A o conjunto de arestas.

As arestas de um grafo podem ser simples segmentos de recta ou setas, diferenciando-se assim os grafos não-dirigidos dos dirigidos. Na Figura 3.2.1 e Figura 3.2.2 podem ser observadas as diferenças entre estes dois tipos de grafos [Ant05]. Dois vértices ou nós de um grafo são considerados adjacentes no caso de estes estarem ligados por uma aresta.

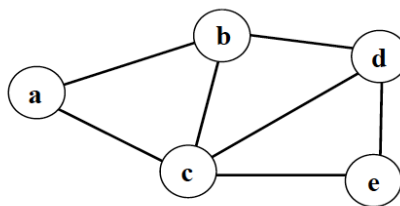


Figura 3.2.1: Grafo não-dirigido.

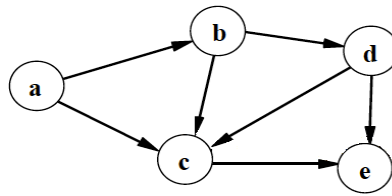


Figura 3.2.2: Grafo dirigido.

A Teoria dos Grafos iniciou-se em 1735, através da publicação de um artigo pelo matemático suíço Leonard Euler, acerca do famoso problema das Sete Pontes de Königsberg, que é representado na Figura 3.2.3 [EncBrit11].

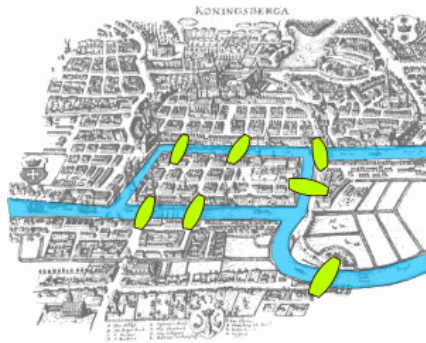


Figura 3.2.3: Problema das Sete Pontes de Königsberg – Euler, 1735. ¹²

Este problema questionava a possibilidade de atravessar as sete pontes, ilustradas na Figura 3.2.3 sem repetir nenhuma. Em 1735 Euler conseguiu provar que tal era impossível. Para provar a sua teoria, Euler criou aquele que é conhecido como o primeiro grafo da história, onde os caminhos são rectas e as intersecções são pontos (Figura 3.2.4) [EncBrit11].

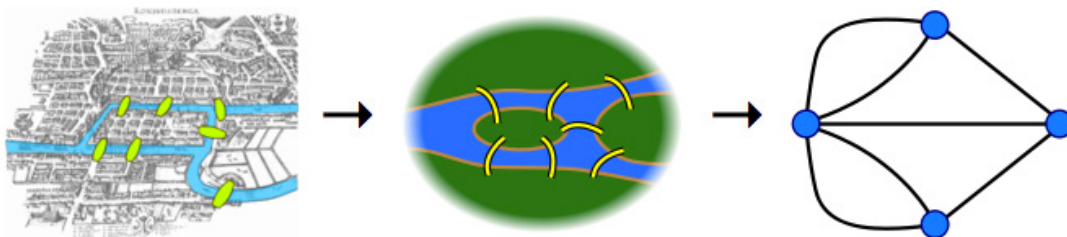


Figura 3.2.4: Problema das Sete Pontes de Königsberg traduzido para grafo. ¹²

¹² Seven Bridges of Königsberg – Wikipedia. 2011. Disponível em: http://en.wikipedia.org/wiki/Seven_Bridges_of_Königsberg Última consulta em 18 de Janeiro de 2011.

Actualmente os grafos possuem muitas aplicações no quotidiano, como por exemplo na representação de um mapa de estradas, sendo utilizados algoritmos para calcular rotas, como por exemplo o algoritmo do caminho mais curto.

Desta forma, as arestas de um grafo podem possuir um valor ao qual se chama custo ou peso, que poderá representar no exemplo do mapa de estradas a distância ou o tempo de percurso entre dois locais, representados por dois nós no grafo. Um caminho num grafo é representado por uma origem e um destino, mesmo que não sejam nós adjacentes, sendo possível existirem diferentes caminhos para a mesma origem e destino. O peso de um caminho corresponde ao somatório de todos os pesos de cada uma das arestas desse caminho [Ant05]. Um grafo é considerado conexo no caso de ser possível obter pelo menos um caminho entre quaisquer dois nós e é considerado acíclico se não possuir ciclos, ou seja existir somente um caminho entre quaisquer dois pontos. É importante ainda referir o conceito de grafo bipartido que consiste num grafo onde é possível dividir os nós em dois conjuntos disjuntos, onde todos os nós de um conjunto se ligam a apenas um outro nó do outro conjunto, e ainda o grafo planar, que é um grafo possível de representar (no plano) sem que haja arestas sobrepostas.

Na matemática existe um ramo denominado teoria dos grafos que tem por base o estudo das relações entre os objectos de um determinado conjunto, bem como os algoritmos para determinar caminhos dentro dos grafos. A teoria dos grafos utiliza dados como o conjunto de vértices e o conjunto de arestas. No caso da Figura 3.2.1 e da Figura 3.2.2, o conjunto de nós é representado pela equação (3.2.2).

$$V = \{a, b, c, d, e\} \quad (3.2.2)$$

O conjunto de arestas é representado pela equação (3.2.3).

$$E = \{\{a, b\}, \{a, c\}, \{b, c\}, \{b, d\}, \{d, c\}, \{c, e\}, \{d, e\}\} \quad (3.2.3)$$

No caso da Figura 3.2.2, em que o grafo é dirigido, o conjunto E também define as suas setas, uma vez que cada subconjunto está na forma $\{origem, destino\}$. O grau de um nó consiste no número de arestas que são incidentes num nó. Assim, no exemplo representado, os nós a e e possuem grau 2, os nós b e d grau 3 e o nó c grau 4 [Sed02].

Para se representar computacionalmente um grafo existem duas abordagens diferentes, a matriz de adjacências e as listas de adjacências. Para poder decidir qual delas a melhor, é necessário saber se o grafo em questão é esparso ou denso. Um grafo é esparso se o seu número de arestas (A) for menor que o número de nós (N) ao quadrado ($A < N^2$). Um grafo denso consiste num grafo cujo número de arestas é bastante próximo do número de nós ao quadrado ($A \approx N^2$) [Muh10].

No caso da matriz de adjacências, é utilizada uma matriz quadrada, sendo o seu número de linhas (l) e colunas (c) igual ao número de nós do grafo em questão, logo é uma matriz simétrica. Cada posição poderá ter o número 0 no caso de não haver aresta entre o nó c e o nó

l correspondente, ou o número 1 no caso de existir aresta. No caso dos grafos com pesos, em vez de 1 coloca-se o valor do peso. Nos grafos dirigidos, coloca-se 1 ou o valor do peso, somente na triangular superior ou inferior da matriz de adjacência, conforme o sentido da ligação. As seguintes matrizes de adjacência correspondem ao exemplo apresentado, sendo a da esquerda para o grafo não-dirigido da Figura 3.2.1 e a da direita para o grafo dirigido da Figura 3.2.2.

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

As vantagens da matriz de adjacências são [IneID05]:

- É a melhor representação quando há espaço disponível, grafos muito densos e os algoritmos requerem mais do que N^2 operações (N é o total de nós do grafo)
- A adição e remoção de nós são feitas de forma eficiente
- Evita-se facilmente a existência de arestas paralelas
- Determina-se rapidamente a existência de ligação entre dois nós

As desvantagens da matriz de adjacências são [IneID05]:

- Grafos esparsos de grande dimensão requerem espaço de memória proporcional a N^2
- Pode não existir memória suficiente para armazenar a matriz

Conclui-se assim que as matrizes de adjacência são o melhor método para representar computacionalmente grafos densos.

No caso das listas de adjacências, cada um dos nós possui uma lista ligada, sendo que os elementos constituintes da lista de cada vértice são os seus vértices adjacentes. Assim, num grafo em que os nós v e w sejam adjacentes, w pertencerá à lista de v e v à lista de w . Na Figura 3.2.5 está representado um exemplo de listas de adjacências para um grafo.

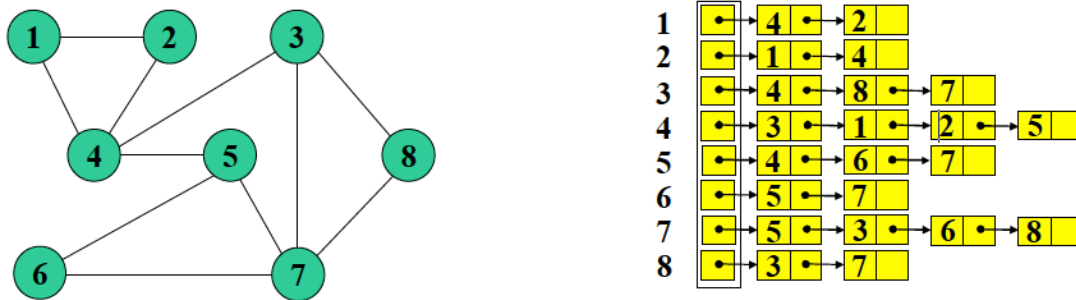


Figura 3.2.5: Exemplo de listas de adjacências.¹³

O grafo representado na Figura 3.2.5 tem 8 nós numerados de 1 até 8, logo existem 8 listas de adjacências, uma para cada nó. Cada uma das listas tem os nós que são adjacentes.

As vantagens da utilização de listas de adjacências são [IneID05]:

- Inicialização é proporcional a N (número de nós)
- Utiliza-se espaço proporcional a $N + A$ (número de nós + número de arestas)
- A adição de arestas é mais eficiente

As desvantagens da utilização de listas de adjacências são [IneID05]:

- Permite a existência de arestas paralelas, podendo ser necessário pesquisar todas as listas, o que pode demorar muito tempo
- A remoção de arestas pode demorar um tempo proporcional a N
- Não é aconselhável para grafos de grandes dimensões que não possam ter arestas paralelas e grafos em que tenham de ser removidas muitas arestas

Assim, conclui-se que o método das listas de adjacências é o mais adequado para representar computacionalmente grafos esparsos.

Para cada problema relacionado com grafos terá de se ponderar qual o tipo de abordagem a ser utilizada.

¹³ INESC-ID. Teoria de Grafos e Algoritmos em Grafos. 2005. Disponível em: <http://algorithms.inesc-id.pt/aed06/downloads/Slides/11-GrafosA.pdf> Última consulta em 1 de Junho de 2011.

Existem diversos algoritmos para encontrar caminhos em grafos, sendo os mais populares o algoritmo de Dijkstra, algoritmo de Kruskal, algoritmo de Prim e algoritmo do vizinho mais próximo.

Os grafos, no contexto deste projecto, serão utilizados para representar computacionalmente as redes filogenéticas, onde os nós representam as sequências existentes e as arestas representam as mutações existentes entre as sequências. São a estrutura ideal para representar as redes, já que permitem a existência de ciclos, representando assim as reticulações existentes, e já existem alguns algoritmos utilizados em grafos que são adaptados para as redes filogenéticas (como o algoritmo *median-joining network* e *reduced-median network* [BFR99], que será apresentado na Secção 3.4).

Tal como já tinha sido referido antes, utiliza-se nesta área um *software* denominado Network 4.6 que, através das sequências previamente alinhadas, gera a rede filogenética correspondente utilizando um dos dois principais algoritmos (*median-joining network* ou *reduced-median network*), o que torna esta aplicação única. No Anexo A é descrita a utilização desta aplicação, sendo apresentadas algumas captações da sua utilização, bem como os três principais formatos de ficheiros utilizados para gerar e representar as networks:

- .rdf, que significa “DNA data file”. Contém as sequências de DNA alinhadas. É gerado por aplicações de *software* que fazem alinhamento de sequências, como por exemplo o FASTA. É o ponto de partida para calcular a rede filogenética.
- .out: “MJ or RM out file”. É o ficheiro que contém a representação da rede filogenética, criada através da aplicação do algoritmo *median-joining network* ou *reduced-median network*.
- .fdi: “Formatted Diagrams”. É um ficheiro que contém toda a informação presente no formato out e ainda informações visuais, ou seja posicionamento dos nós, cores de regiões geográficas, entre outros.

No Anexo A.1 poder-se-á compreender melhor toda esta informação, através da observação de um diagrama representativo do fluxo de trabalho do Network 4.6 e ainda um exemplo de cada um dos formatos de ficheiro.

3.3 Árvores

Uma árvore consiste num grafo com as seguintes características: é conexo, acíclico, bipartido e planar. Toda a árvore é um grafo, mas não se verifica o inverso. Para qualquer árvore, com n nós, o seu número de arestas será igual a $n - 1$ [Kli04, Roc08]. Na Figura 3.3.1 é apresentada uma árvore com 6 nós e 5 arestas.

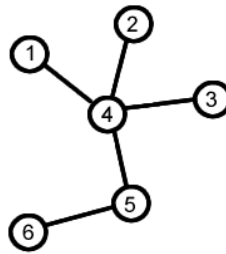


Figura 3.3.1: Árvore com 6 nós e 5 arestas.

A árvore poderá ter um nó raiz (Figura 3.3.2). Nesse caso, esse nó estará normalmente no topo da árvore e os nós subjacentes a esse são os seus filhos, pelo que a raiz será o pai destes. Devido às características anteriormente referidas, existe apenas um caminho desde a raiz até qualquer outro nó da árvore. Os nós mais abaixo da árvore, que não possuam qualquer descendente são conhecidos como as folhas da árvore [Kli04]. Cada nó possui o comprimento ou profundidade, que é o número de arestas percorridas desde a raiz até ao nó, sendo que a profundidade da raiz é 0 e sendo p a profundidade de um nó, a profundidade de um outro nó que seja seu filho será $p + 1$. Outro conceito utilizado é a altura de nó que consiste no comprimento de nó até à folha de maior profundidade. Assim sendo, uma folha terá altura igual a zero.

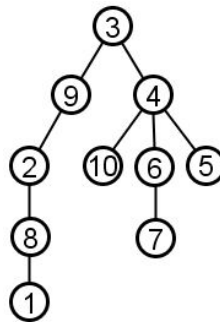


Figura 3.3.2: Árvore com raiz.

No contexto deste projecto, as árvores servirão para representar as árvores filogenéticas, já que nestas não existem reticulações. Será necessário obter todas as árvores filogenéticas possíveis a partir de uma rede filogenética, de modo a poder simplificar as reticulações e escolher assim a melhor opção, ficando-se com várias representações em árvore e, como tal, sem qualquer ciclo. Depois será realizado um sistema de cálculo de *ranking* de cada árvore, de forma a ajudar o especialista a escolher qual a melhor alternativa na reticulação da rede existente.

No que diz respeito à representação informática de árvores filogenéticas, existe um formato de ficheiro denominado “Newick Tree Format”, que consiste num formato simples e

intuitivo para representar árvores filogenéticas. Foi proposto em 1857 por Arthur Cayley, um matemático inglês famoso [Fel10]. A título de exemplo apresenta-se na Figura 3.3.3 um exemplo de uma árvore, cuja representação em Newick será (B,(A,C,E),D);

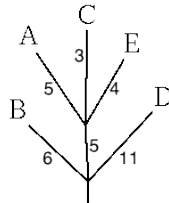


Figura 3.3.3: Exemplo de Árvore.

O nó mais profundo estará sempre na parte mais interior de todos os parênteses. Os nós menos profundos ficarão nas extremidades. O ponto e vírgula (;) é o sinal de terminação da definição de uma árvore. É possível guardar ainda os valores do peso de cada ramo, especificando a seguir ao nome do nó o valor do ramo imediatamente antes desse mesmo nó. Para o exemplo apresentado na Figura 3.3.3, será: (B:6.0,(A:5.0,C:3.0,E:4.0):5.0,D:11.0);. É ainda possível definir antepassados: (B:6.0,(A:5.0,C:3.0,E:4.0)Antepassado1:5.0,D:11.0); [Fel10].

Este formato é utilizado nas principais aplicações de *software* utilizadas na área para visualizar e analisar árvores filogenéticas, tais como TreeView, BEAST, PHYLIP e PAUP*.

3.4 Algoritmos Utilizados na Área

Para construir árvores filogenéticas existem diversos métodos. Existem os métodos algorítmicos, que calculam uma única árvore a partir dos dados, utilizando um protocolo específico e repetível. Dentro deste tipo, o método *neighbor-joining* que presume diferentes taxas de evolução de sequências de DNA é o mais utilizado. Os métodos algorítmicos têm a vantagem de produzirem o mesmo resultado a partir dos mesmos dados de entrada, para além de serem repetíveis e rápidos computacionalmente e nalguns casos são a única alternativa prática, no caso de haver um grande conjunto de dados.

Contudo, muitos profissionais da área preferem os métodos do tipo *tree-searching*, que são capazes de examinar uma grande amostra do número de árvores possíveis mesmo para poucas categorias taxonómicas. Nestes métodos utilizam-se rotinas de pesquisa que maximizam a hipótese de encontrar as mais pequenas árvores compatíveis com os dados de entrada. Ao contrário dos métodos algorítmicos, os métodos *tree-searching* podem obter como resultados diferentes árvores, mesmo que a diferença seja substancial. Um dos métodos deste tipo mais utilizados é o método da máxima parcimónia. A parcimónia consiste em escolher sempre a

explicação mais simples, que requer o menor número de pressupostos desconhecidos. Assim, sempre que se constrói uma árvore filogenética, existem diversas hipóteses de a representar, mas a que deverá ser escolhida será aquela que possua menor número de homoplasias. Outros métodos também muito utilizados do tipo *tree-searching* são o *maximum-likelihood* e os métodos bayesianos [Fut10]. Ambos utilizam um modelo de evolução dos dados, ou seja um modelo que assume que todas as substituições nucleotídicas são iguais e é possível estimar, a partir dos dados de entrada, uma constante taxa de substituição, ou então assume que ocorrem diferentes tipos de substituições com diferentes taxas. O método *maximum-likelihood* calcula a verosimilhança e a melhor estimativa da filogenia é a árvore que maximiza a verosimilhança. Os métodos bayesianos diferem do *maximum-likelihood* porque os métodos bayesianos calculam as probabilidades de um conjunto de árvores para que estas possam ser comparadas como uma árvore de consenso [HRN01].

Contudo, apesar de os métodos do tipo *tree-searching* serem preferidos aos métodos algorítmicos, os *tree-searching* não dão a garantia de ser encontrada a árvore mais pequena, nem a árvore que maximiza os dados, por causa do grande número de árvores que têm de ser pesquisadas.

Estes métodos tradicionais foram considerados insatisfatórios quando aplicados a mtDNA humano [BFS95]. Em 1994, Excoffier e Smouse calcularam que o número de árvores com parcimónias equivalentes para 56 haplótipos (62 posições nucleotídicas, sendo 34 delas polimórficas) excederam os mil milhões [ES94]. Já em 1987, Saitou e Nei observaram que o método *neighbor-joining* produz uma árvore dicotómica [SN87], mesmo quando a árvore filogenética do mtDNA possui algumas ramificações altamente politómicas. A elevada taxa de mutação no mtDNA origina eventos de polimorfismo em paralelo (recorrência), que fazem com que o caminho evolutivo correcto possa ser o não escolhido na árvore seleccionada pelos métodos descritos anteriormente. Assim, foi criado em 1995 um novo método, denominado *reduced-median network* [BFS95], em que para uma pequena quantidade de amostras, é obtida uma rede filogenética, contendo todas as possíveis árvores de máxima parcimónia. São geradas várias reticulações envolvendo as posições recorrentes. A complexidade de reticulação pode ser resolvida em parte pela aplicação de um *reduced factor* que identifica os fenómenos de paralelismo mais frequentes atendendo também às frequências dos haplótipos mitocondriais, e obrigando certas mutações a aparecer várias vezes ao longo da evolução, escolhendo assim caminhos de evolução.

Em 1999, foi apresentado um novo método, *median-joining network* utilizado para construir redes filogenéticas, combinando os algoritmos de Kruskal (árvore de expansão mínima) e o da máxima parcimónia. Consiste em adicionar sequencialmente novos nós à rede, denominados “median vectors”, que consistem em pontos intermédios na evolução, para os quais não foi observado nenhum exemplo na amostragem em estudo. A principal diferença em relação ao algoritmo *reduced-median network* é que neste não se toma nenhuma decisão para resolver as reticulações, sendo representadas todas as possíveis árvores filogenéticas [BFR99].

Estes dois últimos algoritmos, *reduced-median network* e *median-joining network*, estão implementados no *software* Network, podendo ser utilizados para representar sequências previamente alinhadas.

3.5 Estado da Arte da Análise de Fundador

Os autores da Análise de Fundador desenvolveram quatro critérios para decidir se uma sequência presente nas populações de origem e destino consiste ou não num fundador (f0, f1, f2, fs), com base na diversidade dessa sequência específica na população de origem. Pode-se observar as diferenças entre os três primeiros critérios na Figura 3.5.1. O critério f0 considera todas as sequências presentes em ambas as populações como possíveis fundadores sem nenhuma condição adicional [RMH00]. O critério f1 considera que, para ser fundador, terá de existir pelo menos um ramo derivado na população de origem, enquanto f2 considera que terão de existir pelo menos dois ramos derivados na população de origem. O critério fs considera que o número mínimo de ramos derivados na população de origem, para cada fundador, é estipulado pelo logaritmo de base 10 do número de ramos descendentes, desse fundador, arredondado para o inteiro mais próximo e somando 1.

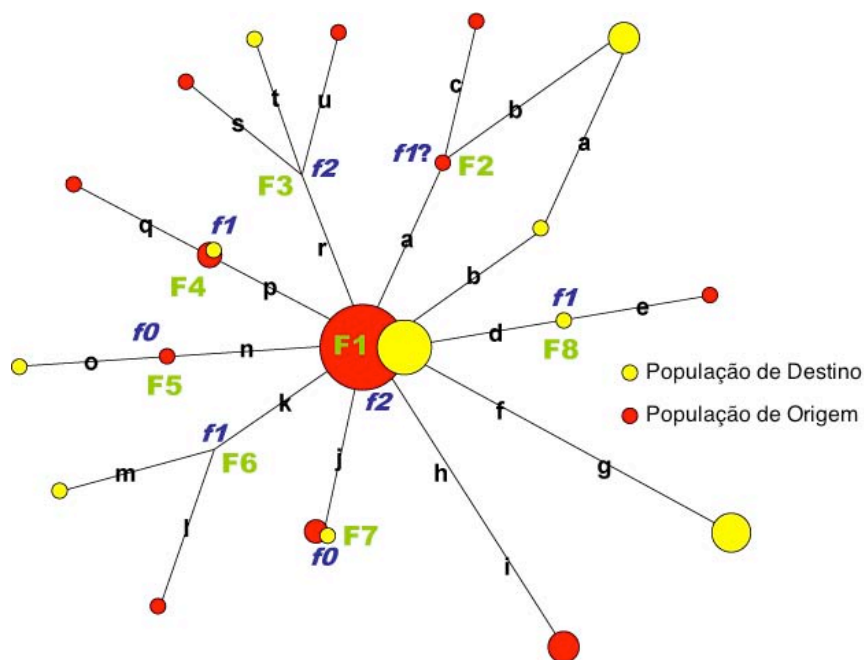


Figura 3.5.1: Diferentes critérios de cálculo de fundadores.

Para estimar a idade do fundador, utiliza-se a diversidade específica da população de destino. Assim, a Análise de Fundador pretende identificar os últimos antepassados em comum entre a população de origem e de destino, sendo depois o cálculo da idade do fundador obtido a

partir dos seus seguintes sub-ramos que pertençam à população de destino [Soa09]. O trabalho já realizado nesta área por Vincent Macaulay e Noel Thomson na Universidade de Glasgow revelou que a Análise de Fundador tem tendência a subestimar a idade actual do fundador através do tempo decorrido desde a chegada de uma linhagem à região e a ocorrência da primeira mutação.

No que diz respeito a fórmulas matemáticas utilizadas para estimar idades, a Equação (3.5.1) serve para estimar a distância média, em mutações, desde uma amostra até à raiz ou até ao ancestral mais recente na árvore ou rede filogenética.

$$\rho = \frac{\sum_{i=1} m_i}{n} \quad (3.5.1)$$

sendo m_i as mutações existentes e n o número total de amostras.

O cálculo da variância era feito a partir da Equação (3.5.2), originária de Forster et al., 1996 [FHT96]. Foi utilizada em 2000 quando a Análise de Fundador foi publicada [RMH00].

$$\sigma^2 = \frac{\rho}{n} \quad (3.5.2)$$

No entanto verificou-se que a maior parte das vezes, a Equação (3.5.3) não é adequada, pois parte-se do princípio que as redes filogenéticas sejam em forma de estrela, o que não acontece em muitos casos [Soa09]. Por esta razão foi proposta uma nova fórmula, representada na equação (3.5.3), para o cálculo da variância [SFL00].

$$\sigma^2 = \frac{\sum_{i=1}^k n_i^2 l_i}{n^2} \quad (3.5.3)$$

sendo l_i o comprimento de cada ramificação entre os vários nós, n_i o número de amostras associadas com cada comprimento de ramificação l_i , k cada nó da rede filogenética e n o número total de amostras.

Esta última fórmula tem a capacidade de apresentar mais realisticamente o desvio padrão. Tem em conta que alguma da variância não é independente e no caso de a rede filogenética ser em forma de estrela, a variância é a mesma que na definição original. Esta fórmula necessitava de ser implementada na Análise de Fundador, e era um dos requisitos deste projecto, sendo também uma grande melhoria para o método a ser analisada com o co-autor.

3.5.1 Informatização da Análise de Fundador

A Análise de Fundador é uma metodologia que tem sido aplicada “manualmente”, ou seja, é utilizado o Network 4.6, para representar a informação filogenética da população e com base nessa observação, são retirados dados para uma folha de cálculo em Excel, para depois serem calculadas estatísticas relevantes. De referir ainda que no caso de existirem reticulações, ou seja,

se a informação estiver representada numa rede filogenética, o especialista, terá de ponderar qual o melhor caminho na rede e transformá-la numa árvore. É ainda necessário identificar, de forma manual, quais os fundadores existentes nessa árvore, perdendo-se assim ainda mais tempo para aplicar esta metodologia.

Com este projecto informatizou-se este processo, através da criação de uma aplicação que lê o *output* do programa Network, cria as melhores árvores filogenéticas possíveis a partir das reticulações da rede filogenética e sugere qual delas a melhor solução, sendo ainda possível ao especialista, que esteja a utilizar o programa, fazer uma escolha diferente. De seguida, a partir da árvore filogenética escolhida, a aplicação em questão fará a identificação dos fundadores e o cálculo de todas as estatísticas necessárias, sendo os dados obtidos automaticamente pelos ficheiros de *input* já utilizados pela aplicação Network, sem que haja a necessidade de um processo de cópia manual de dados entre aplicações, moroso e sujeito a erros, como anteriormente.

Estima-se que, com esta aplicação, a poupança de tempo seja de grandes dimensões, uma vez que é possível fazer em aproximadamente 2 dias aquilo que anteriormente se faria em pelo menos 2 meses.

3.6 Resumo

Após a leitura deste capítulo, já é possível o leitor ter uma ideia completamente clara do que consiste a Análise de Fundador, e quais os seus temas adjacentes. Foi também exposto o estado da arte de todos estes temas, por forma a expor a obra já realizada e as lacunas que faltam preencher, sendo assim apresentado o objectivo principal deste projecto.

Capítulo 4

Redes e Árvores Filogenéticas

A primeira grande decisão a ser tomada ao nível da implementação foi a escolha da linguagem de programação para desenvolver a aplicação pretendida. Após a análise de várias alternativas, foi escolhido o Java devido ao facto de ser uma linguagem que permite uma grande portabilidade nas aplicações que se desenvolvem e ainda por ser necessária uma interface com o utilizador bastante amigável e com várias opções disponíveis. A biblioteca Swing do Java permite criar interfaces bastante amigáveis, o que pesou bastante na escolha. Resta ainda acrescentar que a eficiência não é uma questão importante.

4.1 Implementação das Árvores e Redes

Após ter sido escolhida a linguagem de programação, iniciou-se a implementação, através da leitura de ficheiros construídos pelo Network 4.6, ou seja ficheiros no formato fdi e out.

Ao nível da implementação foi necessário decidir o tipo de estrutura de dados a ser utilizada para representar as redes filogenéticas. Como uma rede filogenética consiste num grafo, em termos de representação computacional, teve de ser escolhida uma das duas formas usuais para representar grafos, que são a matriz de adjacências e as listas de adjacências.

Cada uma destas formas é adequada a diferentes tipos de grafos. Assim, como as redes filogenéticas consistem em grafos bastante esparsos, foi escolhida a utilização de listas de adjacências, com base nas características deste método, descritas na Secção 3.2, tendo sido implementadas no Java através da estrutura de dados LinkedList. A representação de árvores filogenéticas foi realizada da mesma forma que os grafos, uma vez que as características são bastante semelhantes, à excepção dos ciclos que são inexistentes nas árvores.

Concluída a implementação da representação dos grafos e árvores, o passo seguinte foi a escolha do método para gerar árvores a partir de redes filogenéticas. O que se pretende é obter

uma árvore filogenética que na prática é muito próxima da árvore de expansão mínima, que consiste numa árvore com o menor custo total possível, obtida através de um grafo, de onde são eliminadas arestas que constituem ciclos. As arestas da árvore obtida serão as que têm menor custo de todas, no entanto não é garantido o caminho mais curto entre quaisquer dois nós.

Há dois critérios a satisfazer:

- i. Usar as mutações mais frequentes;
- ii. O caminho da raiz a um nó deve ter o maior número de indivíduos.

Estes critérios envolvem informação relativa às arestas da rede (mutações) e aos nós (número de indivíduos).

Após analisar vários algoritmos de árvore de expansão mínima, optou-se pelo algoritmo de Prim, uma vez que a sua implementação é simples e rápida. A alternativa estudada foi o algoritmo de Kruskal, que tal como o de Prim, é um algoritmo ganancioso (*greedy algorithm*), ou seja é um algoritmo que toma sempre aquela que é a melhor decisão no momento. Outro factor preponderante, que favoreceu a escolha do algoritmo de Prim, foi a facilidade de implementação; como o procedimento é ir acrescentando novos nós juntamente com arestas à árvore já existente, ao invés de ir adicionando arestas sem seguir uma linha de conexão em concreto (como no algoritmo de Kruskal), é mais fácil verificar que não se formam ciclos na árvore. Foi também implementado o algoritmo de Dijkstra para comparar resultados em relação ao algoritmo de Prim. Uma vantagem que o algoritmo de Dijkstra irá ter em relação ao de Prim, e mesmo ao de Kruskal, é que o de Dijkstra encontrará garantidamente o caminho mais curto entre a raiz e os restantes nós do grafo, o que poderá não acontecer num algoritmo cujo resultado seja a árvore de expansão mínima (Prim e Kruskal), senão observemos a Figura 4.1.1, onde temos um grafo com 4 nós e 4 arestas, estando também representado o peso de cada uma das arestas. A raiz do grafo é o nó A.

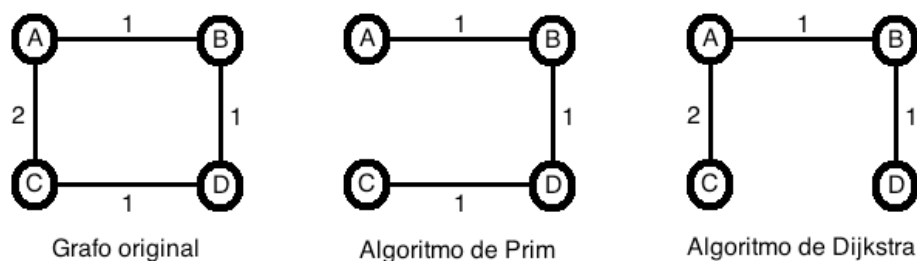


Figura 4.1.1: Diferenças na aplicação dos algoritmos de Prim e Dijkstra.

Como o algoritmo de Prim obtém as arestas com menor peso, acaba por obter a árvore de expansão mínima do grafo original, como representado na Figura 4.1.1. No entanto o peso entre A e C é 3 nesta árvore obtida, o que não é o caminho mais curto, pois a aresta entre A e C tinha peso 2. No algoritmo de Dijkstra tal não acontece, uma vez que se obtêm os caminhos mais curtos desde a raiz até qualquer outro nó do grafo original. Assim, poderão existir casos onde se

obtenham resultados mais satisfatórios com o algoritmo de Dijkstra e outros com o algoritmo de Prim.

No Anexo B.2 é apresentado o diagrama de classes representativo da estrutura do código-fonte da aplicação.

Os pesos atribuídos a cada aresta das redes filogenéticas são calculados com base na frequência da mutação (ou mutações) que existe(m) nessa aresta. No artigo de Soares et al., 2009 [SET09] foi publicada uma tabela que associa a cada mutação um valor de ocorrência da mesma numa árvore filogenética, contendo 2196 sequências completas de mtDNA. Este número de ocorrência consiste na frequência de cada mutação. Assim, quanto maior for a frequência de uma mutação, maior será a probabilidade da mesma acontecer, logo menor será o peso dessa aresta no grafo correspondente à rede filogenética.

4.2 Construção de Árvores a Partir de Redes

Depois de ter sido implementada a representação de árvores e redes filogenéticas, passou-se ao desenvolvimento do algoritmo de geração de árvores a partir de redes. Como já foi referido, foi escolhido o algoritmo de Prim para calcular a árvore de expansão mínima do grafo representativo da rede filogenética. Foi também implementado o algoritmo de Dijkstra para calcular árvores com o caminho mais curto desde a raiz. O resultado obtido por cada um destes dois algoritmos poderá não ser o melhor, do ponto de vista dos especialistas de genética populacional. Isto porque as redes filogenéticas têm algumas características que as diferem dos grafos, como por exemplo:

- Os nós das redes filogenéticas possuem uma frequência associada, representativa do número de indivíduos com a sequência de DNA desse nó. Por esta razão, os nós das redes filogenéticas têm diferentes tamanhos, como demonstrado na Figura 4.2.1.
- Existem nós com frequência nula (*median vectors*), ou seja, representam sequências hipotéticas, não verificadas em nenhum indivíduo.
- Os ciclos são sempre de 4 nós, sendo os pesos das arestas iguais 2 a 2, tal como representado na Figura 4.2.1.

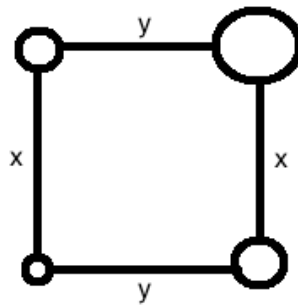


Figura 4.2.1: Tipo de ciclos (reticulações) presentes nas redes filogenéticas, diferentes tamanhos de nós e diferentes custos das arestas.

4.2.1 Algoritmo de Prim

Apresenta-se de seguida o pseudo-código do algoritmo de Prim, adaptado para este problema.

Algoritmo de Prim

Entrada: Grafo

Saída: Árvore de expansão mínima

Ordenar arestas por ordem crescente de custo

Lista de nós na árvore = {nó inicial}

Enquanto lista de nós na árvore não tiver todos os nós do grafo FAZER:

Escolher aresta de menor custo que liga 1 nó na lista a 1 nó 1 nó que não está na lista

Adicionar o nó que não está na lista à lista

Retornar árvore

Se o número de árvores pretendido ainda não foi excedido

Fazer *backtracking* para arranjar outra árvore possível, sem ultrapassar o custo total permitido

A utilização “directa” do algoritmo de Prim não produz necessariamente uma árvore filogenética que seja aceite como tal pelos especialistas. No entanto produz uma árvore que está, geralmente, muito próxima da árvore correcta sob o ponto de vista filogenético. As razões para esta discrepância são as duas seguintes. Havendo diversas arestas com o mesmo custo na árvore de expansão mínima não é necessariamente única. É preciso analisar (ordenar por preferência) as alternativas. Existem mutações menos frequentes (reversões), que por vezes os especialistas incluem na árvore final.

Esta situação levou-nos a propor uma adaptação do algoritmo de Prim, de modo a:

- Gerar árvores alternativas.

- Ordenar as árvores alternativas com regras que traduzem as preferências dos especialistas.

O algoritmo de Prim foi tomado como algoritmo de base, com as seguintes inovações. O algoritmo gera todas as árvores alternativas. O algoritmo recebe como parâmetro um custo e só devolve as árvores de expansão com o custo especificado. Foi implementado um “meta ciclo” que chama o algoritmo de base com valores crescentes de custo, começando com o custo mínimo.

4.2.2 Algoritmo de Dijkstra

De seguida apresenta-se o algoritmo de Dijkstra, adaptado para resolver este problema.

Algoritmo de Dijkstra

```
Para cada nó do grafo
    Definir distância do nó infinita
    Colocar os nós anteriores como não visitados
Definir distância da raiz igual a zero
Colocar todas as arestas numa lista de arestas não usadas
Criar uma lista ordenada Q com os nós do grafo
Percorrer lista Q enquanto não estiver vazia
    Obter nó N, com a menor distância
    Se a distância de N for infinita, sai do ciclo (todos os nós restantes estão
        inacessíveis)
    Remover N da lista
    Para cada vizinho V de N
        Se (distância de N + distância de N a V) for menor que distância actual de V
            Adicionar aresta de N para V à árvore
            Remover aresta da lista das não usadas
            Reordenar V na lista Q
Retornar árvore obtida
Se o número de árvores pretendido ainda não foi excedido
    Fazer backtracking para arranjar outra árvore possível, sem ultrapassar o custo total
    permitido
```

4.2.3 Melhoria aos Algoritmos

Pretende-se com estes dois algoritmos obter a melhor árvore possível do ponto de vista destes algoritmos, bem como um conjunto de alternativas, uma vez que o resultado do algoritmo poderá não ser o óptimo aos olhos dos especialistas desta área. Assim, é possível escolher dentro de várias alternativas. Como forma de melhorar a eficiência dos algoritmos, já que, para árvores de maior dimensão, poderão demorar muito tempo a serem executados, apresenta-se uma melhoria feita nos algoritmos, que consiste em retirar da árvore os nós que sejam folha no grafo (e na árvore), ou seja nós que só tenham uma aresta no grafo. Como estes nós só possuem uma aresta, nunca estarão envolvidos em ciclos, logo sendo retirados do grafo, irá ser reduzido o número de arestas a analisar e, conseqüentemente, o algoritmo ficará mais eficiente.

Apresenta-se na Figura 4.2.2 um exemplo deste tipo de nós com caminho único existentes nas redes. O nó E é um destes nós, já que só possui uma aresta com o nó D.

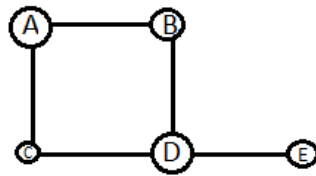


Figura 4.2.2: Remoção de nós com caminho único.

Apresenta-se de seguida o pseudo-código desta melhoria. Irá ficar situada na parte onde se percorrem os nós para a obtenção da primeira árvore.

Se for a primeira árvore do algoritmo

Se V tiver uma única aresta (nó V é uma folha da árvore)

Adicionar aresta de N para V à lista das folhas

Remover aresta de N para V da árvore e do grafo

4.3 Algoritmo de Ranking de Árvores

Como é necessário resolver todos os ciclos existentes numa rede, será necessário obter as melhores árvores a partir de uma rede, o que é contemplado pelos algoritmos expostos na Secção 4.2, para posteriormente o especialista escolher a melhor alternativa. Para apoiar a decisão do especialista, este sugeriu um conjunto de regras que permite o desenvolvimento de um algoritmo de *ranking* para árvores, que atribui uma estimativa de qualidade a cada uma. As melhores árvores são mostradas ao especialista, que poderá depois escolher qualquer uma das árvores, sendo o *ranking* apenas para o ajudar na escolha.

Este algoritmo tem em conta os seguintes critérios, fornecidos pelo especialista:

- Numa reticulação deverão ser evitados nós que não estejam preenchidos com sequências observadas, logo caminhos com *median vectors*¹⁴ deverão ser subvalorizados
- Deverão ser escolhidos os caminhos que passem por nós com maior número de indivíduos e com menor número possível de mutações nos ramos, sendo assim respeitada a máxima parcimónia

¹⁴ *Median Vector* – é um ponto intermédio na evolução, para o qual não foi observado nenhum exemplo na amostragem em estudo.

- Como há heterogeneidade na taxa de mutação entre as várias posições de uma sequência genética, esta variabilidade tem que ser tida em conta, dando-se um peso a cada aresta com base no número de ocorrências de cada mutação. No Anexo B.3 apresenta-se a tabela com o número de ocorrências para cada mutação.

4.4 Resultados Experimentais

Para testar o correcto funcionamento do algoritmo que propomos foram utilizados vários casos de teste, que consistem em algumas redes filogenéticas, nas quais foi aplicado o algoritmo implementado na aplicação. Assim, apresentam-se de seguida as redes filogenéticas utilizadas na aplicação e a correspondente árvore filogenética obtida, ou seja a solução obtida com o algoritmo implementado.

4.4.1 Caso de Teste 1

O primeiro caso apresenta-se de seguida. Na Figura 4.4.1 é mostrada a rede filogenética original, com todas as reticulações (ciclos) que necessitam de ser resolvidas.

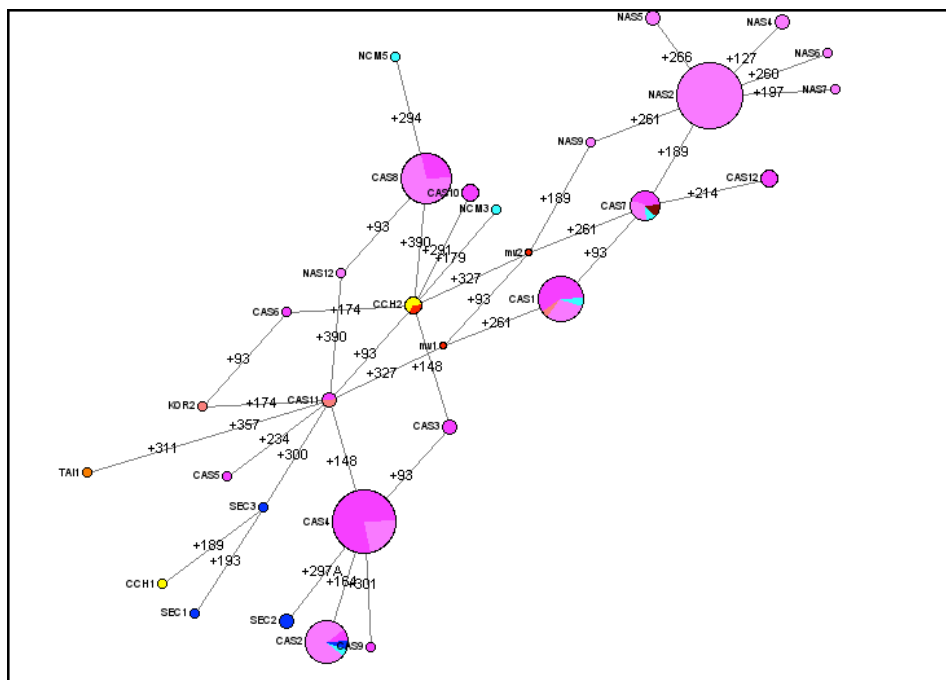


Figura 4.4.1: Caso de teste 1 – Rede filogenética.

Apresenta-se na Tabela 4.4.1 os dados relativos à rede filogenética em questão.

Tabela 4.4.1: Caso de teste 1 – Características da rede filogenética original.

Caso de Teste 1 – Rede Filogenética Original	
Número total de nós	30
Número de <i>median vectors</i>	2
Frequência total	184
Número de categorias taxonômicas	43
Categorias taxonômicas contidas dentro de outros nós	13
Número total de ciclos (reticulações)	6

Na Figura 4.4.2 apresenta-se a árvore obtida através da aplicação do algoritmo de Prim desenvolvido, explicado com maior detalhe na Secção 4.2. O algoritmo foi executado, requisitando-se 3 árvores, ou seja, será gerada a óptima, segundo o algoritmo de Prim e de seguida, através do *backtracking*, serão geradas mais duas árvores alternativas.

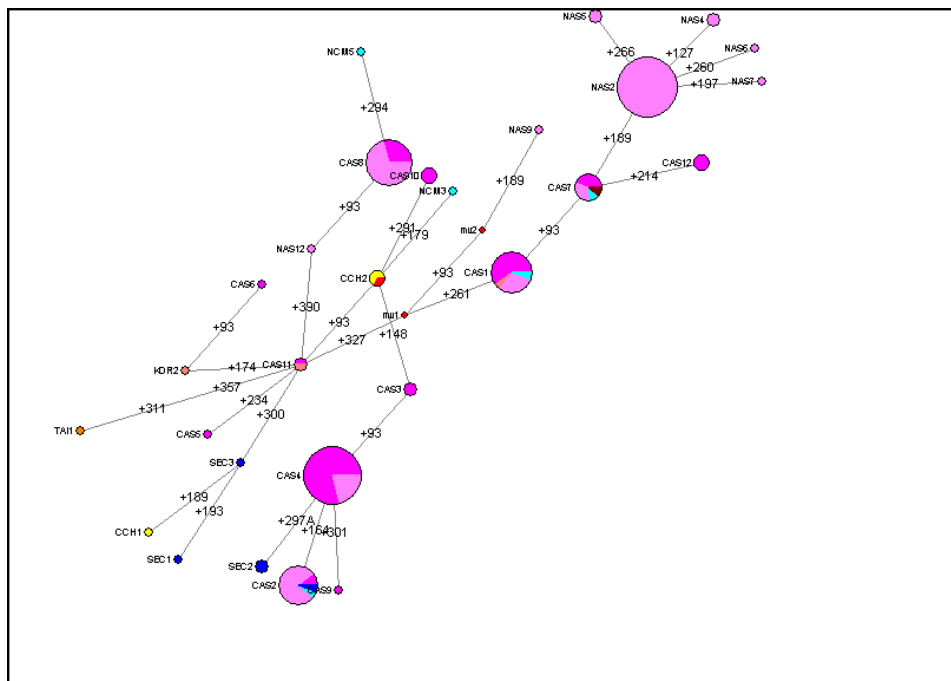


Figura 4.4.2: Caso de teste 1 – Primeira árvore filogenética obtida.

De notar que na árvore obtida, representada na Figura 4.4.2, no caso dos ciclos que existiam, foi-lhes removida uma das arestas com maior peso, ou seja uma daquelas em que a frequência das suas mutações fosse menor, segundo o Anexo B.3.

Para este caso executou-se a aplicação, tendo-se dado com ficheiro de entrada a rede filogenética representada na Figura 4.4.1. Esta figura é a representação da rede no *software*

Redes e Árvores Filogenéticas

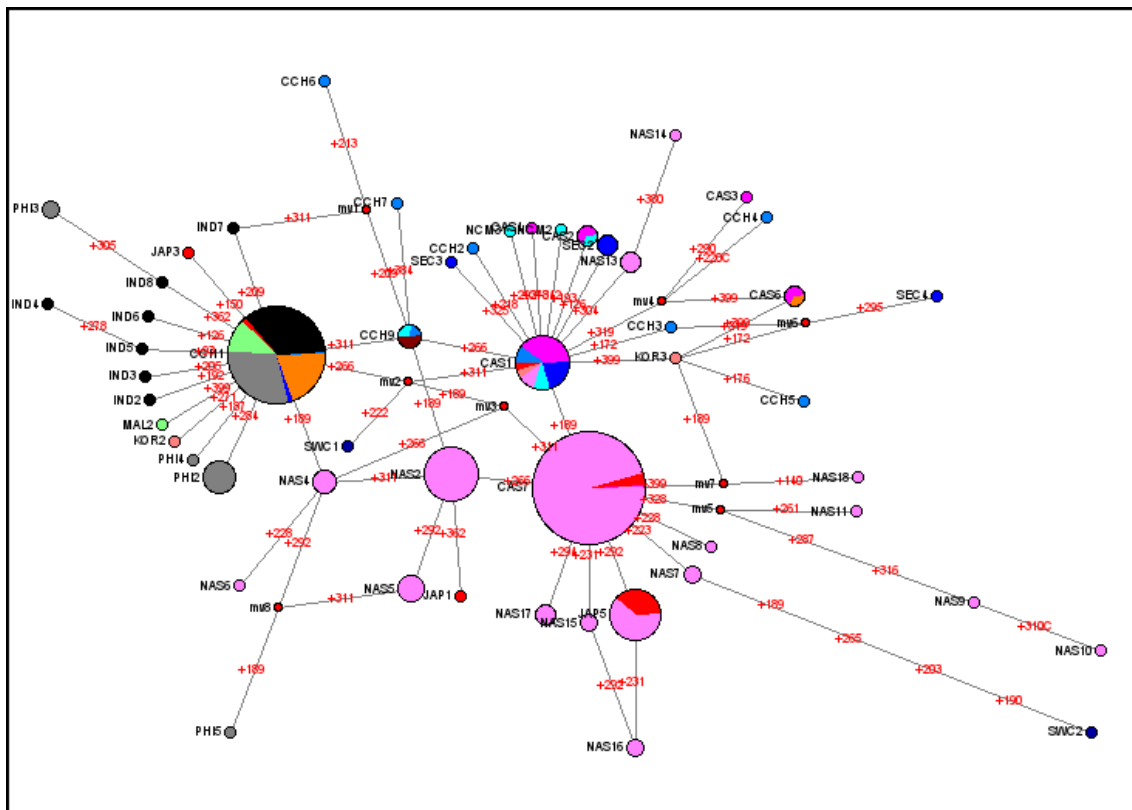


Figura 4.4.5: Caso de teste 2 – Rede filogenética.

Apresenta-se na Tabela 4.4.2 as características desta rede filogenética.

Tabela 4.4.2: Caso de teste 2 – Características da rede filogenética original.

Caso de Teste 2 – Rede Filogenética Original	
Número total de nós	58
Número de <i>median vectors</i>	8
Frequência total	286
Número de categorias taxonômicas	80
Categorias taxonômicas contidas dentro de outros nós	19
Número total de ciclos (reticulações)	10

Apresenta-se na Figura 4.4.13 o resultado obtido após a aplicação do algoritmo de Prim.

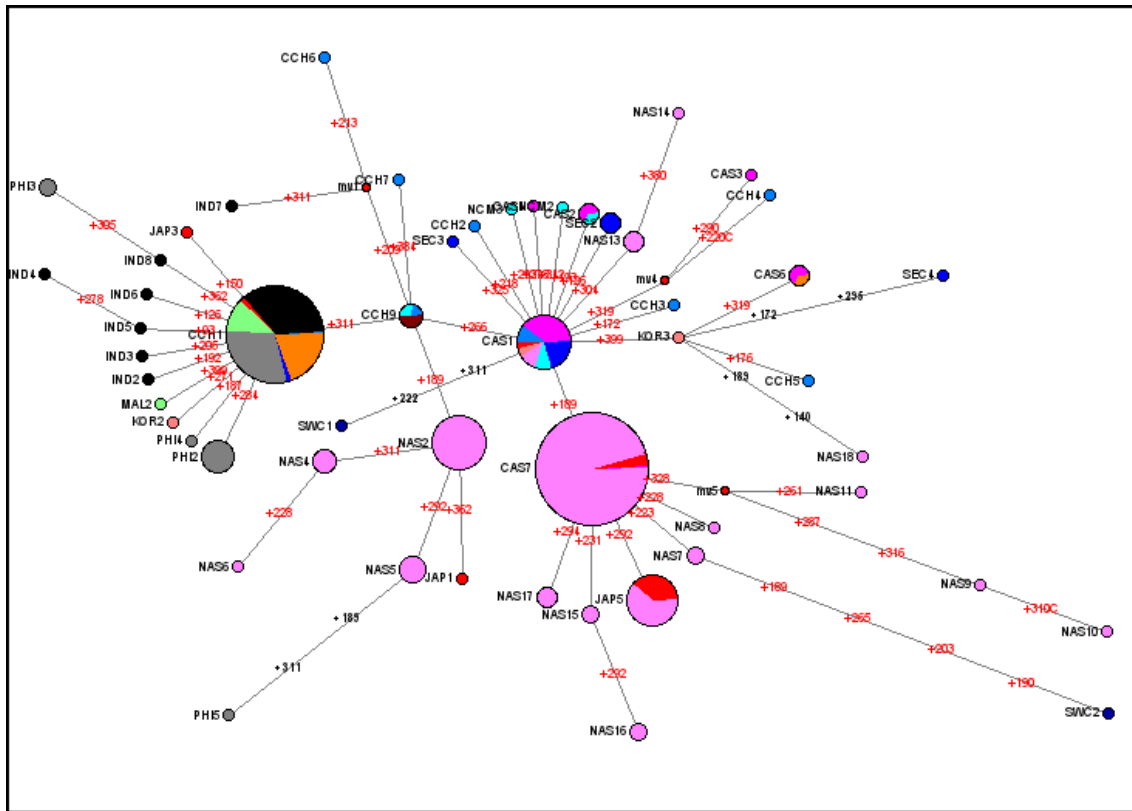


Figura 4.4.6: Caso de teste 2 – Árvore filogenética obtida com o algoritmo de Prim.

Após analisar o resultado, verificamos que houve *median vectors* (*mv*) que desapareceram. Isto aconteceu porque não tem sentido existir um *median vector* apenas com uma ou duas arestas. Para um *median vector* ter razão de existir, terá de estar ligado a pelo menos outros três nós. Assim, os *median vectors* que tinham menos de 3 arestas foram eliminados, e no caso de ter duas arestas, estas foram ligadas uma à outra, formando uma nova aresta, cujas mutações são a junção das mutações de cada uma das arestas anteriormente existentes. Neste caso, o mv2 foi eliminado, formando a aresta SWC1-CAS1, mv5 deu lugar à aresta KOR3-SEC4, mv7 deu lugar à aresta KOR3-NAS18 e mv8 a NAS5-PHI5. Isto aconteceu, porque sendo eliminadas arestas, há *median vectors* que acabam por ficar sem razão para continuarem a existir na árvore. No algoritmo de Dijkstra e na solução do especialista os *median vectors* desnecessários são de igual modo eliminados da árvore.

De seguida, apresenta-se na Figura 4.4.14, o resultado obtido através do algoritmo de Dijkstra e na Figura 4.4.15, apresenta-se a solução proposta pelo especialista de genética populacional.

Redes e Árvores Filogenéticas

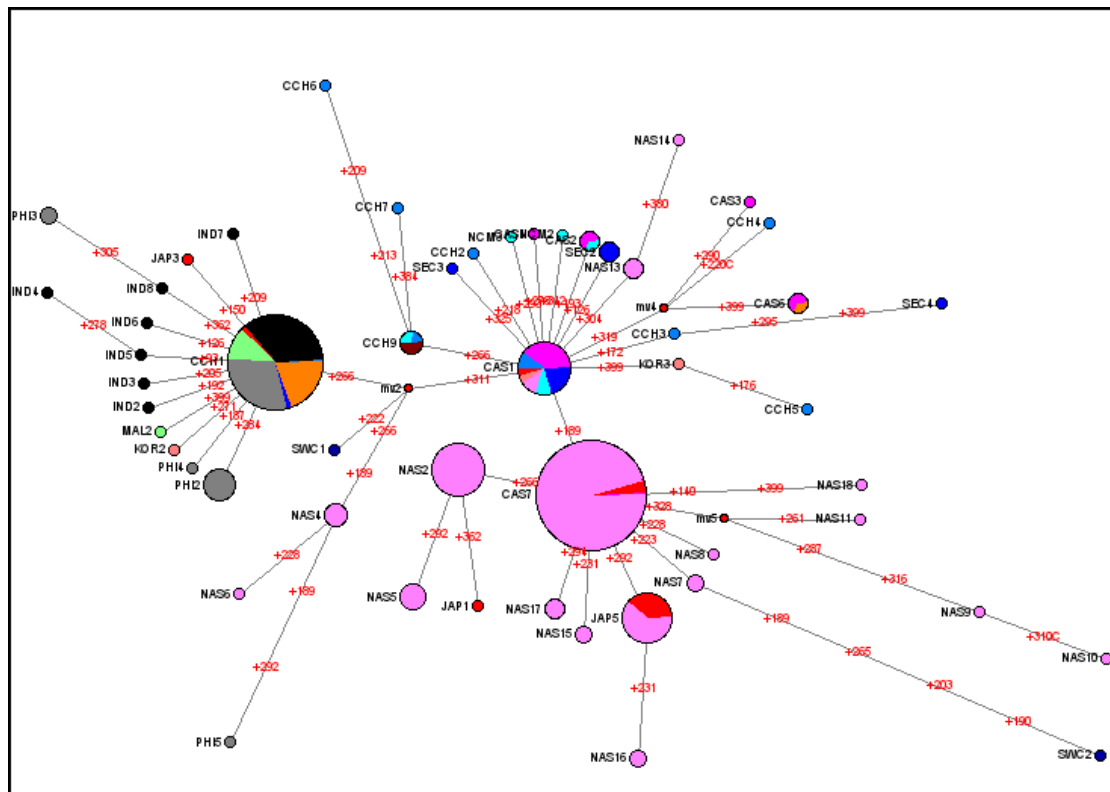


Figura 4.4.7: Caso de teste 2 – Árvore filogenética obtida com o algoritmo de Dijkstra.

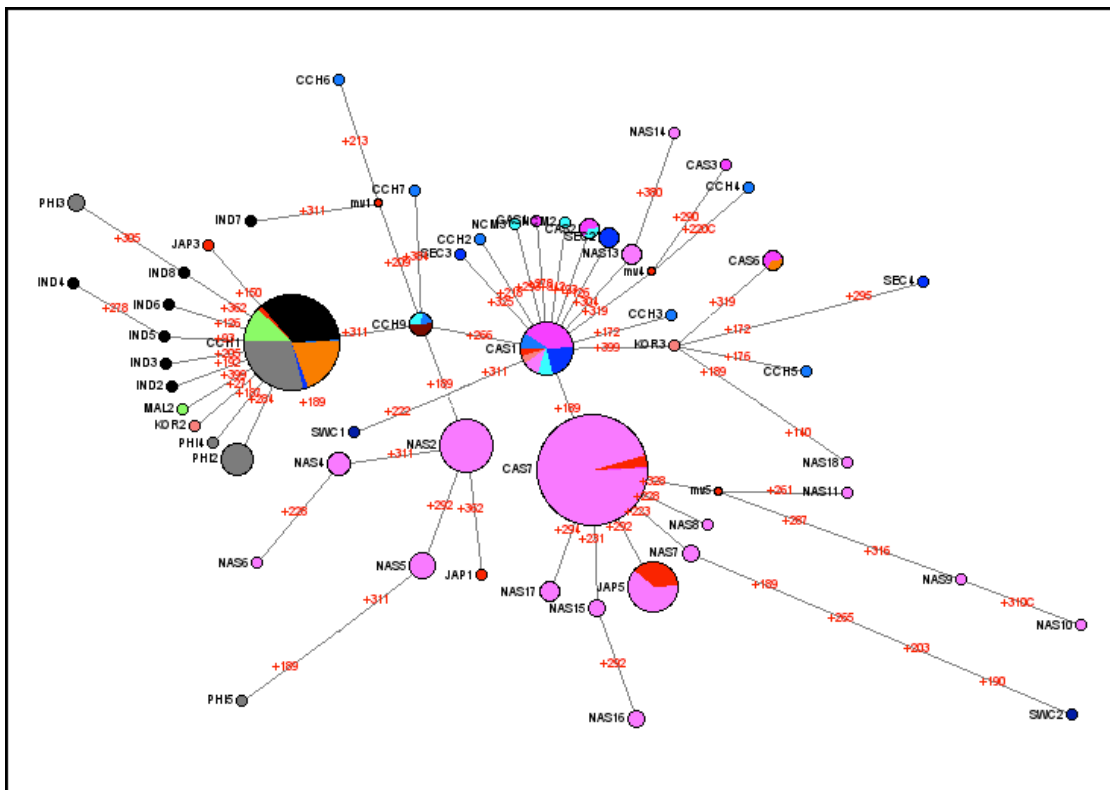


Figura 4.4.8: Caso de teste 2 – Solução proposta pelo especialista da área.

Redes e Árvores Filogenéticas

Após a análise dos resultados obtidos, verifica-se que o algoritmo de Prim resolveu de modo correcto toda a árvore, tal como a solução do especialista. Já o algoritmo de Dijkstra resolveu de forma incorrecta todos os ciclos da rede. Para analisar melhor este caso de teste, apresenta-se de seguida a Tabela 4.4.3.

Tabela 4.4.3: Caso de teste 2 – Resumo dos resultados obtidos com os dois algoritmos.

	Ciclos na rede filogenética	10	Taxa de sucesso
Algoritmo de Dijkstra	Ciclos bem resolvidos	0	0%
	Ciclos mal resolvidos	10	
Algoritmo de Prim	Ciclos bem resolvidos	10	100%
	Ciclos mal resolvidos	0	

Conclui-se que, para este caso, o algoritmo de Prim foi totalmente mais adequado que o de Dijkstra.

4.4.3 Caso de Teste 3

Apresenta-se o seguinte caso de teste com a rede filogenética apresentada na Figura 4.4.16. Consiste numa rede filogenética com 7 ciclos.

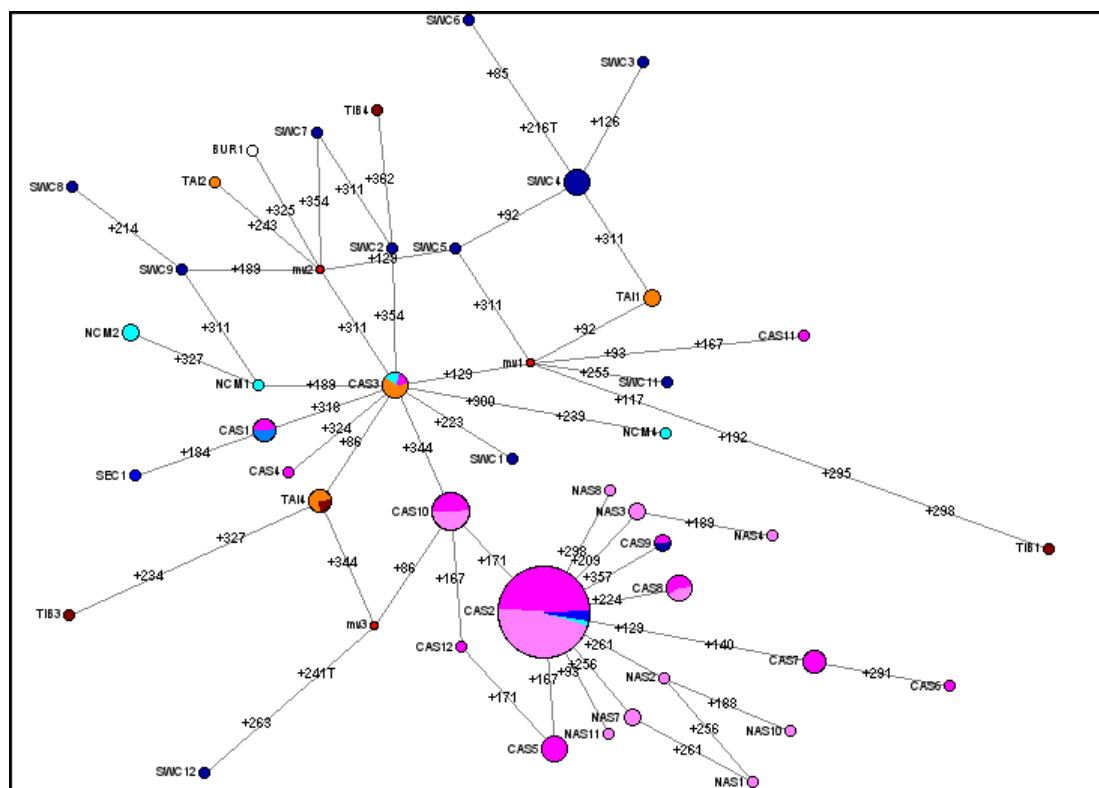


Figura 4.4.9: Caso de teste 3 – Rede filogenética.

Redes e Árvores Filogenéticas

Apresenta-se na Tabela 4.4.4 as características desta rede filogenética.

Tabela 4.4.4: Caso de teste 3 – Características da rede filogenética original.

Caso de Teste 3 – Rede Filogenética Original	
Número total de nós	45
Número de <i>median vectors</i>	3
Frequência total	141
Número de categorias taxonômicas	55
Categorias taxonômicas contidas dentro de outros nós	10
Número total de ciclos (reticulações)	7

Apresenta-se na Figura 4.4.17 o resultado obtido após a aplicação do algoritmo de Prim.

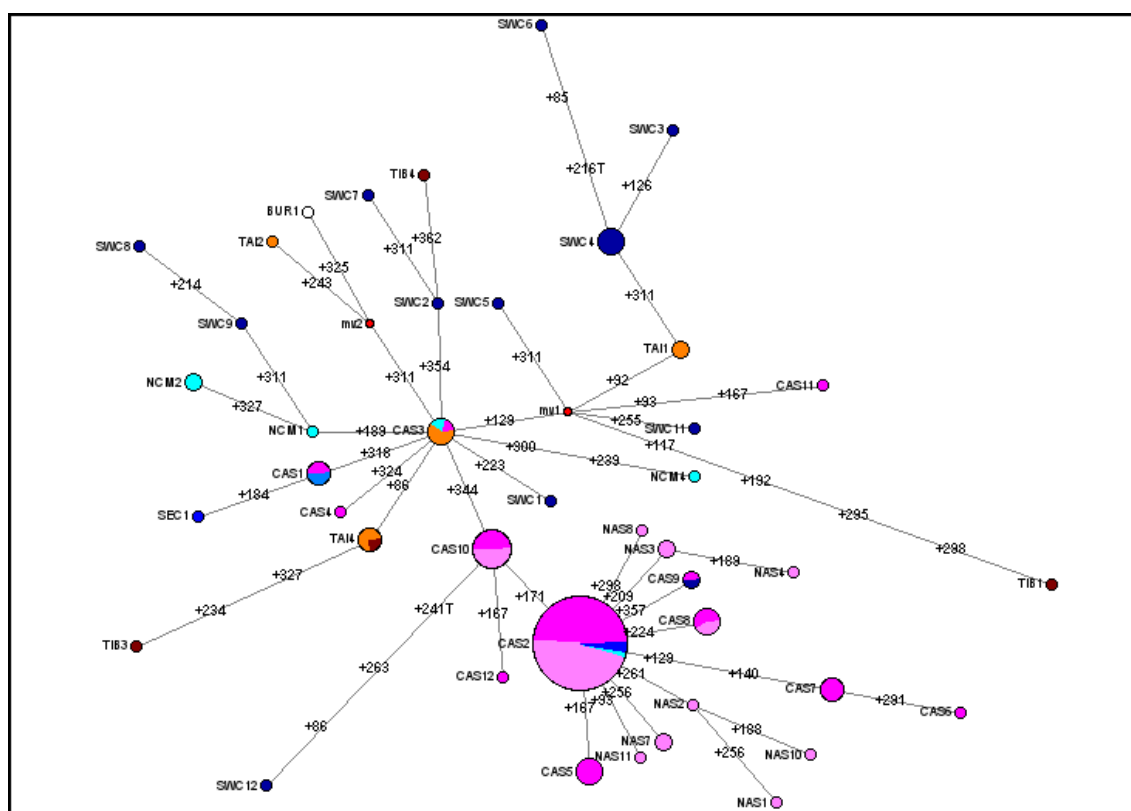


Figura 4.4.10: Caso de teste 3 – Árvore filogenética obtida com o algoritmo de Prim.

De seguida, apresenta-se na Figura 4.4.18, o resultado obtido através do algoritmo de Dijkstra e na Figura 4.4.19, apresenta-se a solução proposta pelo especialista de genética populacional.

Após a análise dos resultados obtidos, verifica-se que o algoritmo de Prim resolveu a maior parte dos ciclos correctamente, de acordo como a solução do especialista. Já o algoritmo de Dijkstra resolveu de forma incorrecta quase todos os ciclos da rede. Tal como no caso de teste anterior, os *median vectors* que ficaram sem razão para permanecerem na árvore, foram eliminados. Para analisar melhor este caso de teste, apresenta-se de seguida a Tabela 4.4.5.

Tabela 4.4.5: Caso de teste 3 – Resumo dos resultados obtidos com os dois algoritmos.

	Ciclos na rede filogenética	7	Taxa de sucesso
Algoritmo de Dijkstra	Ciclos bem resolvidos	1	14,3%
	Ciclos mal resolvidos	6	
Algoritmo de Prim	Ciclos bem resolvidos	5	71,4%
	Ciclos mal resolvidos	2	

Conclui-se que, para este caso, o algoritmo de Prim foi mais adequado que o de Dijkstra.

4.4.4 Caso de Teste 4

Apresenta-se o seguinte caso de teste com a rede filogenética apresentada na Figura 4.4.20. Consiste numa rede filogenética com 6 ciclos.

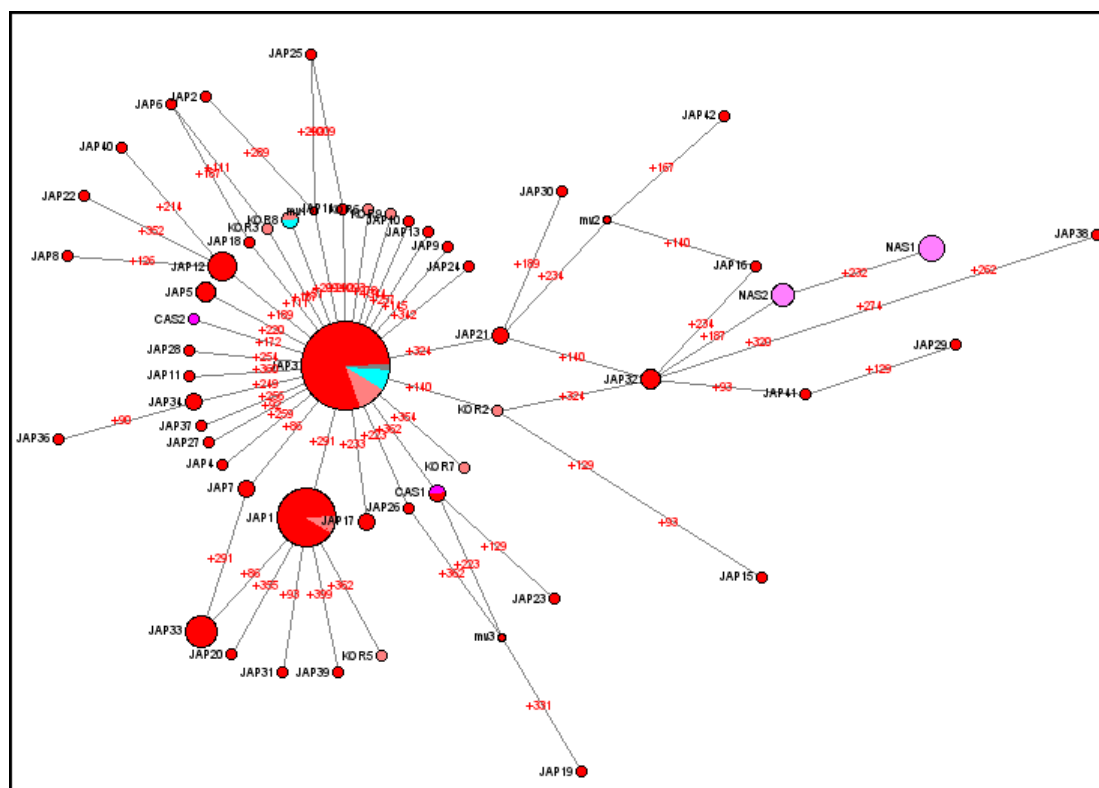


Figura 4.4.13: Caso de teste 4 – Rede filogenética.

Redes e Árvores Filogenéticas

Apresenta-se na Tabela 4.4.6 as características desta rede filogenética.

Tabela 4.4.6: Caso de teste 4 – Características da rede filogenética original.

Caso de Teste 4 – Rede Filogenética Original	
Número total de nós	55
Número de <i>median vectors</i>	3
Frequência total	162
Número de categorias taxonômicas	62
Categorias taxonômicas contidas dentro de outros nós	7
Número total de ciclos (reticulações)	6

Apresenta-se de seguida o resultado obtido através da execução do algoritmo de Prim (Figura 4.4.21).

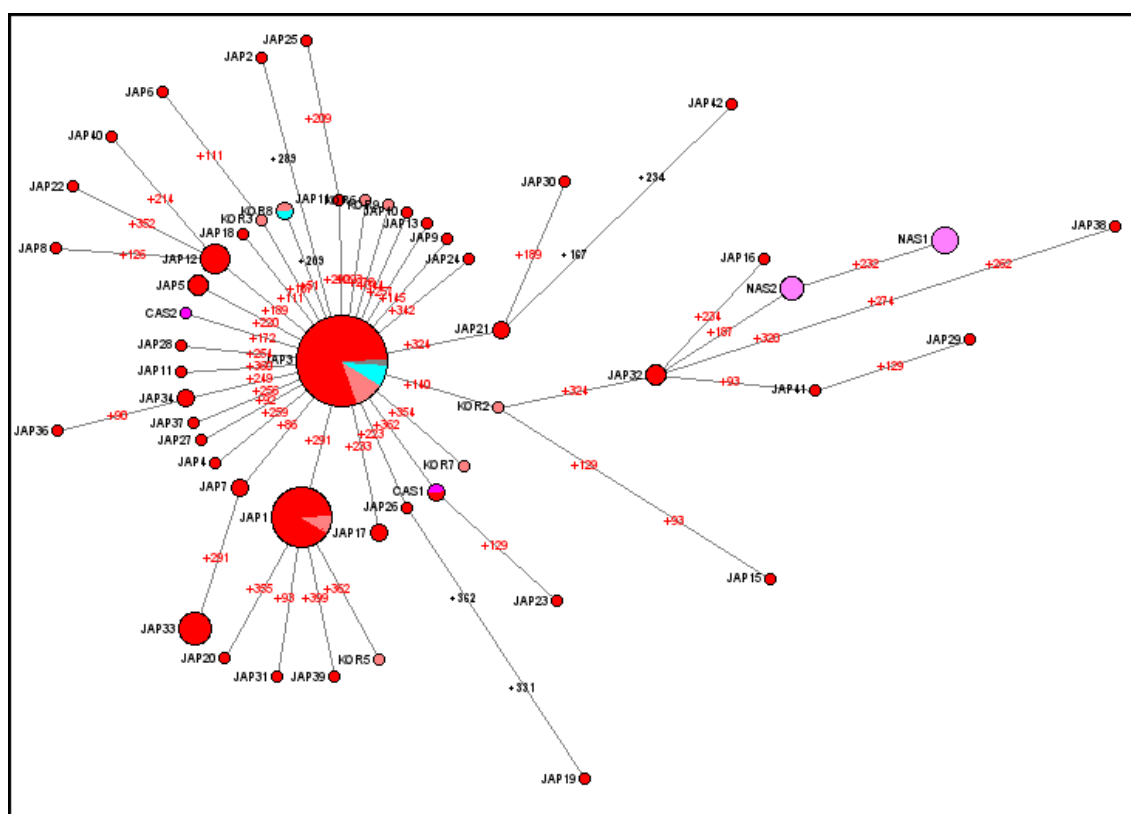


Figura 4.4.14: Caso de teste 4 – Árvore filogenética obtida com o algoritmo de Prim.

De seguida, apresenta-se na Figura 4.4.22, o resultado obtido através do algoritmo de Dijkstra e na Figura 4.4.23, apresenta-se a solução proposta pelo especialista de genética populacional.

Redes e Árvores Filogenéticas

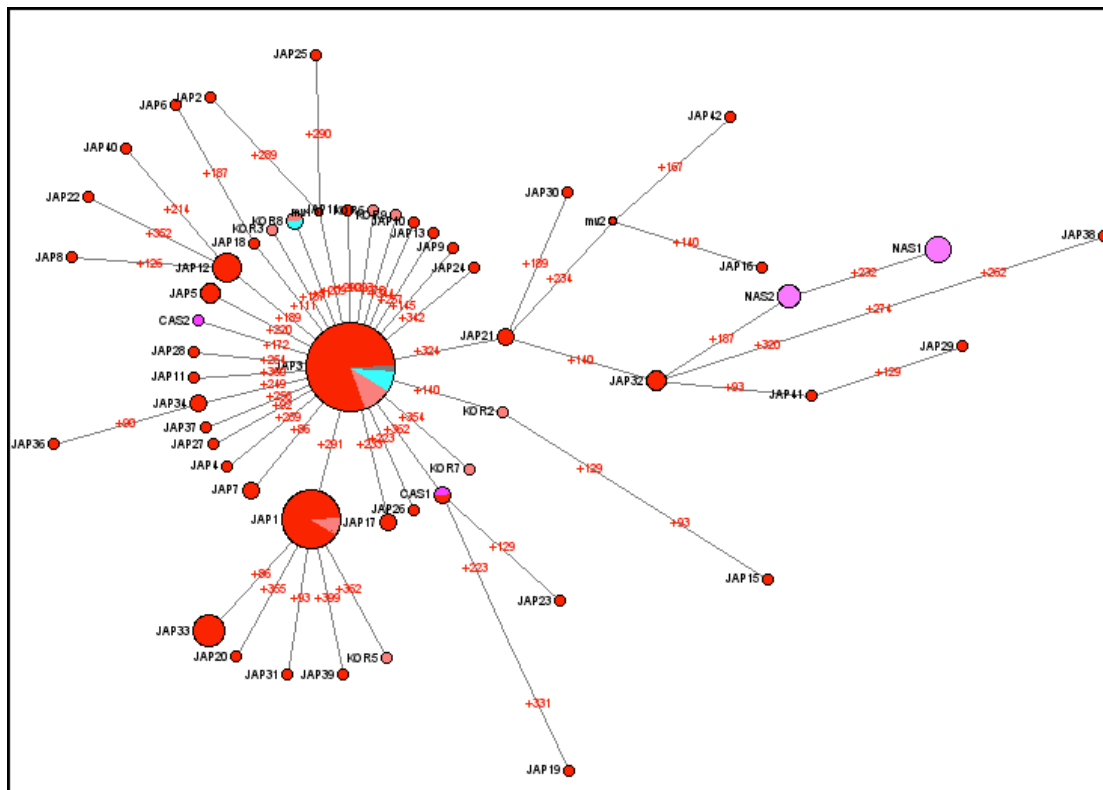


Figura 4.4.15: Caso de teste 4 – Árvore filogenética obtida com o algoritmo de Dijkstra.

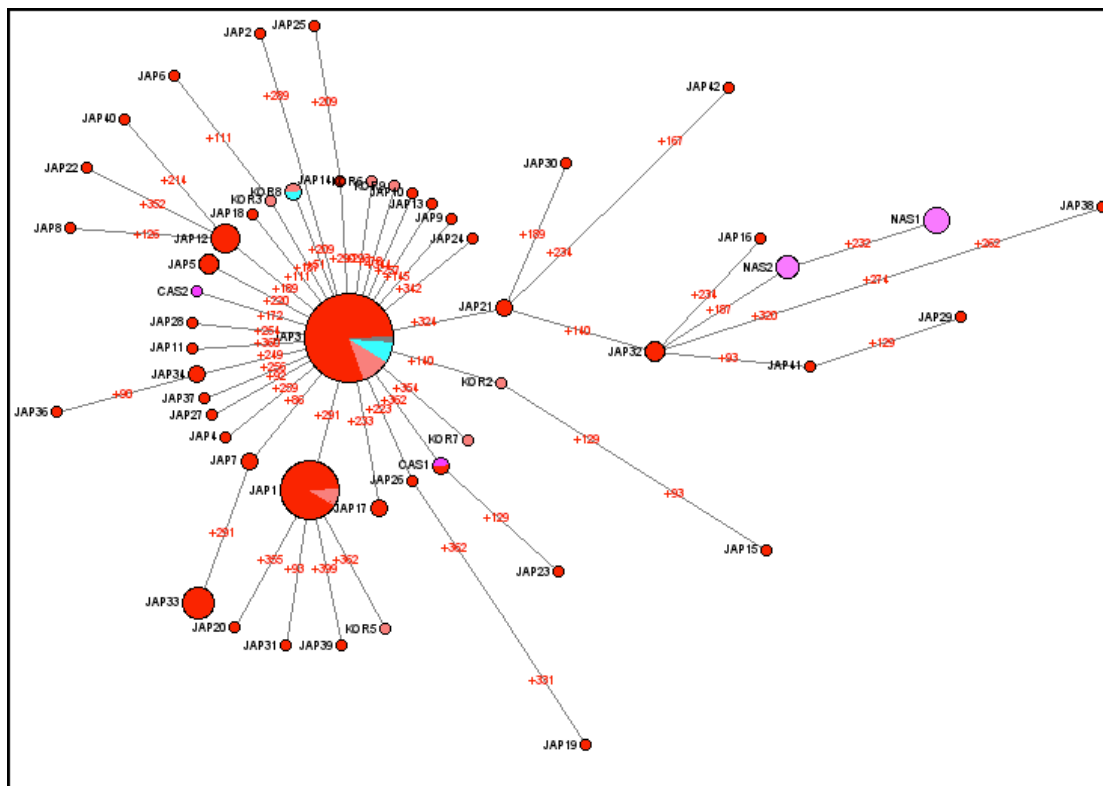


Figura 4.4.16: Caso de teste 4 – Solução proposta pelo especialista da área.

Após a análise dos resultados obtidos, verifica-se que o algoritmo de Prim resolveu a maior parte dos ciclos correctamente, de acordo como a solução do especialista. Já o algoritmo de Dijkstra resolveu de forma incorrecta quase todos os ciclos da rede. Para analisar melhor este caso de teste, apresenta-se de seguida a Tabela 4.4.7.

Tabela 4.4.7: Caso de teste 4 – Resumo dos resultados obtidos com os dois algoritmos.

	Ciclos na rede filogenética	6	Taxa de sucesso
Algoritmo de Dijkstra	Ciclos bem resolvidos	1	16,7%
	Ciclos mal resolvidos	5	
Algoritmo de Prim	Ciclos bem resolvidos	5	83,3%
	Ciclos mal resolvidos	1	

Conclui-se que, para este caso, o algoritmo de Prim foi muito mais adequado que o de Dijkstra.

4.4.5 Caso de Teste 5

De seguida apresenta-se outro caso, com outra rede filogenética que foi submetida à aplicação desenvolvida, representada na Figura 4.4.24. É uma rede com 8 ciclos.

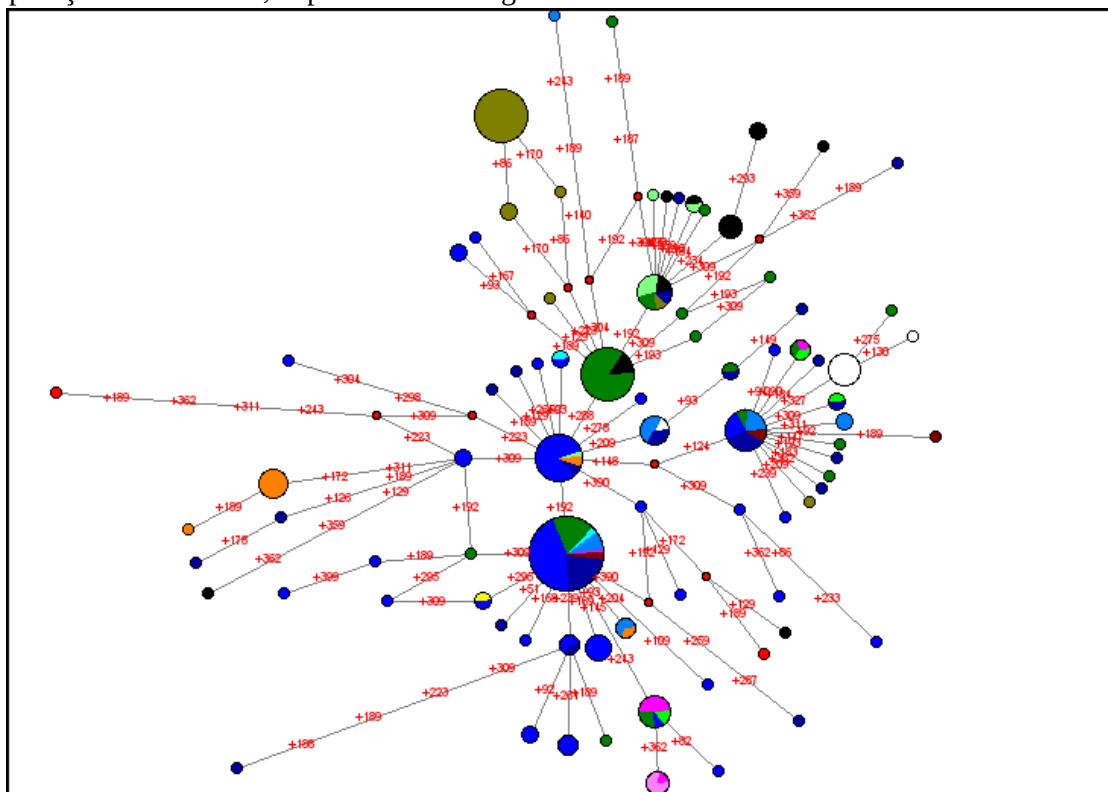


Figura 4.4.17: Caso de teste 5 – Rede filogenética.

Apresenta-se na Tabela 4.4.8, as características desta rede filogenética. Por uma questão de apresentar a rede filogenética e suas soluções de forma clara, ocultaram-se os nomes de cada nó.

Tabela 4.4.8: Caso de teste 5 – Características da rede filogenética original.

Caso de Teste 5 – Rede Filogenética Original	
Número total de nós	92
Número de <i>median vectors</i>	10
Frequência total	243
Número de categorias taxonômicas	125
Categorias taxonômicas contidas dentro de outros nós	33
Número total de ciclos (reticulações)	8

Apresenta-se na Figura 4.4.25 o resultado obtido através da aplicação do algoritmo de Prim.

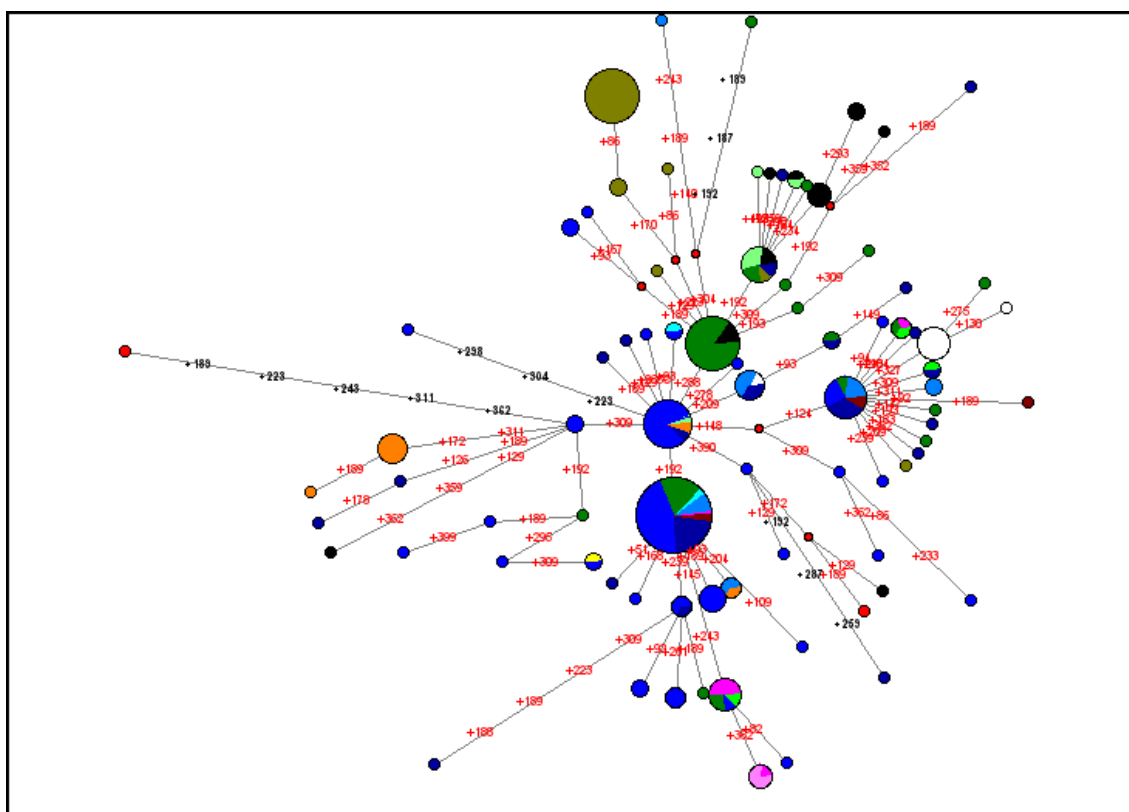


Figura 4.4.18: Caso de teste 5 – Árvore filogenética obtida com o algoritmo de Prim.

Através da aplicação do algoritmo de Prim, verifica-se que são eliminados todos os ciclos anteriormente existentes na rede e que as arestas que se mantêm na árvore resultante são as que têm menor custo (mutações mais frequentes).

Redes e Árvores Filogenéticas

De seguida, apresenta-se na Figura 4.4.26, o resultado obtido através do algoritmo de Dijkstra. Como este algoritmo não garante que todas as arestas da árvore sejam as de menor custo, apenas garante o caminho mais curto de qualquer nó até à raiz, prevê-se um resultado diferente.

Na Figura 4.4.27, apresenta-se a solução proposta pelo especialista de genética populacional.

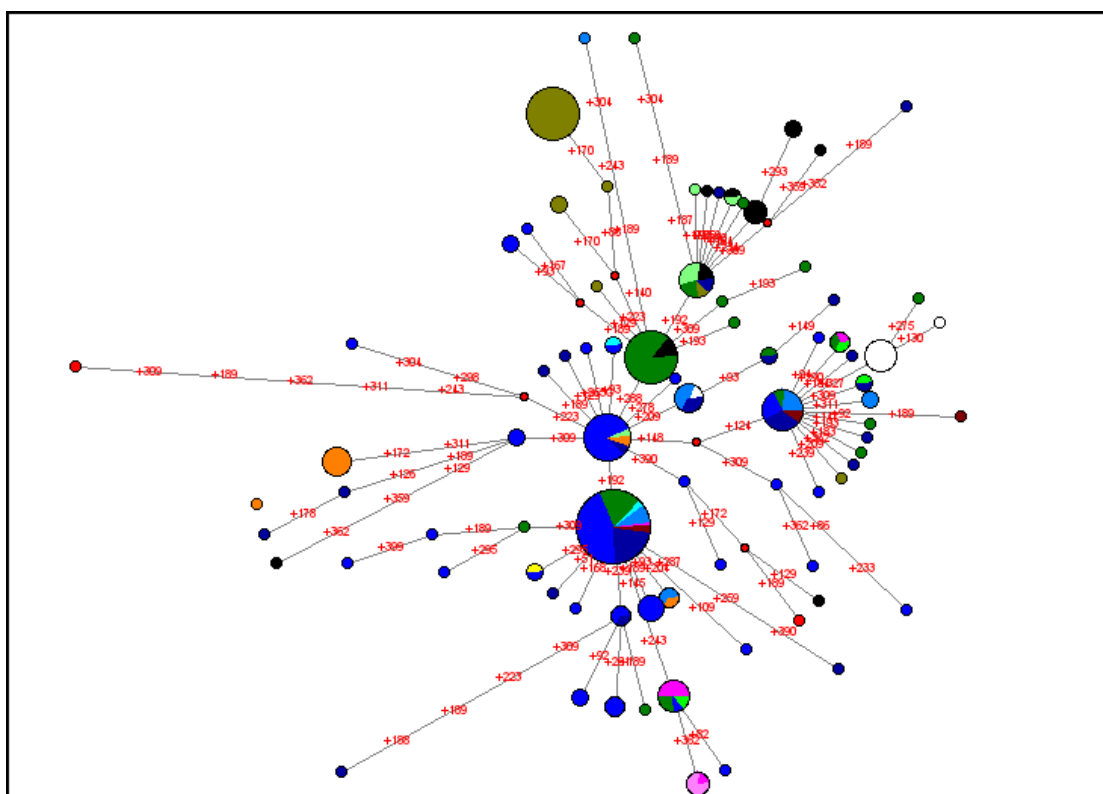


Figura 4.4.19: Caso de teste 5 – Árvore filogenética obtida com o algoritmo de Dijkstra.

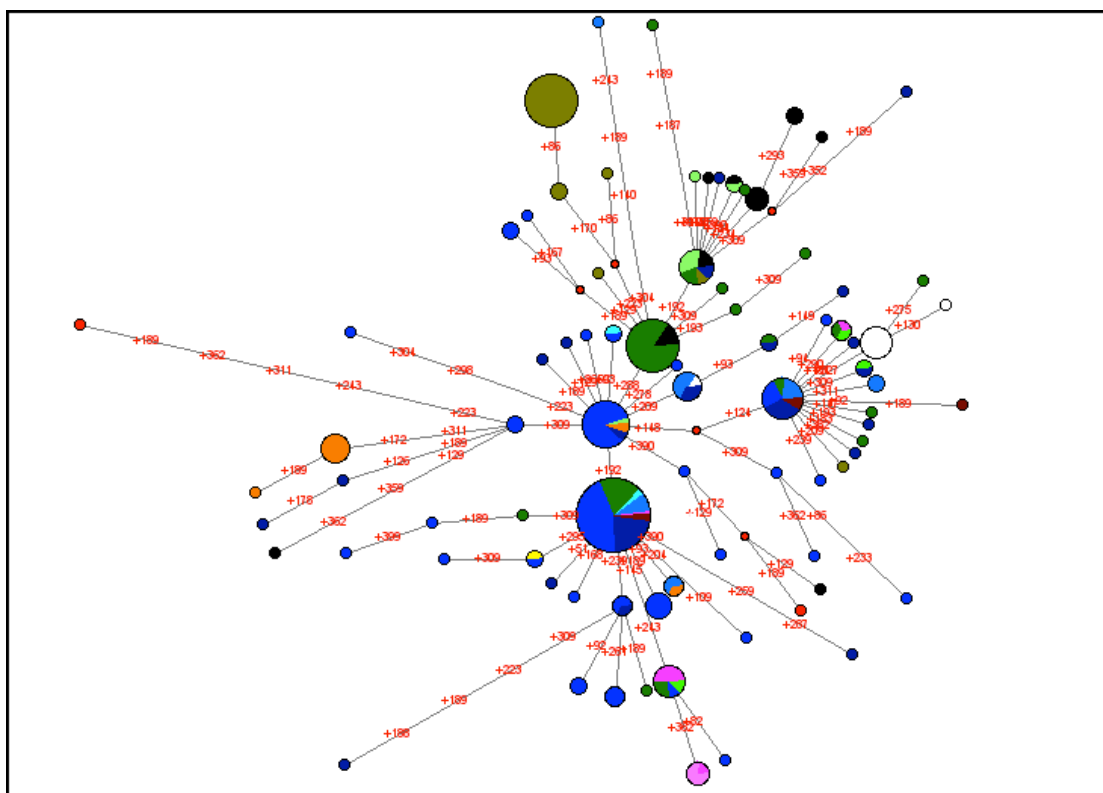


Figura 4.4.20: Caso de teste 5 – Solução proposta pelo especialista da área.

Após a análise dos resultados obtidos, verifica-se que para alguns ciclos, o algoritmo de Prim resolve do modo correcto, tal como a solução do especialista. No entanto também há outros casos em que o algoritmo de Dijkstra é o mais adequado. Para analisar melhor este caso de teste, apresenta-se de seguida a Tabela 4.4.9.

Tabela 4.4.9: Caso de teste 5 – Resumo dos resultados obtidos com os dois algoritmos.

	Ciclos na rede filogenética	8	Taxa de sucesso
Algoritmo de Dijkstra	Ciclos bem resolvidos	4	50%
	Ciclos mal resolvidos	4	
Algoritmo de Prim	Ciclos bem resolvidos	3	37,5%
	Ciclos mal resolvidos	5	

Conclui-se que, para este caso, o algoritmo de Dijkstra foi ligeiramente mais adequado.

4.5 Resumo

O algoritmo implementado constrói árvores filogenéticas a partir do resultado do Network e aceita os 2 formatos (fdi e out). As árvores geradas são próximas das árvores idealizadas pelos

Redes e Árvores Filogenéticas

especialistas pelo que os casos de estudo obtiveram uma aceitação positiva por parte dos especialistas da área. Para melhorar esta componente do trabalho, estamos ainda a afirmar as regras de ranking das árvores, de modo a que a árvore sugerida não necessite de ajustes ou necessite de um número mínimo de ajustes. Na Tabela 4.5.1 apresenta-se as características das diferentes redes filogenéticas utilizadas em cada caso de teste.

Tabela 4.5.1: Características da rede filogenética de cada caso de tese.

Caso de teste	N.º de nós	N.º de arestas	N.º de <i>median vectors</i>	N.º de reticulações
1	30	35	2	6
2	61	71	8	10
3	45	51	3	7
4	55	60	3	6
5	92	99	10	8

Apresenta-se na Tabela 4.5.2, as conclusões dos casos de teste, excepto do primeiro, uma vez que este só utilizou o algoritmo de Prim e com várias árvores como solução.

Tabela 4.5.2: Resumo dos resultados gerais.

Caso	Solução	N.º de Nós	N.º de Arestas	Ciclos bem resolvidos	Ciclos mal resolvidos	Taxa de sucesso
2	Dijkstra	56	55	0	10	0%
	Prim	56	55	10	0	100%
	Especialista	56	55			
3	Dijkstra	44	43	1	6	14,3%
	Prim	44	43	5	2	71,4%
	Especialista	44	43			
4	Dijkstra	54	53	1	5	16,7%
	Prim	52	51	5	1	83,3%
	Especialista	52	51			
5	Dijkstra	88	87	4	4	50%
	Prim	88	87	3	5	37,5%
	Especialista	87	86			

Redes e Árvores Filogenéticas

Em cada um dos casos verifica-se que na árvore obtida houve uma redução do número de nós em relação à rede filogenética original. Isto deveu-se ao facto de existirem *median vectors* que foram eliminados, tal como abordado nos casos de estudo, ou seja, um *median vector* terá de possuir pelo menos 3 arestas adjacentes, senão não tem razão, para existir, pois é uma sequência hipotética. Assim, no caso de serem duas as arestas adjacentes, dever-se-á unir as duas arestas correspondentes e eliminar o *median vector* da árvore e no caso de ser apenas uma aresta, esta deverá ser eliminada da árvore, bem como o *median vector*.

Conclui-se que o algoritmo de Prim é o mais adequado, pois raramente tem mais erros que o algoritmo de Dijkstra e normalmente possui uma taxa de sucesso muito mais elevada, sendo que há casos onde a taxa de sucesso do algoritmo de Dijkstra é muito baixa, existindo mesmo casos, como o caso de estudo 2, onde não tomou a decisão certa para qualquer ciclo.

Para estes 5 casos de estudo, executou-se a aplicação com ficheiros homónimos no formato out em vez de fdi, como aqui representado, e o resultado foi igual, já que as redes filogenéticas são exactamente as mesmas.

Um ponto extremamente importante é que as árvores são geradas automaticamente a partir do resultado do Network (o que demora alguns segundos). Actualmente o especialista constrói as árvores manualmente no Microsoft Excel. O passo que está a ser desenvolvido consiste em produzir automaticamente as árvores e permitir ao especialista fazer pequenas alterações de correcção de modo a obter a árvore pretendida.

Capítulo 5

Estatísticas

A segunda parte deste projecto é a automatização do cálculo de estatísticas nas árvores filogenéticas. Assim, foi desenvolvido um módulo na aplicação desenvolvida que serve para calcular, de forma automática, todas estas estatísticas.

5.1 Implementação das Estatísticas

O objectivo final deste projecto consistia em automatizar o cálculo das estatísticas, que anteriormente era realizado pelos profissionais da área de forma manual, através da observação das árvores filogenéticas. Pretendia-se ainda melhorar o cálculo das estatísticas, através da substituição da Equação (3.5.2) pela (3.5.3), apresentadas na Secção 3.5.

Assim, foi desenvolvido um módulo na aplicação, que consiste no seguinte fluxo de trabalho:

- Através da análise de uma árvore filogenética, calcular quais as sequências (nós) que representam fundadores.
- Para cada um dos nós fundadores, calcular os valores de n e m , que permitirão obter para cada sequência o valor de ρ e o desvio padrão correspondente.
- Exportar os dados obtidos para um ficheiro do tipo “formatted text” (separação por tabulações), legível em Microsoft Excel, para os profissionais da área poderem utilizar os dados obtidos.

A aplicação lê um ficheiro, especificado pelo utilizador, que pode ser do formato fdi ou out, que contenha uma árvore filogenética.

Após a leitura do ficheiro especificado pelo utilizador, os dados da árvore filogenética são guardados em várias estruturas de dados semelhantes às utilizadas para guardar as redes filogenéticas.

O utilizador terá ainda de especificar alguns dados necessários para calcular os fundadores: a raiz da árvore, o critério utilizado para calcular os fundadores (f_1 , f_2 ou f_s , anteriormente referidos na Secção 3.5) e especificar, para cada região geográfica, se consiste numa região ancestral, descendente ou indeterminada.

De seguida são calculados os fundadores da árvore, com base nos dados especificados pelo utilizador. Para cada um dos fundadores são calculados os valores de frequência absoluta (soma da sua frequência mais as frequências de todas as outras sequências nos ramos derivados) e a frequência de descendentes (soma da sua frequência descendente mais as frequências de todas as outras sequências descendentes nos ramos derivados). Estas frequências correspondem ao n nas Equações (3.5.1), (3.5.2) e (3.5.3), apresentadas na Secção 3.5.

Após serem calculadas as frequências, são calculados os valores das mutações para cada fundador (m na Equação (3.5.1)). Este valor consiste em, para cada fundador, multiplicar o valor de cada uma das frequências das sequências dos ramos derivados pelo número de mutações existentes no ramo derivado correspondente.

Com base nos valores das frequências obtidas (n) e no valor das mutações (m), é possível calcular o valor de ρ , bem como o valor do desvio padrão. De seguida, para todos os fundadores existentes, os valores de n , ρ e desvio padrão são escritos num ficheiro legível em Excel, para os profissionais da área poderem por fim utilizar para estimar idades com base num relógio molecular que já se encontra implementado. No ficheiro criado estarão os valores para apenas as sequências descendentes (utilizando as frequências relativas) e os valores para todas as sequências (utilizando as frequências absolutas).

Na seguinte secção são apresentados os resultados experimentais obtidos através da execução deste módulo da aplicação desenvolvida.

5.2 Resultados Experimentais

Para testar o correcto funcionamento da aplicação, foram realizados alguns casos de teste. Como a forma para determinar as estatísticas possui um vasto leque de propriedades e restrições, estes casos de teste foram determinantes para conseguir implementar o cálculo de estatísticas de forma correcta, uma vez que à medida que se foram executando estes casos detectaram-se várias lacunas e erros da aplicação, que ao longo do tempo se foram corrigindo até se chegar ao ponto de não ocorrer qualquer erro.

Os casos de teste para este módulo da aplicação, consistem em árvores filogenéticas (algumas obtidas pela aplicação do algoritmo de Prim desenvolvido), para as quais se determinam todos os dados estatísticos necessários.

Todos estes casos de teste foram também resolvidos manualmente, ou seja, da forma que tem sido resolvida até actualmente, por um especialista de genética populacional. O número de erros, por ele cometidos, são apresentados em cada um dos casos, demonstrando o quão complexo é este processo para resolver manualmente e a grande probabilidade de serem cometidos erros. Estes casos de teste foram executados, recorrendo aos critérios f1, f2 e fs (apresentados na Secção 3.5), no entanto os resultados para o critério fs não foram apresentados devido ao facto de serem muito semelhantes com os de f1.

5.2.1 Caso de Teste 1

Apresenta-se de seguida na Figura 5.2.1 e na Figura 5.2.2 o primeiro caso de teste.

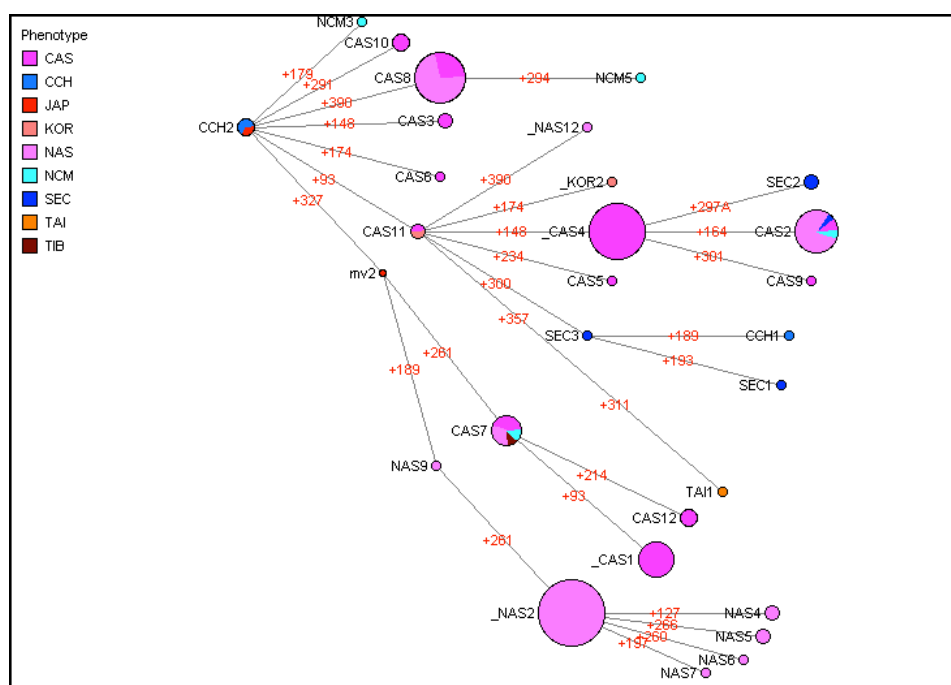


Figura 5.2.1: Caso de teste 1 – Árvore filogenética.

Apresenta-se na Tabela 5.2.1 as características desta árvore.

Tabela 5.2.1: Caso de teste 1 – Características da árvore filogenética.

Caso de Teste 1 – Árvore Filogenética	
Número total de nós	29
Número de <i>median vectors</i>	1
Frequência total	168
Número de categorias taxonómicas	42
Categorias taxonómicas contidas dentro de outros nós	13
Número de erros cometidos pelo especialista	1

Esta é uma árvore filogenética simples, cuja raiz é o nó CCH2 e foi submetida ao módulo de estatísticas com os parâmetros de entrada apresentados na Figura 5.2.2. Estes dados significam que a população de origem ou ancestral será a das regiões CAS, TAI e KOR, logo todas as sequências com estas iniciais no nome são ancestrais. As restantes regiões são de destino ou descendentes, à excepção da região TIB, para a qual não há certeza do seu tipo, sendo classificada como indeterminada, não entrando nos critérios de selecção dos fundadores, mas sendo incluída no cálculo de frequências absolutas.

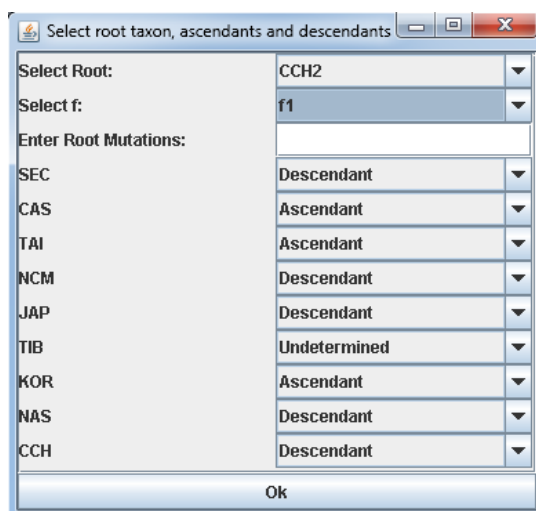


Figura 5.2.2: Caso de teste 1 – Parâmetros de entrada para f1.

O resultado obtido, com estes parâmetros de entrada, é apresentado na Figura 5.2.3. Na coluna A, está presente, na linha 1, o critério que foi utilizado (f1 ou f2) e nas linhas seguintes estão os fundadores detectados, identificados pelo número das mutações existentes desde a raiz da árvore até ao fundador. Os valores que estão nas colunas B, C e D são os correspondentes à frequência de descendentes (soma das frequência dos descendentes) e nas colunas E, F e G são os valores correspondentes à frequência absoluta.

Estatísticas

	A	B	C	D	E	F	G
1	f1						
2		n	rho	se	n	rho	se
3	+261 +327	4	0.0	0.0	24	0.625	0.5153882032022076
4	+148 +93	19	1.0	0.9009075141380889	52	0.4230769230769231	0.3679062782636344
5	+327	48	1.1041666666666667	0.9813804761094898	72	1.9444444444444444	1.0065372739996354
6	+93	4	1.5	0.8660254037844386	61	1.3770491803278688	0.9106848794971245
7		22	0.9090909090909091	0.8207031856939904	168	2.3214285714285716	0.7960182942097206

Figura 5.2.3: Caso de teste 1 – Ficheiro Excel resultante para f1.

O n consiste no valor da frequência de sequências da população de destino no respectivo fundador, isto no caso das colunas B, C e D. Nas colunas E, F e G n é o valor da frequência absoluta de todas as populações no respectivo fundador. Para cada fundador, com base no valor de n e do número de mutações dos ramos adjacentes, é ainda calculado o valor de ρ (rho) e o desvio padrão ou *standard error* (se).

Este resultado traduz que os fundadores desta árvore são os assinalados na Figura 5.2.4.

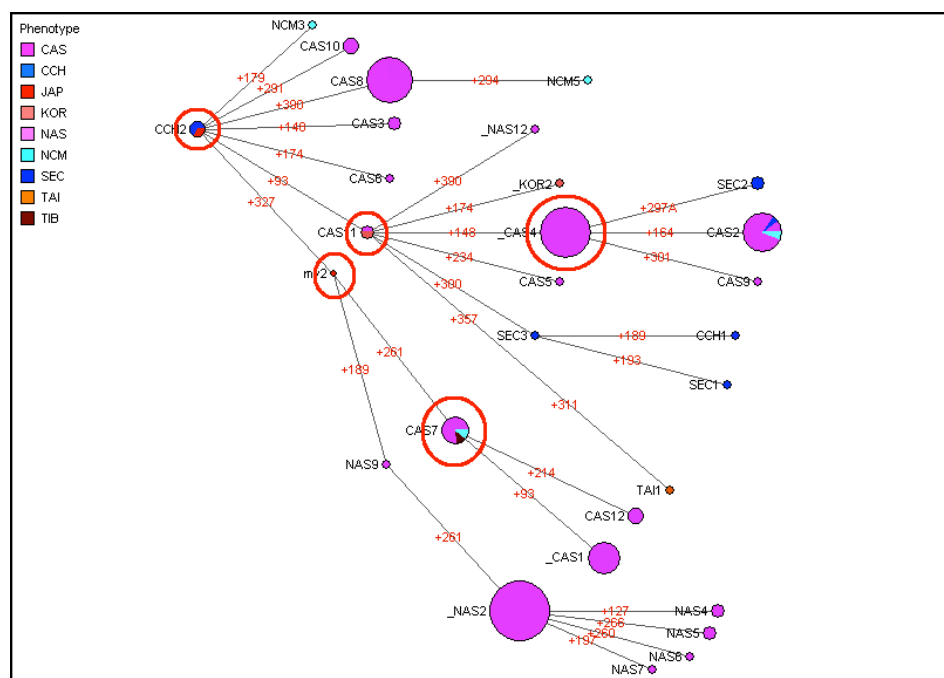


Figura 5.2.4: Caso de teste 1 – Árvore filogenética com fundadores para f1.

De seguida, submeteu-se a mesma árvore ao critério f2, como apresentado na Figura 5.2.5.

Estatísticas

Figura 5.2.5: Caso de teste 1 – Parâmetros de entrada para f2.

Para este caso, o resultado obtido é o apresentado na Figura 5.2.6. Na Figura 5.2.7 vê-se a tradução destes dados obtidos, em relação aos fundadores encontrados.

	A	B	C	D	E	F	G
1	f2						
2		n	rho	se	n	rho	se
3	+261 +327	4	0.0	0.0	24	0.625	0.5153882032022076
4	+148 +93	19	1.0	0.9009075141380889	52	0.4230769230769231	0.3679062782636344
5	+93	4	1.5	0.8660254037844386	61	1.3770491803278688	0.9106848794971245
6		70	2.414285714285714	1.2082201448276497	168	2.3214285714285716	0.7960182942097206

Figura 5.2.6: Caso de teste 1 – Ficheiro Excel resultante para f2.

Estatísticas

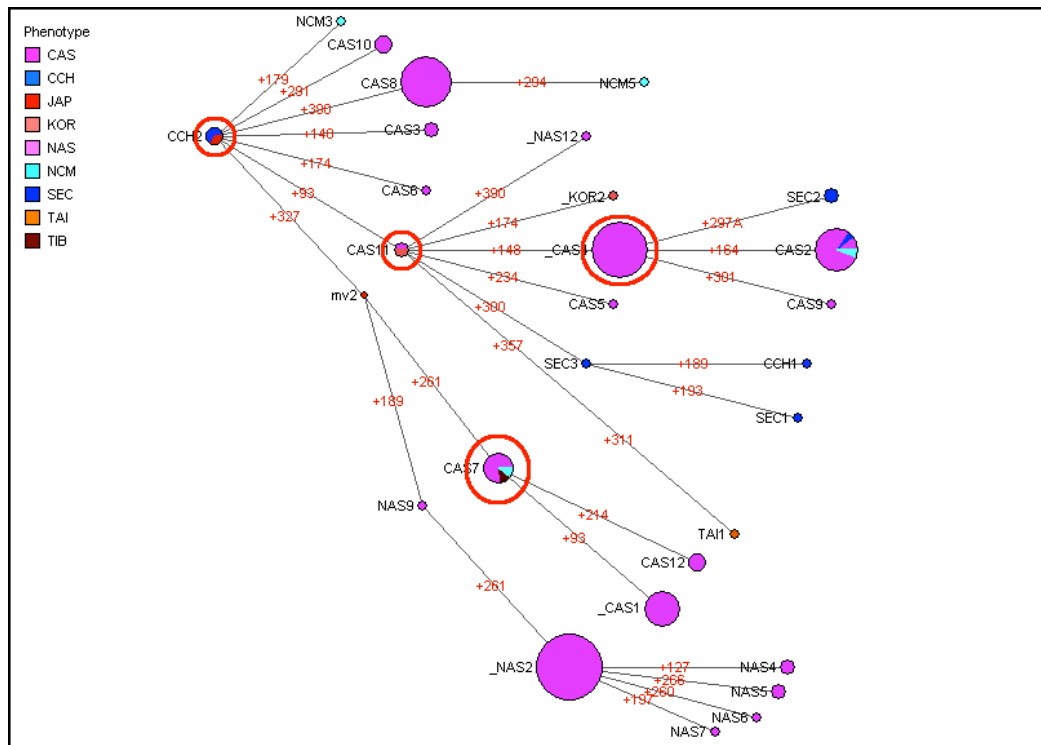


Figura 5.2.7: Caso de teste 1 – Árvore filogenética com fundadores para f2.

Foi obtido menos um fundador, o mv2, uma vez que este só tem um ramo derivado na população de origem, que era suficiente para f1, mas para f2 precisava de ter pelo menos dois. Conclui-se assim que o critério f2 é mais restritivo que o f1, uma vez que são normalmente obtidos menos fundadores.

5.2.2 Caso de Teste 2

O caso de teste 2 utiliza outra árvore filogenética, apresentada na Figura 5.2.8. É uma árvore mais complexa que a do caso de teste 1, uma vez que possui mais sequências e mais alguns casos particulares para o cálculo dos fundadores, como por exemplo a existência de sequências derivadas da população de origem que apesar de não estarem nos nós vizinhos do potencial fundador, existem no caminho evolutivo, sendo necessário guardar informação sobre a existência destas sequências ao longo dos caminhos que as possuem. A raiz da árvore é o nó CAS3.

Estatísticas

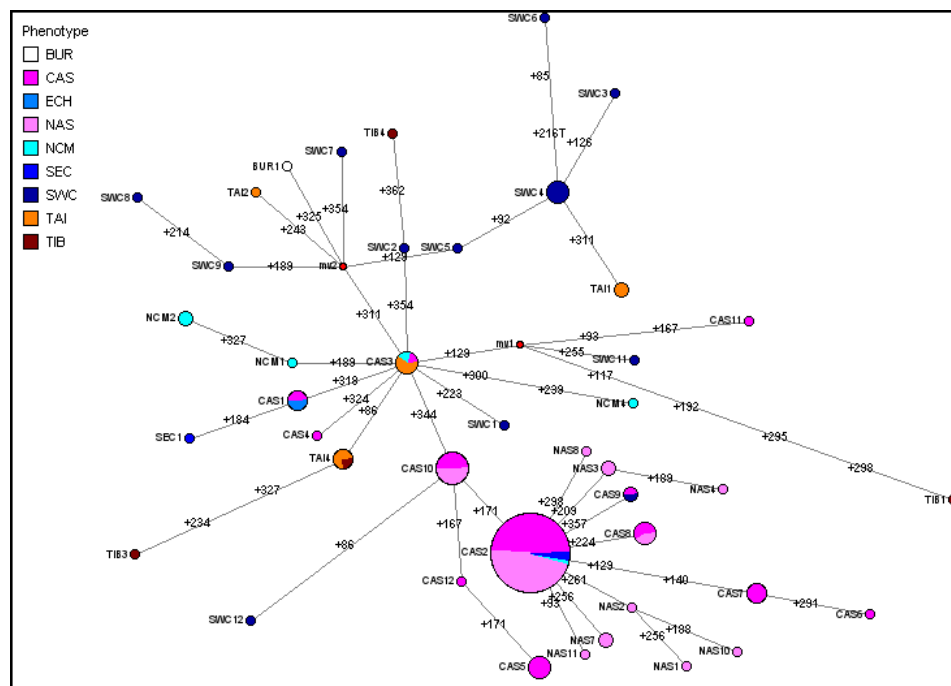


Figura 5.2.8: Caso de teste 2 – Árvore filogenética.

Apresenta-se na Tabela 5.2.2 as características desta árvore.

Tabela 5.2.2: Caso de teste 2 – Características da árvore filogenética.

Caso de Teste 2 – Árvore Filogenética	
Número total de nós	44
Número de <i>median vectors</i>	2
Frequência total	141
Número de categorias taxonómicas	54
Categorias taxonómicas contidas dentro de outros nós	10
Número de erros cometidos pelo especialista	1

Apresenta-se na Figura 5.2.9 os parâmetros de entrada para o cálculo das estatísticas desta árvore. Definiu-se como população descendente, as populações das regiões CAS, TAI e NAS, as populações das restantes regiões como ancestrais, à excepção de BUR que é indeterminada.

Estatísticas

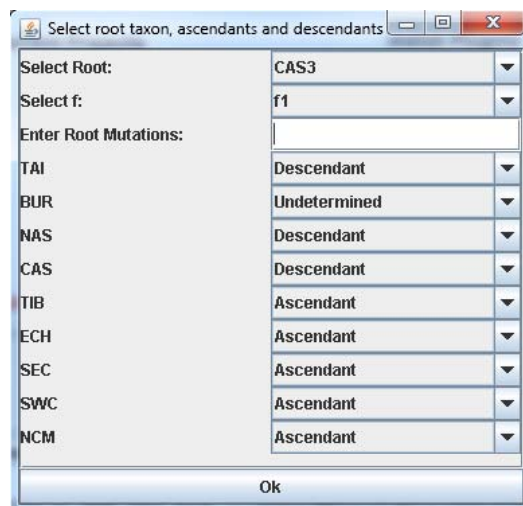


Figura 5.2.9: Caso de teste 2 – Parâmetros de entrada para f1.

O ficheiro em Excel obtido para f1 é o apresentado na Figura 5.2.10.

	A	B	C	D	E	F	G
1	f1						
2		n	rho	se	n	rho	se
3	+92 +129 +311	2	0.0	0.0	9	0.5555555555555556	0.2939723678960656
4	+171 +344	79	0.379746835443038	0.1290891016099439	83	0.37349397590361444	0.12462747509383856
5	+311	1	0.0	0.0	15	2.0	0.9333333333333333
6	+129	1	0.0	0.0	3	2.3333333333333335	0.8819171036881969
7	+86	3	0.0	0.0	5	0.4	0.282842712474619
8	+344	16	0.6875	0.4881406047441659	100	1.28	0.8402380615040002
9	+318	2	0.0	0.0	5	0.2	0.2
10		5	0.2	0.2	141	2.1843971631205674	0.9401377350244273

Figura 5.2.10: Caso de teste 2 – Ficheiro Excel resultante para f1.

Traduzindo este ficheiro em termos de fundadores encontrados, temos na Figura 5.2.11 os 8 fundadores encontrados na árvore.

Estatísticas

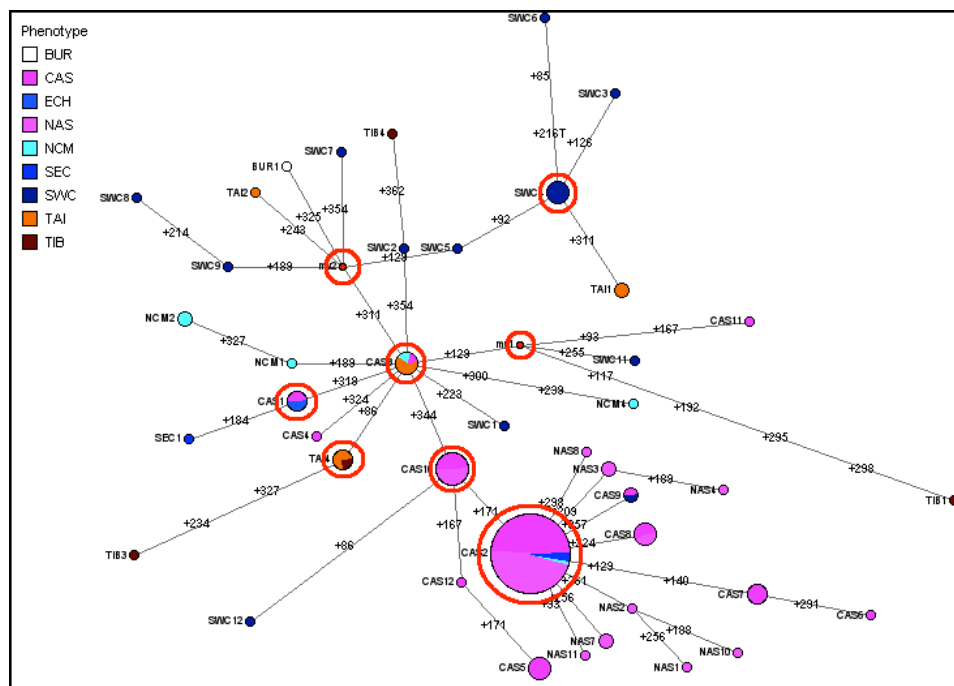


Figura 5.2.11: Caso de teste 2 – Árvore filogenética com fundadores para f1.

De seguida, submeteu-se a mesma árvore com os mesmos parâmetros, excepto o critério f2, em vez de f1, como apresentado na Figura 5.2.12.

Figura 5.2.12: Caso de teste 2 – Parâmetros de entrada para f2.

O ficheiro em Excel obtido para f2 é o apresentado na Figura 5.2.13.

Estatísticas

	A	B	C	D	E	F	G
1	f2						
2		n	rho	se	n	rho	se
3	+92 +129 +311	2	0.0	0.0	9	0.555555555555556	0.2939723678960656
4	+311	1	0.0	0.0	15	2.0	0.933333333333333
5	+129	1	0.0	0.0	3	2.33333333333333	0.8819171036881969
6	+344	95	1.263157894736842	0.8424999075268943	100	1.28	0.8402380615040002
7		10	0.6	0.37416573867739417	141	2.1843971631205674	0.9401377350244273

Figura 5.2.13: Caso de teste 2 – Ficheiro Excel resultante para f2.

Traduzindo este ficheiro Excel em termos de fundadores encontrados, temos na Figura 5.2.14 os 5 fundadores encontrados na árvore utilizando o critério f2. Assim, comparando com f1, vemos que se obtiveram menos 3 fundadores: CAS1, CAS2 e TAI4, uma vez que estes só têm um ramo derivado na população de origem e precisavam de ter pelo menos dois, para poderem continuar a ser fundador com f2.

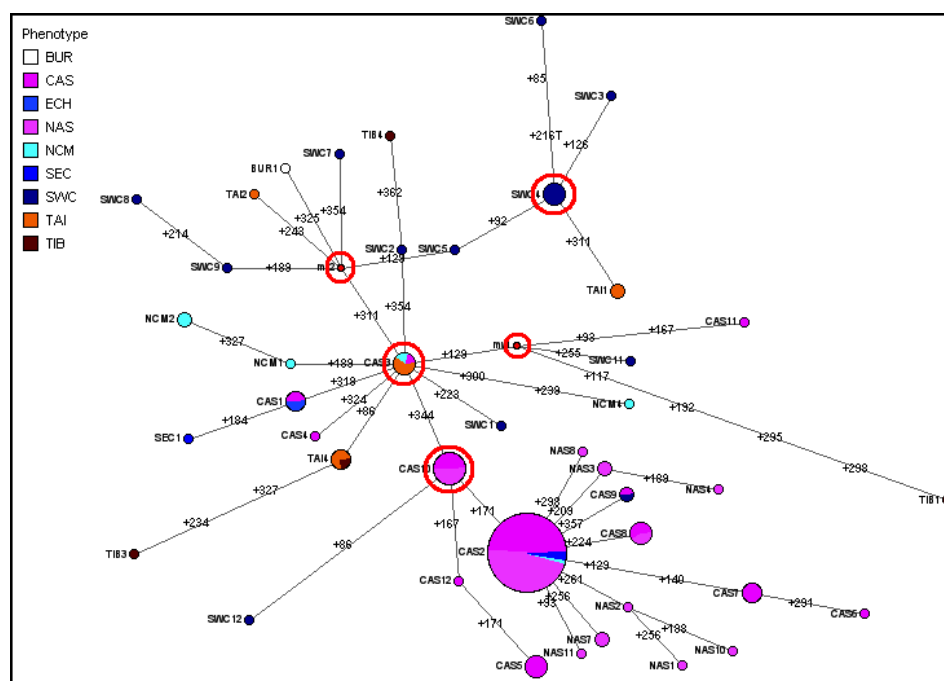


Figura 5.2.14: Caso de teste 2 – Árvore filogenética com fundadores para f2.

5.2.3 Caso de Teste 3

O caso de teste 3 utiliza outra árvore filogenética, apresentada na Figura 5.2.15. É uma árvore mais complexa que a dos anteriores casos de estudo, uma vez que possui mais sequências e mais alguns casos particulares para o cálculo dos fundadores. A raiz da árvore é o nó MAL3.

Estatísticas

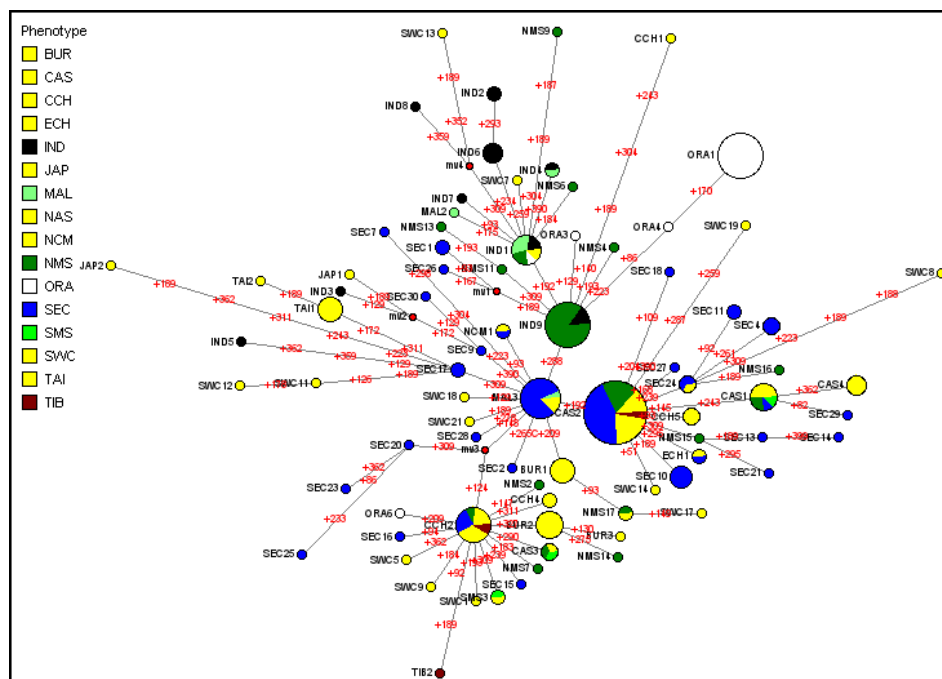


Figura 5.2.15: Caso de teste 3 – Árvore filogenética.

Apresenta-se na Tabela 5.2.3 as características da árvore.

Tabela 5.2.3: Caso de teste 3 – Características da árvore filogenética.

Caso de Teste 3 – Árvore Filogenética	
Número total de nós	85
Número de <i>median vectors</i>	4
Frequência total	241
Número de categorias taxonômicas	118
Categorias taxonômicas contidas dentro de outros nós	33
Número de erros cometidos pelo especialista	2

Apresenta-se na Figura 5.2.16 os parâmetros de entrada para o cálculo das estatísticas desta árvore. Definiu-se como população descendente, as populações das regiões NMS, SMS, IND e MAL, as populações das restantes regiões como ancestrais, à exceção de TIB e ORA que são indeterminadas.

Estatísticas

Select Root:	MAL3
Select f:	f1
Enter Root Mutations:	
TIB	Undetermined
TAI	Ascendant
SEC	Ascendant
MAL	Descendant
BUR	Ascendant
IND	Descendant
ECH	Ascendant
JAP	Ascendant
NCM	Ascendant
NAS	Ascendant
CCH	Ascendant
SMS	Descendant
NMS	Descendant
ORA	Undetermined
SWC	Ascendant
CAS	Ascendant

Ok

Figura 5.2.16: Caso de teste 3 – Parâmetros de entrada para f1.

O ficheiro em Excel obtido para f1 é o apresentado na Figura 5.2.17.

	A	B	C	D	E	F	G
1	f1						
2		n	rho	se	n	rho	se
3	+327 +124 +148	1	0.0	0.0	9	0.222222222222222	0.15713484026367722
4	+309 +192 +288	1	0.0	0.0	2	1.5	0.8660254037844386
5	+124 +148	6	0.833333333333333	0.44095855184409843	37	0.756756756756756	0.2834618508567977
6	+172 +390	1	0.0	0.0	2	1.0	0.7071067811865476
7	+192 +288	19	0.8421052631578947	0.37216146378239345	24	0.916666666666666	0.31732387941099616
8	+239 +192	1	0.0	0.0	10	1.0	0.4242640687119285
9	+309 +192	1	0.0	0.0	4	1.0	0.6123724356957945
10	+145 +243 +192	3	0.0	0.0	13	0.3846153846153846	0.31716197120135853
11	+93 +209	1	0.0	0.0	3	0.333333333333333	0.333333333333333
12	+309	1	0.0	0.0	13	2.1538461538461537	0.8284868934053082
13	+288	23	0.17391304347826086	0.10649955403405123	73	1.6849315068493151	0.6035175553948134
14	+192	7	0.0	0.0	79	0.9620253164556962	0.2941506338891709
15		1	0.0	0.0	241	2.215767634854772	0.5497259804706787

Figura 5.2.17: Caso de teste 3 – Ficheiro Excel resultante para f1.

Traduzindo este ficheiro em termos de fundadores encontrados, temos na Figura 5.2.18 os 13 fundadores encontrados na árvore.

Estatísticas

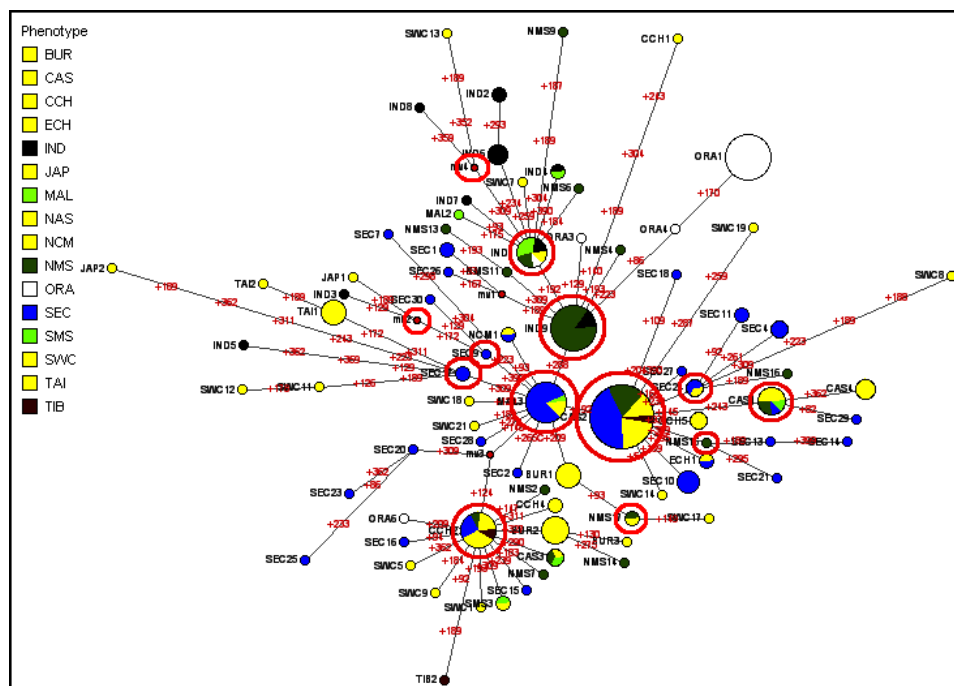


Figura 5.2.18: Caso de teste 3 – Árvore filogenética com fundadores para f1.

De seguida, submeteu-se a mesma árvore com os mesmos parâmetros, excepto o critério f2, em vez de f1, como apresentado na Figura 5.2.19.

Select Root:	Select f:	Enter Root Mutations:
MAL3	f2	
TIB	Undetermined	
TAI	Ascendant	
SEC	Ascendant	
MAL	Descendant	
BUR	Ascendant	
IND	Descendant	
ECH	Ascendant	
JAP	Ascendant	
NCM	Ascendant	
NAS	Ascendant	
CCH	Ascendant	
SMS	Descendant	
NMS	Descendant	
ORA	Undetermined	
SWC	Ascendant	
CAS	Ascendant	

Ok

Figura 5.2.19: Caso de teste 3 – Parâmetros de entrada para f2.

Estatísticas

O ficheiro em Excel obtido para f2 é o apresentado na Figura 5.2.20.

	A	B	C	D	E	F	G
1	f2						
2		n	rho	se	n	rho	se
3	+124 +148	7	1.0	0.4285714285714286	37	0.7567567567567568	0.2834618508567977
4	+192 +288	20	0.9	0.36055512754639896	24	0.9166666666666666	0.31732387941099616
5	+239 +192	1	0.0	0.0	10	1.0	0.4242640687119285
6	+309 +192	1	0.0	0.0	4	1.0	0.6123724356957945
7	+145 +243 +192	3	0.0	0.0	13	0.38461538461538464	0.31716197120135853
8	+309	1	0.0	0.0	13	2.1538461538461537	0.8284868934053082
9	+390	1	0.0	0.0	4	1.25	0.6614378277661477
10	+288	23	0.17391304347826086	0.10649955403405123	73	1.6849315068493151	0.6035175553948134
11	+192	7	0.0	0.0	79	0.9620253164556962	0.2941506338891709
12		2	0.5	0.5	241	2.215767634854772	0.5497259804706787

Figura 5.2.20: Caso de teste 3 – Ficheiro Excel resultante para f2.

Traduzindo este ficheiro Excel em termos de fundadores encontrados, temos na Figura 5.2.21 os 10 fundadores encontrados na árvore utilizando o critério f2. Assim, comparando com f1, vemos que se obtiveram menos 3 fundadores: mv2, mv4 e NMS17.

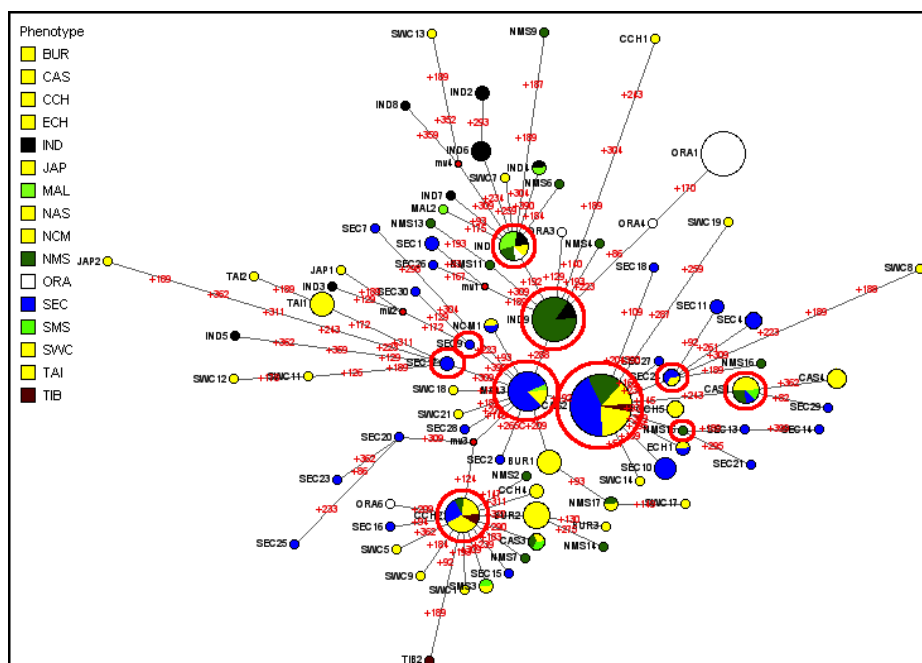


Figura 5.2.21: Caso de teste 3 – Árvore filogenética com fundadores para f2.

5.2.4 Caso de Teste 4

O caso de teste 4 utiliza outra árvore filogenética, apresentada na Figura 5.2.22. É uma árvore de complexidade semelhante à dos anteriores casos de estudo. A raiz da árvore é o nó IND1.

Estatísticas

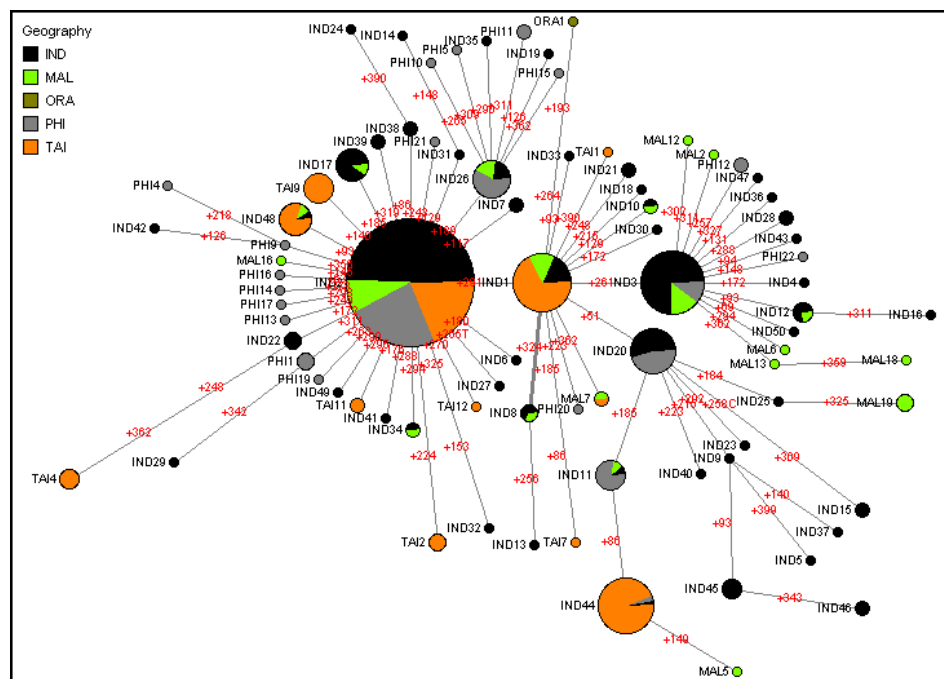


Figura 5.2.22: Caso de teste 4 – Árvore filogenética.

Apresenta-se na Figura 5.2.23 os parâmetros de entrada para o cálculo das estatísticas desta árvore. Definiu-se como descendente, a população da região TAI. As populações das restantes regiões foram definidas como ancestrais.

Apresenta-se na Tabela 5.2.4 as características da árvore.

Tabela 5.2.4: Caso de teste 4 – Características da árvore filogenética.

Caso de Teste 4 – Árvore Filogenética	
Número total de nós	83
Número de <i>median vectors</i>	não tem
Frequência total	444
Número de categorias taxonómicas	105
Categorias taxonómicas contidas dentro de outros nós	22
Número de erros cometidos pelo especialista	0

Estatísticas

Figura 5.2.23: Caso de teste 4 – Parâmetros de entrada para f1.

O ficheiro em Excel obtido para f1 é o apresentado na Figura 5.2.24.

	A	B	C	D	E	F	G
1	f1						
2		n	rho	se	n	rho	se
3	+86 +185 +51	28	0.0	0.0	31	0.03225806451612903	0.03225806451612903
4	+291	58	0.6724137931034483	0.2631782331460991	258	0.4573643410852713	0.11781925700442375
5		25	0.16	0.08	444	1.481981981981982	0.6358535327577617

Figura 5.2.24: Caso de teste 4 – Ficheiro Excel resultante para f1.

Temos na Figura 5.2.25 os 3 fundadores encontrados na árvore representados.

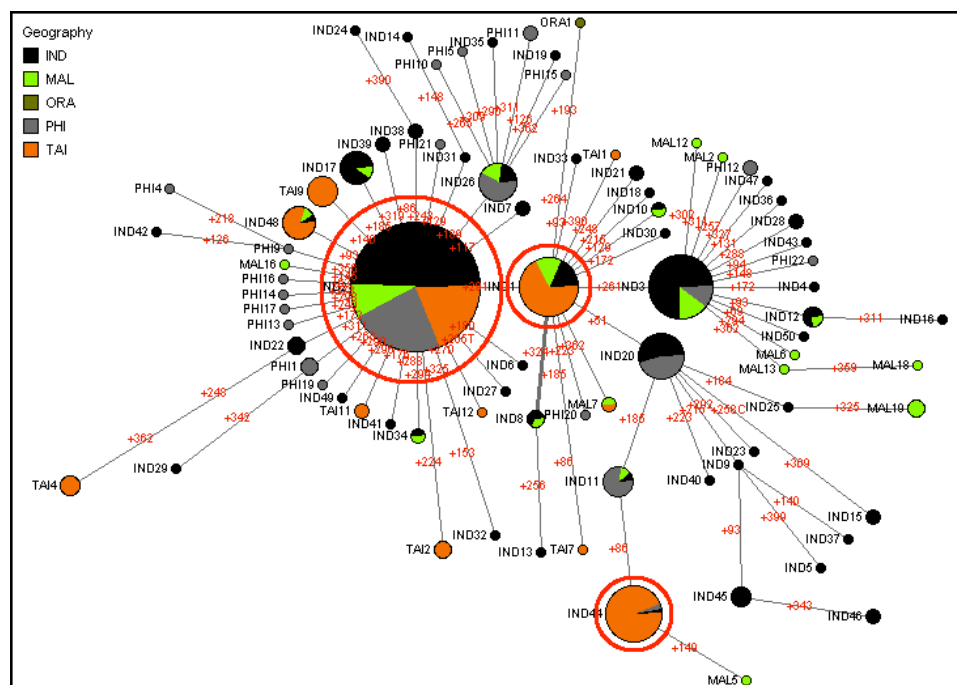


Figura 5.2.25: Caso de teste 4 – Árvore filogenética com fundadores para f1.

Estatísticas

De seguida, submeteu-se a mesma árvore com os mesmos parâmetros, excepto o critério f2, em vez de f1, como apresentado na Figura 5.2.26.

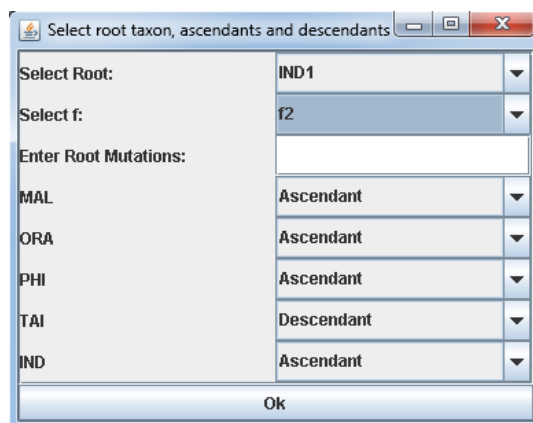


Figura 5.2.26: Caso de teste 4 – Parâmetros de entrada para f2.

O ficheiro em Excel obtido para f2 é o apresentado na Figura 5.2.27.

	A	B	C	D	E	F	G
1	f2						
2		n	rho	se	n	rho	se
3	+51	28	0.0	0.0	77	1.3506493506493507	0.6773197304106647
4	+291	58	0.6724137931034483	0.2631782331460991	258	0.4573643410852713	0.11781925700442375
5		25	0.16	0.08	444	1.481981981981982	0.6358535327577617

Figura 5.2.27: Caso de teste 4 – Ficheiro Excel resultante para f2.

Traduzindo este ficheiro Excel em termos de fundadores encontrados, temos na Figura 5.2.28 os 3 fundadores encontrados na árvore utilizando o critério f2. Curiosamente, neste caso de estudo, ao contrário do que tem acontecido nos anteriores, o número de fundadores não baixou com o critério f2 em relação ao f1. Isto aconteceu porque o fundador IND44 (em f1) foi substituído pelo IND20 (em f2), isto porque em f1 o IND44, que contém a sequência descendente (TAI), só tem um ramo derivado na população de origem (MAL5), o suficiente para ser fundador. No entanto, em f2, como IND44 só possui esse ramo derivado já não é suficiente para ser fundador, e assim IND20, que em f1 não possuía nenhuma sequência com a população de destino (uma vez que IND44 era fundador e consequentemente não contava), em f2 passa a possuir um ramo derivado com a população de destino e em relação a ramos derivados com a população de origem tem mais do que 2, o que faz com que passe a ser fundador em f2.

Estatísticas

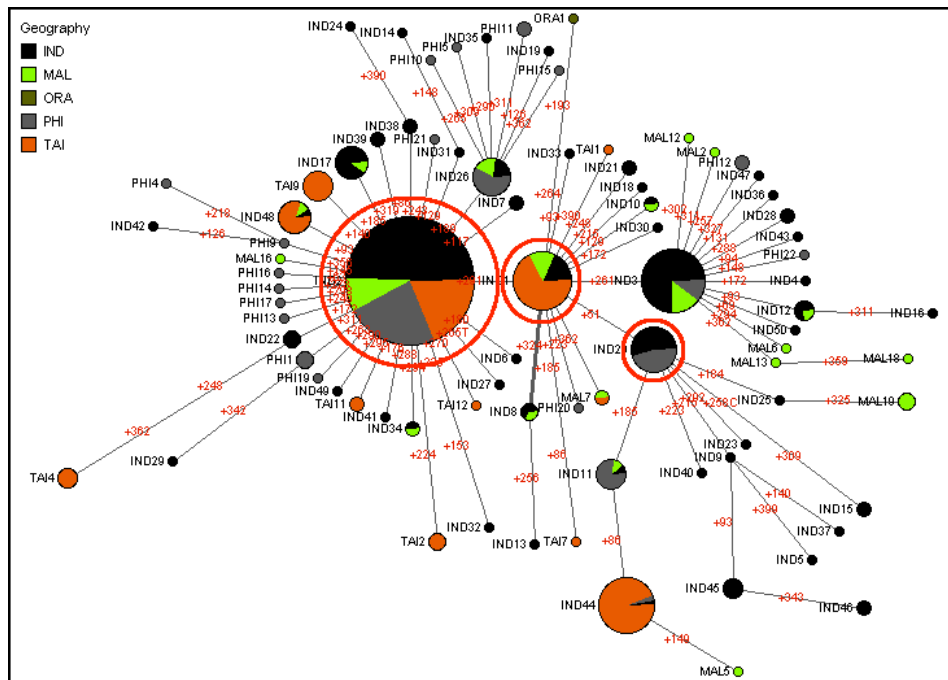


Figura 5.2.28: Caso de teste 4 – Árvore filogenética com fundadores para f2.

5.2.5 Caso de Teste 5

O caso de teste 5 utiliza outra árvore filogenética, apresentada na Figura 5.2.29. É uma árvore mais complexa que a dos anteriores casos de estudo, uma vez que possui mais arestas e nós. A raiz da árvore é o nó CCH4.

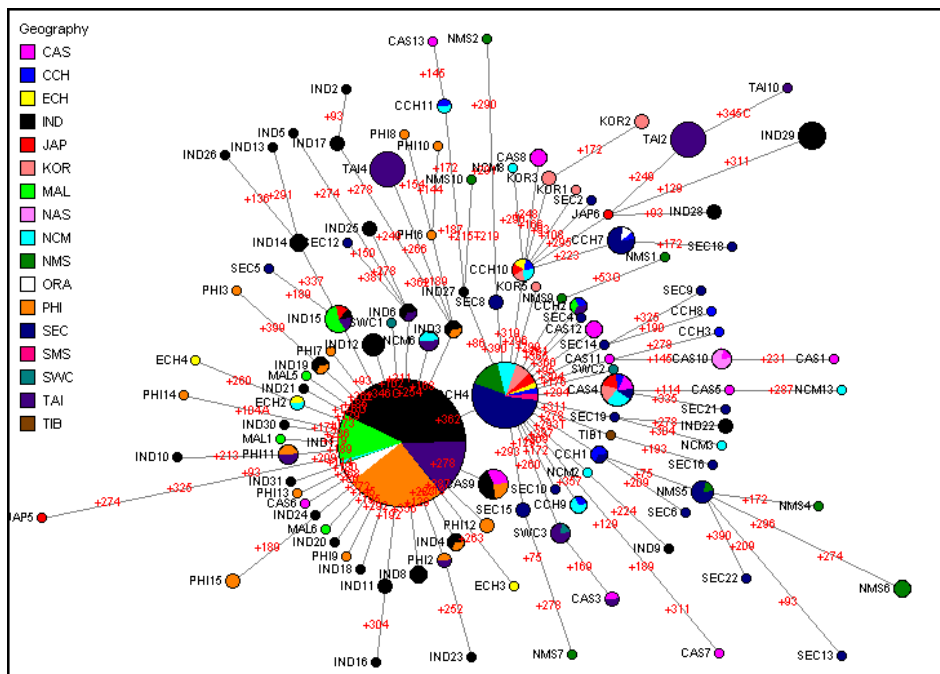


Figura 5.2.29: Caso de teste 5 – Árvore filogenética.

Estatísticas

Apresenta-se na Tabela 5.2.5 as características da árvore.

Tabela 5.2.5: Caso de teste 5 – Características da árvore filogenética.

Caso de Teste 5 – Árvore Filogenética	
Número total de nós	111
Número de <i>median vectors</i>	não tem
Frequência total	440
Número de categorias taxonómicas	158
Categorias taxonómicas contidas dentro de outros nós	47
Número de erros cometidos pelo especialista	3

Apresenta-se na Figura 5.2.30 os parâmetros de entrada para o cálculo das estatísticas desta árvore. Definiu-se como descendente, a população das regiões IND e PHI. As populações das restantes regiões foram definidas como ancestrais.

Enter Root Mutations:	
JAP	Ascendant
NCM	Ascendant
ORA	Ascendant
NAS	Ascendant
TAI	Ascendant
MAL	Ascendant
ECH	Ascendant
CAS	Ascendant
IND	Descendant
CCH	Ascendant
PHI	Descendant
SEC	Ascendant
TIB	Ascendant
KOR	Ascendant
NMS	Ascendant
SWC	Ascendant
SMS	Ascendant

Figura 5.2.30: Caso de teste 5 – Parâmetros de entrada para f1.

O ficheiro em Excel obtido para f1 é o apresentado na Figura 5.2.31.

Estatísticas

	A	B	C	D	E	F	G
1	f1						
2		n	rho	se	n	rho	se
3	+93 +362	6	0.333333333333333	0.23570226039551584	13	0.6153846153846154	0.40703866324070626
4	+311 +362	5	0.8	0.4898979485566356	7	0.7142857142857143	0.3779644730092272
5	+168 +362	9	1.666666666666667	0.693888666488711	22	1.2727272727272727	0.655554777357089
6	+295 +319	9	1.7777777777777777	1.1221672153735642	24	1.2916666666666667	0.7204840194079409
7	+311	2	0.0	0.0	4	0.75	0.5590169943749475
8	+86	1	0.0	0.0	5	1.6	0.8944271909999159
9	+362	155	0.3225806451612903	0.0784872584554609	271	0.5830258302583026	0.12761340565159102
10		1	0.0	0.0	440	1.5772727272727274	0.6378553943741727

Figura 5.2.31: Caso de teste 5 – Ficheiro Excel resultante para f1.

Traduzindo este ficheiro em termos de fundadores encontrados, temos na Figura 5.2.32, os 8 fundadores encontrados na árvore.

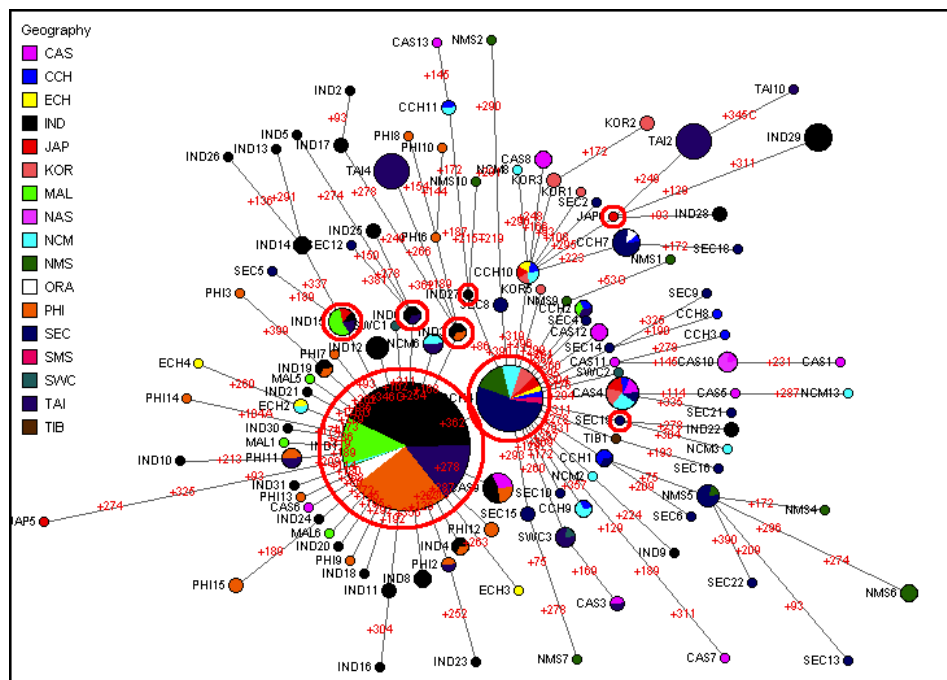


Figura 5.2.32: Caso de teste 5 – Árvore filogenética com fundadores para f1.

De seguida, submeteu-se a mesma árvore com os mesmos parâmetros, excepto o critério f2, em vez de f1, como apresentado na Figura 5.2.33.

Estatísticas

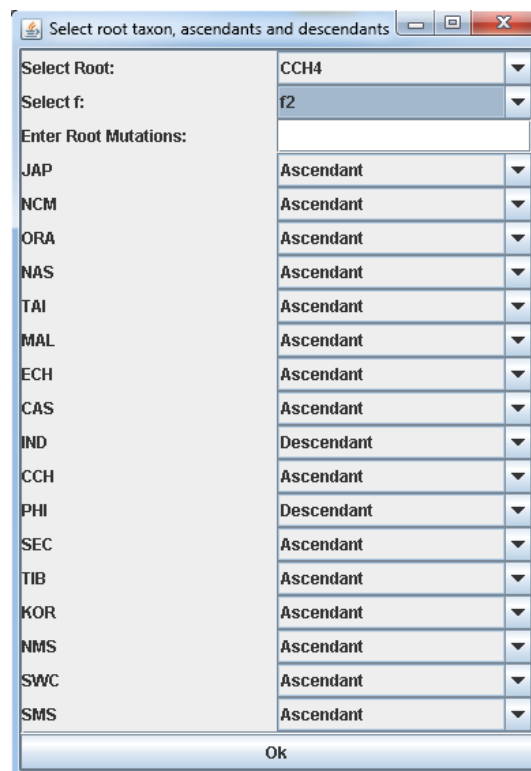


Figura 5.2.33: Caso de teste 5 – Parâmetros de entrada para f2.

O ficheiro em Excel obtido para f2 é o apresentado na Figura 5.2.34.

	A	B	C	D	E	F	G
1	f2						
2		n	rho	se	n	rho	se
3	+86	1	0.0	0.0	5	1.6	0.8944271909999159
4	+362	175	0.5485714285714286	0.10872170051680256	271	0.5830258302583026	0.12761340565159102
5	+319	9	1.7777777777777777	1.1221672153735642	48	1.6041666666666667	0.6551743830631828
6		3	0.6666666666666666	0.6666666666666666	440	1.5772727272727274	0.6378553943741727

Figura 5.2.34: Caso de teste 5 – Ficheiro Excel resultante para f2.

Traduzindo este ficheiro Excel em termos de fundadores encontrados, temos na Figura 5.2.35 os 4 fundadores encontrados na árvore utilizando o critério f2. O que aconteceu foi que os fundadores IND3, IND6, IND15, SEC19 e JAP6, existentes com o critério f1, deixaram de ser fundadores em f2, tendo-se mantido todos os outros, mas houve um novo fundador em f2, o CCH10. Isto aconteceu, pela mesma razão do caso de estudo 5, ou seja, com JAP6 a deixar de ser fundador, CCH10 passa a ter sequências da população de destino nos ramos derivados, e como já possui vários ramos derivados com a população de origem, passa a ser fundador em f2.

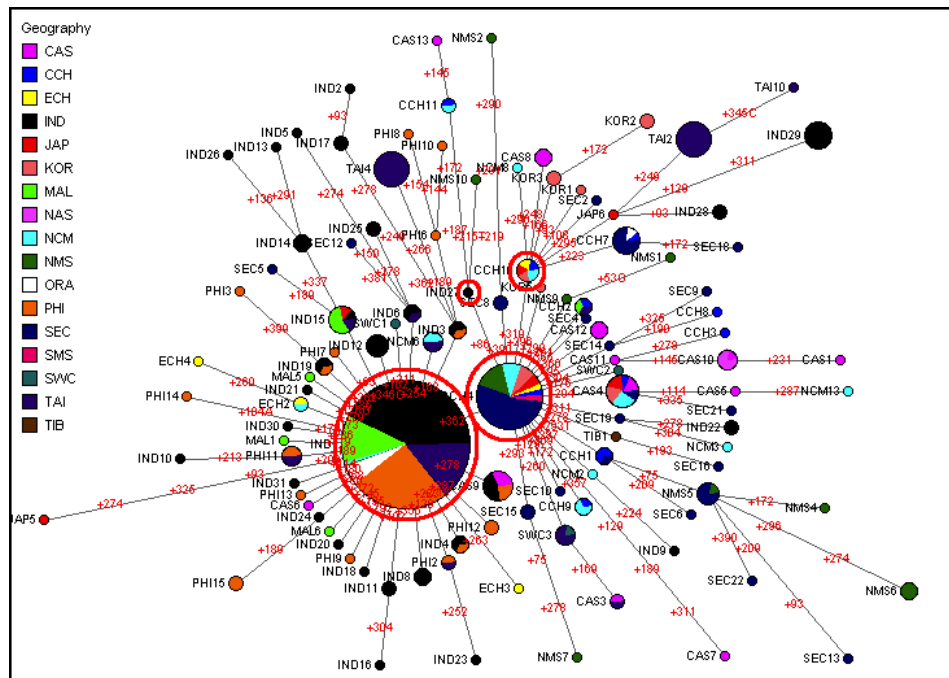


Figura 5.2.35: Caso de teste 5 – Árvore filogenética com fundadores para f2.

5.3 Resumo

Com estes resultados experimentais é possível concluir que a implementação desta ferramenta para cálculo de estatísticas foi um sucesso, pois automatizou-se um processo que anteriormente era demasiado trabalhoso e sujeito a erros, sendo que de agora em diante será possível executar com maior frequência este processo e obter dados mais rapidamente e com maior fiabilidade.

Estes casos de teste apresentados (e outros utilizados) ajudaram a automatizar este cálculo de estatísticas de forma correcta, uma vez que serviram também para, durante o desenvolvimento, verificar erros da aplicação e resolvê-los, uma vez que este cálculo possui um vasto conjunto de regras a ter em conta.

Na Secção 7.1 apresenta-se com maior detalhe o objectivo destes dados estatísticos obtidos, exportados para ficheiro legível em Excel.

Capítulo 6

Aplicação Desenvolvida

6.1 Casos de Utilização

Nesta secção pretende-se apresentar de forma clara todas as funcionalidades a que o utilizador tem acesso na aplicação desenvolvida.

A aplicação consiste em quatro módulos distintos, nomeadamente:

- Geração de Árvores – contém as funções para obter árvores a partir de redes filogenéticas, através dos algoritmos desenvolvidos
- Escolha de Arestas – contém as funções necessárias para alterar uma árvore ao gosto do utilizador, ou seja é possível a escolha de caminhos alternativos
- Limpar *Median Vectors* – módulo encarregue por receber uma árvore, pesquisar todos os *median vectors* desnecessários e removê-los, tendo de reestabelecer as conexões existentes, através de um algoritmo desenvolvido para o efeito
- Estatísticas – contém as funções para o cálculo de estatísticas através de um dos critérios existentes (f1, f2 ou fs)

Apresenta-se na Figura 6.1.1 o diagrama de casos de utilização da aplicação desenvolvida.

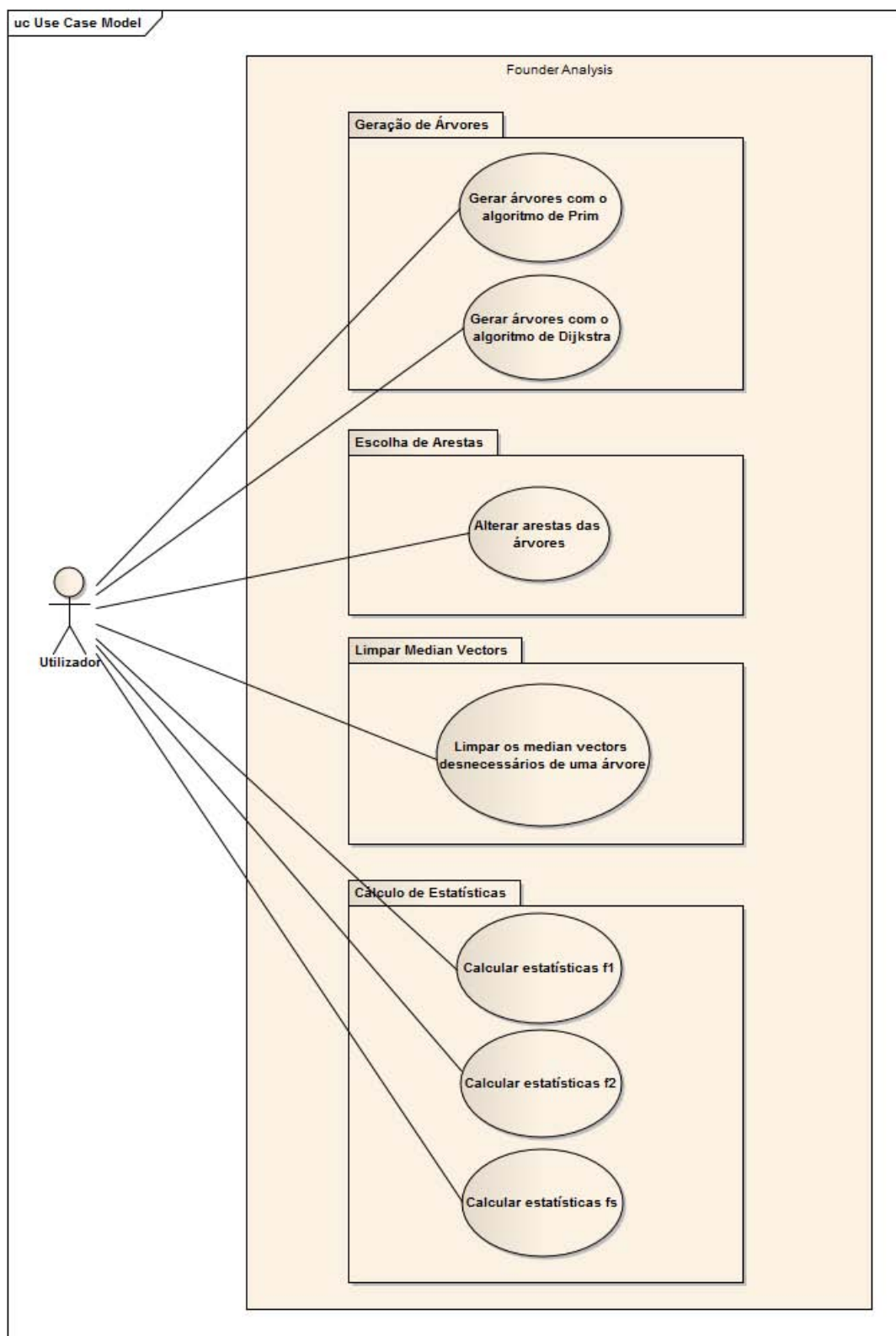


Figura 6.1.1: Diagrama de casos de utilização da aplicação

6.2 Interface da Aplicação

Nesta secção apresentam-se alguns *screenshots* da aplicação desenvolvida para apresentar o seu aspecto final.

Na Figura 6.2.1 é apresentado o menu inicial da aplicação, onde estão quatro botões que ligam aos quatro módulos desenvolvidos.

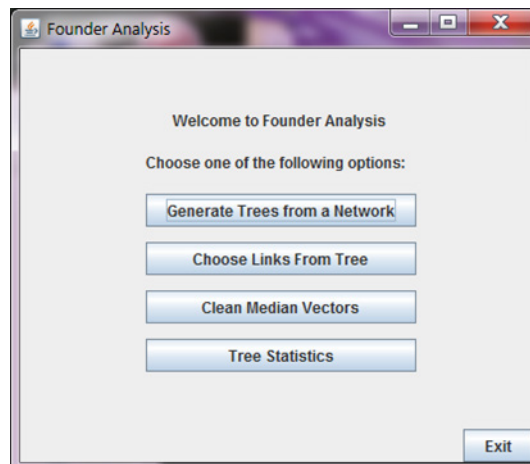


Figura 6.2.1: Menu inicial da aplicação

6.2.1 Módulo de Geração de Árvores

Neste módulo é pedido ao utilizador para escolher um ficheiro de *input* do Network (fdi ou out) que contenha uma rede filogenética da qual se queira obter árvores. Após a escolha do ficheiro, é apresentada a interface da Figura 6.2.2, onde é possível especificar a raiz da árvore, o algoritmo pretendido, o número de árvores a gerar e o valor máximo do custo total de cada árvore.

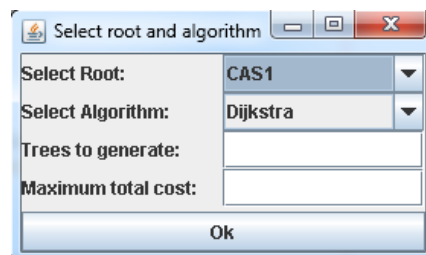


Figura 6.2.2: Módulo de geração de árvores

De seguida o algoritmo escolhido será executado com os parâmetros introduzidos e os ficheiros de cada árvore são criados e colocados no directório da aplicação.

6.2.2 Módulo de Escolha de Arestas

Neste módulo o utilizador terá de escolher os ficheiros da árvore e da rede filogenética correspondente (para obter as arestas alternativas que não estão presentes na árvore). De seguida aparecerá uma janela para escolher as arestas pretendidas (Figura 6.2.3).

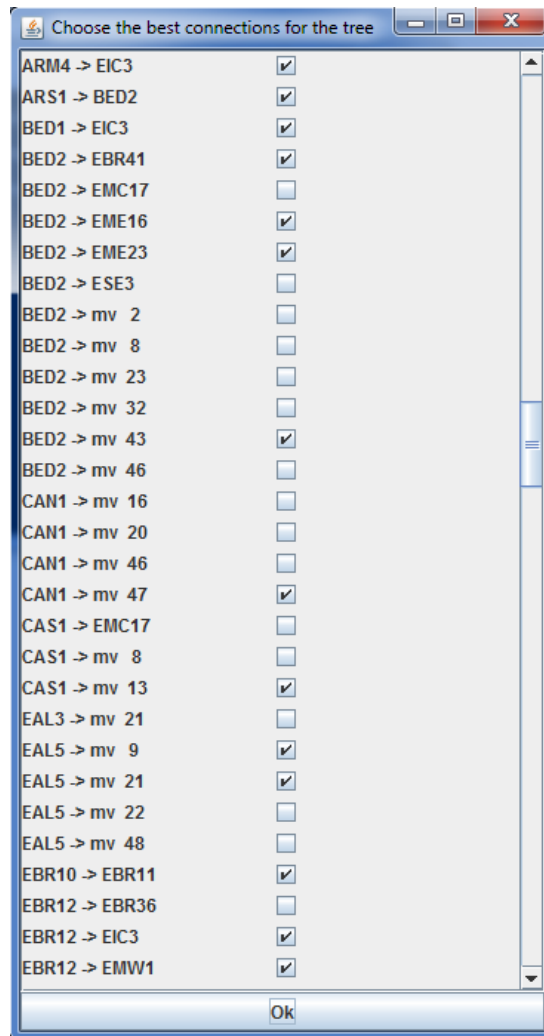


Figura 6.2.3: Módulo de escolha de arestas

6.2.3 Módulo de Limpeza de *Median Vectors*

Neste módulo é pedido ao utilizador para inserir o ficheiro de *input* de uma árvore e escolher a sua raiz. De seguida são removidos os *median vectors* desnecessários da árvore e é criado um novo ficheiro (out ou fdi) com a nova árvore.

6.2.4 Módulo de estatísticas

Neste módulo é também pedido ao utilizador para escolher um ficheiro de *input* de uma árvore filogenética. De seguida é necessário preencher o formulário que irá aparecer (Figura 6.2.4). Basicamente, este formulário serve para escolher a raiz da árvore, o critério estatístico de cálculo de fundadores pretendido, bem como as populações ascendentes e descendentes. Caso exista no directório da aplicação um ficheiro de texto auxiliar, *phenotypes.txt*, que contém esta informação, as populações já terão a opção escolhida, pelo que será apenas necessário nesta janela especificar a raiz da árvore e o critério, caso contrário, as populações estarão todas definidas como descendentes e será necessário o utilizador escolher a opção correcta para cada uma. No Anexo B.4 está um exemplo do ficheiro *phenotypes.txt*.

Taxon	Selection
ANA	Ascendant
ANC	Undetermined
ANE	Undetermined
ANW	Undetermined
ARC	Ascendant
ARE	Ascendant
ARM	Ascendant
ARS	Ascendant
BED	Undetermined
CAN	Descendant
CAS	Descendant
EAE	Undetermined
EAL	Descendant
EBR	Descendant
EIC	Descendant
EMC	Descendant
EME	Descendant
EMW	Descendant
ENC	Descendant
ENE	Descendant
ENW	Descendant
ESC	Descendant

Figura 6.2.4: Módulo de estatísticas

6.3 Informação Adicional

Resta acrescentar que a aplicação guarda sempre o caminho do último directório dos ficheiros especificados pelo utilizador num ficheiro auxiliar `last_dir.txt`, isto porque as janelas de escolha de ficheiro da biblioteca Swing apresentam como directório inicial a pasta “Meus Documentos” e seria necessário o utilizador percorrer várias vezes o caminho para chegar à pasta onde tem os ficheiros que utiliza. Assim, sempre que é aberta uma janela de escolha de ficheiro, o directório inicial apresentado será o que está guardado no ficheiro `last_dir.txt`. No Anexo [B.1](#) apresenta-se um guia de utilização, onde é apresentado com maior detalhe o modo de utilização da aplicação. No Anexo [B.2](#) apresenta-se o diagrama de classes da aplicação.

Capítulo 7

Conclusões e Trabalho Futuro

7.1 Satisfação dos Objectivos

Apresenta-se a Tabela 7.1.1, que apresenta a descrição e execução dos objectivos do projecto.

Tabela 7.1.1: Cumprimento dos objectivos do projecto.

Objectivo	Descrição	Implementado	Não implementado
1	Algoritmo de geração de árvores a partir de uma rede filogenética	X	
2	Ferramenta para automatizar o cálculo de estatísticas	X	
3	Interface gráfica para utilização das ferramentas	X	
4	Melhorias na Análise de Fundador		X

O primeiro objectivo deste projecto, que consistia em criar um algoritmo para geração de árvores a partir de uma rede filogenética, previamente existente, foi cumprido com sucesso. As etapas deste objectivo consistiram em estudar qual o algoritmo de árvore de expansão mínima que se deveria utilizar como base do algoritmo a desenvolver, optando-se pelo algoritmo de Prim, implementar o algoritmo escolhido e implementar as modificações e casos particulares necessários para gerar não só uma árvore, mas várias árvores com um custo total aceitável.

O segundo objectivo foi, de igual modo, cumprido com sucesso. Consistia em produzir uma ferramenta para automatizar o cálculo de estatísticas nas árvores filogenéticas. Esta ferramenta calcula os fundadores, as frequências relativas e absolutas, os valores de ρ (rho) e de

desvio padrão para cada uma das frequências. Para além disso tem ainda de exportar os dados para um ficheiro legível em aplicações de folha de cálculo do género Microsoft Excel. O ficheiro criado, legível em Excel, servirá para ser submetido a uma aplicação, já existente, que contém um relógio molecular que, com base nos dados estatísticos presentes no ficheiro, estima as idades de cada um dos fundadores com a respectiva margem de erro.

O terceiro principal objectivo consiste em desenvolver uma interface gráfica intuitiva que permita ao utilizador final da aplicação realizar todas as operações pretendidas para executar a Análise de Fundador. Este objectivo também foi concretizado, utilizando a biblioteca Swing do Java. Na interface gráfica desenvolvida é possível escolher as opções de gerar as melhores árvores a partir de uma rede filogenética, introduzida pelo utilizador, e calcular as estatísticas obtidas a partir de uma árvore filogenética, introduzida pelo utilizador.

Os valores de ρ calculados nas estatísticas serão transformados em tempos de migração (em anos) utilizando um relógio molecular (16677 anos em média para uma mutação ocorrer) para a sequência em estudo [SET09]. As idades calculadas para cada um dos fundadores na população descendente e o respectivo erro-padrão permitirão distribuir estatisticamente cada fundador por possíveis tempos de migração. A junção de todos os fundadores obtidos na população permitirá obter um padrão de possíveis tempos de migração para a região em estudo.

O gráfico seguinte (Figura 7.1.1) mostra o resultado final de uma análise de fundador utilizando um critério f1. O objectivo desta análise era detectar os tempos de migração da Ásia continental para as Ilhas do Japão.

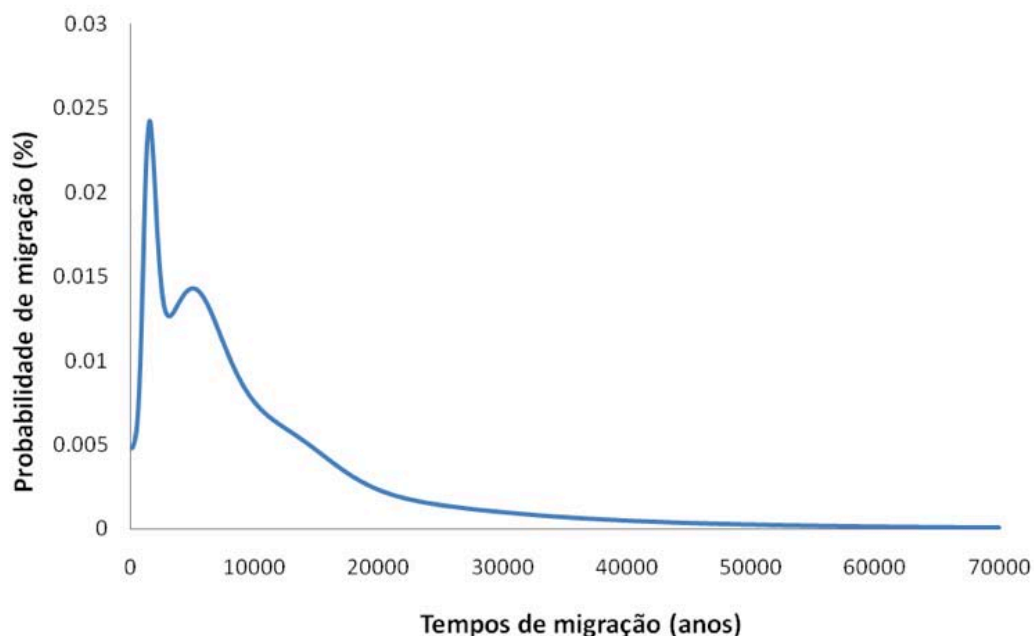


Figura 7.1.1: Distribuição da probabilidade de migração para as Ilhas do Japão baseado na estimativa das idades de 214 fundadores utilizando ρ .

Conclusões e Trabalho Futuro

A distribuição dos diferentes fundadores detectados de acordo com as idades calculadas utilizando ρ indica essencialmente dois picos de provável tempo de migração, um maior há 1800 anos e um segundo há 5000 anos. O primeiro pico (maior) apoia o facto de a maioria da população japonesa actual ser descendente de uma comunidade de agricultores, os Yayoi, que migraram para as ilhas há cerca de 2000 anos.

Este resultado foi obtido utilizando uma base de dados de 14361 amostras asiáticas das quais 1366 correspondem a indivíduos do Japão (população descendente). Foram produzidas 63 redes filogenéticas, as quais foram transformadas em árvores, e tendo sido detectados 214 fundadores. Um geneticista populacional experiente produziria resultados esperados ao ritmo de 1 rede analisada por dia (construção da rede, transformação em árvore, detecção de fundadores, cálculos de ρ dos fundadores e desvios padrão) e desse modo levaria mais de 60 dias úteis para produzir estes resultados. Com a ferramenta produzida e pelo facto de o investigador poder trabalhar directamente a partir de ficheiros out do Network, a duração deste trabalho poderia ser reduzida a 2 ou 3 dias.

Concluído o desenvolvimento, foram efectuados diversos testes à aplicação, assim como durante o período de desenvolvimento, para testar a sua conformidade com os objectivos do projecto. Conclui-se que a aplicação satisfaz todos os objectivos principais, restando apenas alguns objectivos adicionais que não foram implementados.

Através da apresentação da aplicação aos profissionais da área, quer durante o seu desenvolvimento, quer a versão final, foi possível torná-la no mais intuitiva possível do ponto de vista dos seus utilizadores finais, o que fez com que o nível de satisfação final seja aceitável.

7.2 Trabalho Futuro

O trabalho futuro consistirá em implementar o único objectivo que não foi cumprido, tal como apresentado na Tabela 7.1.1, ou seja, estudar possíveis melhorias na análise de fundador. Houve uma melhoria que foi introduzida neste projecto, que foi a substituição da Equação (3.5.2) pela (3.5.3) para cálculo do desvio padrão. Esta característica pode ser considerada uma melhoria à Análise de Fundador, já que antes era aplicada com a primeira fórmula de cálculo.

Desde o início do projecto havia a possibilidade de fazer algumas melhorias à Análise de Fundador, juntamente com um dos autores da metodologia, Dr. Martin Richards (Universidade de Leeds) ou Dr. Vincent Macaulay (Universidade de Glasgow). Tal não chegou a acontecer durante o período de realização do projecto devido à indisponibilidade de ambos. Assim, pretende-se como perspectivas de trabalho futuro, estudar possíveis melhorias juntamente com os autores, o que poderá acontecer já no próximo mês de Julho, quando irá existir disponibilidade de um dos autores. Para além do estudo destas melhorias, pretende-se ainda que as mesmas sejam implementadas na aplicação desenvolvida e testadas, de forma a tirar conclusões acerca dos resultados que estas produzem.

Conclusões e Trabalho Futuro

Referências

- [AKC99] Richard M. Andrews, Iwona Kubacka, Patrick F. Chinnery, Robert N. Lightowlers, Douglass M. Turnbull, Neil Howell. *Reanalysis and revision of the Cambridge reference sequence for human mitochondrial DNA*. Nat Genet 23 (2): 147. doi:10.1038/13779. PMID 10508508. 1999.
- [And81] Eric Anderson, Alan T. Bankier, Bart G. Barrell, M. H. L. Bruijn, Alan R. Coulson, Jacques Drouin, I. C. Eperon, D. P. Nierlich, Bruce A. Roe, Frederick Sanger, Peter H. Schreier, Andrew J. H. Smith, Rodger Staden, Ian G. Young. *Sequence and organization of the human mitochondrial genome*. Nature 290: 457-465. 1981.
- [Ant05] Luís Antunes. *Grafos*. Faculdade de Ciências da Universidade do Porto. 2005. Disponível em: <http://www.dcc.fc.up.pt/~lfa/aulas/0506/mcc/praticas/grafos.pdf> Última consulta em 17 de Janeiro de 2011.
- [Aya86] Francisco J. Ayala. *On the virtues and pitfalls of the molecular evolutionary clock*. Journal of Heredity 77, 226-235. 1986.
- [BFR99] Hans-Jürgen Bandelt, Peter Forster, Arne Röhl. *Median-Joining Networks for Inferring Intraspecific Phylogenies*. Mol. Biol. Evol. 16(1):37-48. 1999.
- [BFS95] Hans-Jürgen Bandelt, Peter Forster, Bryan C. Sykes, Martin B. Richards. *Mitochondrial portraits of human populations using median networks*. Genetics 141:743-753. 1995.
- [Car08] Marcelo F. Carazzolle. *Métodos de Alinhamento de Sequências Biológicas*. 2008. <http://www.lge.ibi.unicamp.br/lgeextensao2008/extsup/alinhamentos.pdf> Última consulta em Janeiro de 2011.

Referências

- [CiViv11] Ciência Viva. *Acções de Divulgação - IPATIMUP*. Disponível em: <http://www.cienciaviva.pt/divulgacao/arteciencia/ipatimup.asp> Último acesso em Junho de 2011.
- [EncBrit11] Encyclopædia Britannica. *Königsberg bridge problem*. 2011. Disponível em <http://www.britannica.com/EBchecked/topic/321794/Konigsberg-bridge-problem> Última consulta em 18 de Janeiro de 2011.
- [ES94] Laurent Excoffier, Peter E. Smouse. *Using allele frequencies and geographic subdivision to reconstruct gene trees within a species: molecular variance parsimony*. Genetics 136: 343-359. 1994.
- [Fel10] Joe Felsenstein. *The Newick Tree Format*. University of Washington. 2010. Disponível em: <http://evolution.genetics.washington.edu/phylip/newicktree.html> Última consulta em 9 de Fevereiro de 2011.
- [FHT96] Peter Forster, Rosalind Harding, Antonio Torroni, Hans-Jürgen Bandelt. *Origin and evolution of native American mtDNA variation: a reappraisal*. The American Journal of Human Genetics 59, 935–45. 1996.
- [Fut10] Douglas J. Futuyma. *Evolution*. Chapter 2. 2nd ed. Sunderland, Massachusetts: Sinauer associates, Inc., cop. ISBN: 978-0-87893-223-8. 2009.
- [GBC80] Richard E. Giles, Hugues Blanc, Howard M. Cann, Douglas C. Wallace. *Maternal inheritance of human mitochondrial DNA*. Proc Natl Acad Sci USA 77:6715-6719. 1980.
- [HEF02] Corinna Herrnstadt, Joanna L. Elson, Eoin Fahy, Gwen Preston, Douglass M. Turnbull, Christen Anderson, Soumitra S. Ghosh, Jerrold M. Olefsky, M. Flint Beal, Robert E. Davis, Neil Howell. *Reduced-median-network analysis of complete mitochondrial DNA coding-region sequences for the major African, Asian, and European haplogroups*. The American Journal of Human Genetics 70:1152–1170 (erratum 71:448–449) 2002.
- [HRN01] John P. Huelsenbeck, Fredrik Ronquist, Rasmus Nielsen, Jonathan P. Bollback. *Bayesian Inference of Phylogeny and Its Impact on Evolutionary Biology*. Science's Compass Vol. 294 no. 5550 pp. 2310-2314. 2001.
- [IKP00] Max Ingman, Henrik Kaessmann, Svante Pääbo, Ulf Gyllensten. *Mitochondrial genome variation and the origin of modern humans*. Macmillian Magazines Ltd. 0028-0836. 2000.
- [IneID05] INESC-ID. *Teoria de Grafos e Algoritmos em Grafos*. 2005. Disponível em: <http://algos.inesc-id.pt/aed06/downloads/Slides/11-GrafosA.pdf> Última consulta em 1 de Junho de 2011.

Referências

- [Kli04] Caroline Klivans. *Handouts – Graph Theory 3*. 2004. Disponível em: <http://www.cis.cornell.edu/Courses/cis295/2004sp/graphs3.pdf> Última consulta em 17 de Janeiro de 2011.
- [LW00] Jeffrey T. Leell, Douglas C. Wallace. *The Peopling of Europe from the Maternal and Paternal Perspectives*. American Society of Human Genetics 67(6): 1376–1381. 2000.
- [MGF99] Gundela Meyer, André M. Goffinet, Alfonso Fairén. *What is a Cajal-Retzius cell? A reassessment of a classical cell type based on recent observations in the developing neocortex*. PMID: 10600995 [PubMed - indexed for MEDLINE] 1999.
- [Muh10] Rashid B. Muhammad. *Algorithms Lecture Notes*. 2010. Disponível em: <http://www.personal.kent.edu/~rmuhamma/Algorithms/MyAlgorithms/GraphAlgor/primAlgor.htm> Última consulta em 13 de Junho de 2011.
- [Pit01] Jay Pitocchelli. *Lecture Notes for Genetics*. Saint Anselm College. 2001.
- [RMH00] Martin B. Richards, Vincent Macaulay, Eileen Hickey, Emilce Vega, Bryan Sykes, Valentina Guida, Chiara Rengo, Daniele Sellitto, Fulvio Cruciani, Toomas Kivisild, Richard Villems, Mark Thomas, Serge Rychkov, Oksana Rychkov, Yuri Rychkov, Mukaddes Gölge, Dimitar Dimitrov, Emmeline Hill, Dan Bradley, Valentino Romano, Francesco Calì, Giuseppe Vona, Andrew Demaine, Surinder Papiha, Costas Triantaphyllidis, Gheorghe Stefanescu, Jiři Hatina, Michele Belledi, Anna Di Rienzo, Andrea Novelletto, Ariella Oppenheim, Søren Nørby, Nadia Al-Zaheri, Silvana Santachiara-Benerecetti, Rosaria Scozzari, Antonio Torroni, Hans-Jürgen Bandelt. *Tracing European Founder Lineages in the Near Eastern mtDNA Pool*. The American Journal of Human Genetics 67(5): 1251–1276. 2000.
- [Roc08] Ana Paula Rocha. *Slides da disciplina de Algoritmos e Estruturas de Dados I*. Faculdade de Engenharia da Universidade do Porto. 2008. Disponível em: <http://paginas.fe.up.pt/~arocha/AED1/APONTC/arvres.pdf> Última consulta em 18 de Janeiro de 2011.
- [Sch11] Michael Schimek. *Referate Bioinformatik I: Alignment of Pairs of Sequences*. University of Graz. Disponível em: http://user.medunigraz.at/michael.schimek/Alignment_of_pairs_of_sequences.pdf Última consulta em Janeiro de 2011
- [Sed02] Robert Sedgewick. *Algorithms in C* (part 5: Graph Algorithms), 3rd. edition. Addison-Wesley/Longman. 2002.

Referências

- [SET09] Pedro A. Soares, Luca Ermini, Noel Thomson, Maru Mormina, Teresa Rito, Arne Röhl, Antonio Salas, Stephen Oppenheimer, Vincent Macaulay, Martin B. Richards. *Correcting for Purifying Selection: An Improved Human Mitochondrial Molecular Clock* – Supplemental Data. The American Journal of Human Genetics 84, 740–759. 2009.
- [SFL00] Juliette Saillard, Peter Forster, Niels Lynnerup, Hans-Jürgen Bandelt, Søren Nørby. *mtDNA Variation among Greenland Eskimos: The Edge of the Beringian Expansion*. The American Journal of Human Genetics. 67:000–000. 2000.
- [SN87] Naruya Saitou, Masatoshi Nei. *The neighbor-joining method: a new method for reconstructing phylogenetic trees*. Mol. Biol. Evol. 4: 406-425. 1987.
- [Soa09] Pedro A. Soares. *Phylogenetics and Phylogeography of Human Mitochondrial DNA in Island Southeast Asia*. Institute of Integrative and Comparative Biology. Faculty of Biological Sciences – The University of Leeds. 2009.
- [ZP65] Emile Zuckerkandl, Linus Pauling. *Evolutionary divergence and convergence in proteins*. In: Bryson, V., Vogel, H. J. (Eds.). *Evolving Genes and Proteins*. Academic Press, New York, pp. 97-166. 1965.

Anexo A

Utilização do Network

A.1 Descrição da Utilização do Network 4.6

Para a criação da rede filogenética através da aplicação de um dos algoritmos já abordados é necessário ser escolhido no menu qual o algoritmo pretendido, como se pode ver na Figura A.1.

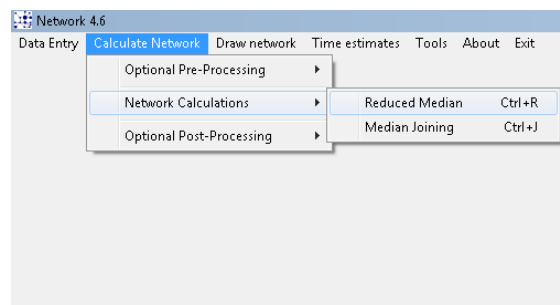


Figura A.1: Utilização do Network 4.6 (1).

De seguida introduz-se o ficheiro rdf que contém as sequências, previamente alinhadas, da forma que é representado na Figura A.2 e Figura A.3.

Utilização do Network

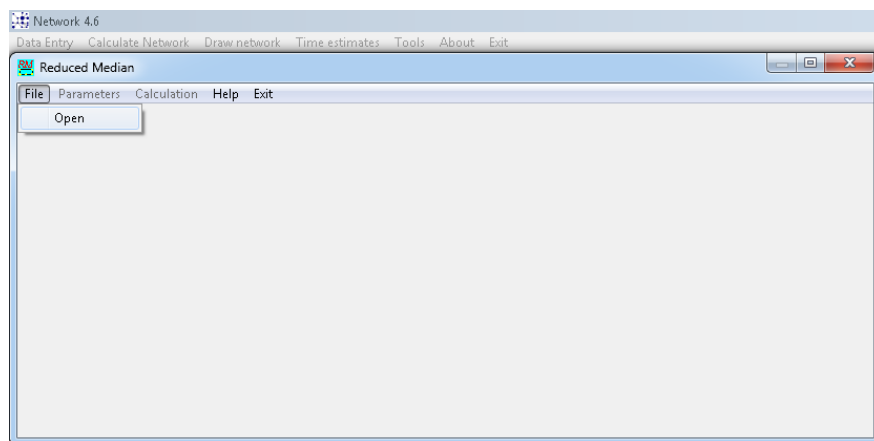


Figura A.2: Utilização do Network 4.6 (2).

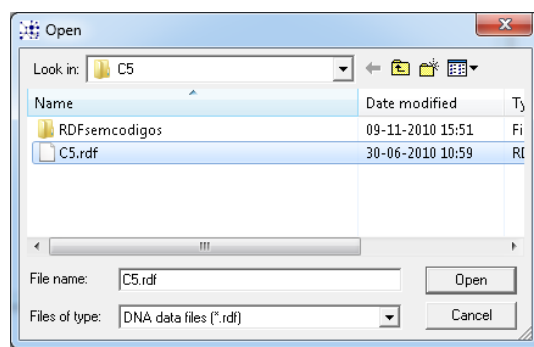


Figura A.3: Utilização do Network 4.6 (3).

De seguida, ao clicar em “Calculation” a rede é gerada através do algoritmo seleccionado anteriormente (Figura A.4).

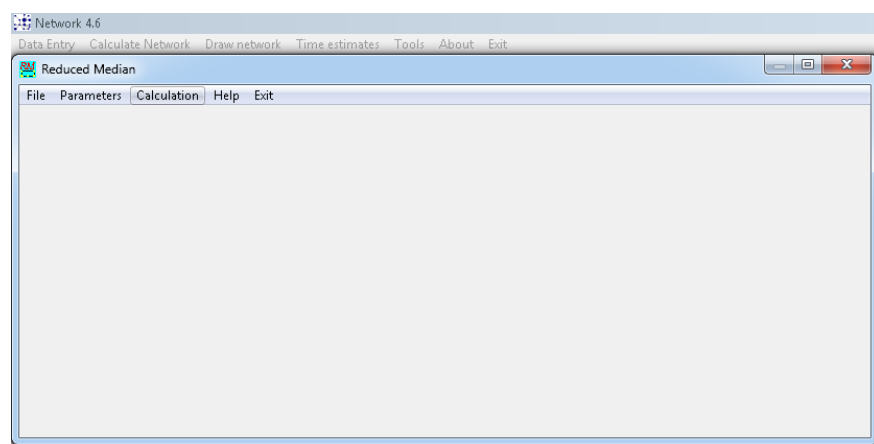


Figura A.4: Utilização do Network 4.6 (4).

Utilização do Network

Selecciona-se de seguida o nome do ficheiro no formato out que irá ser criado (Figura A.5).

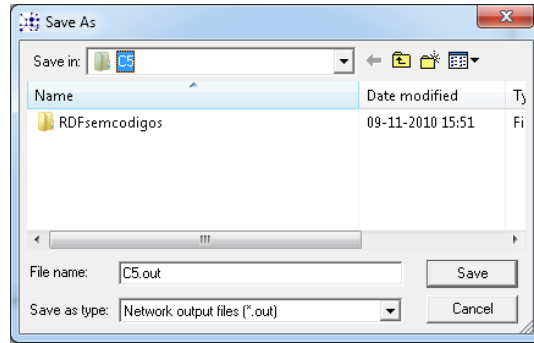


Figura A.5: Utilização do Network 4.6 (5).

Por fim, selecciona-se a opção “Draw Network” no menu inicial se se quiser visualizar graficamente a rede filogenética gerada, e para tal, selecciona-se o ficheiro C5.out gerado no passo anterior (Figura A.6 e Figura A.7).

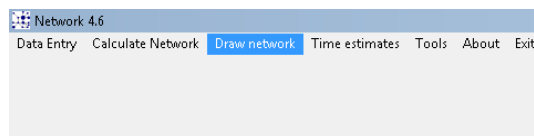


Figura A.6: Utilização do Network 4.6 (6).

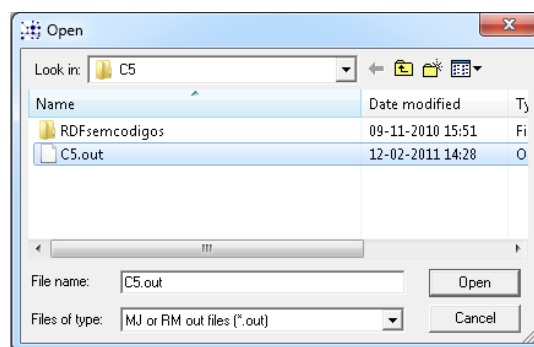


Figura A.7: Utilização do Network 4.6 (7).

Utilização do Network

O resultado obtido é o apresentado na Figura A.8.

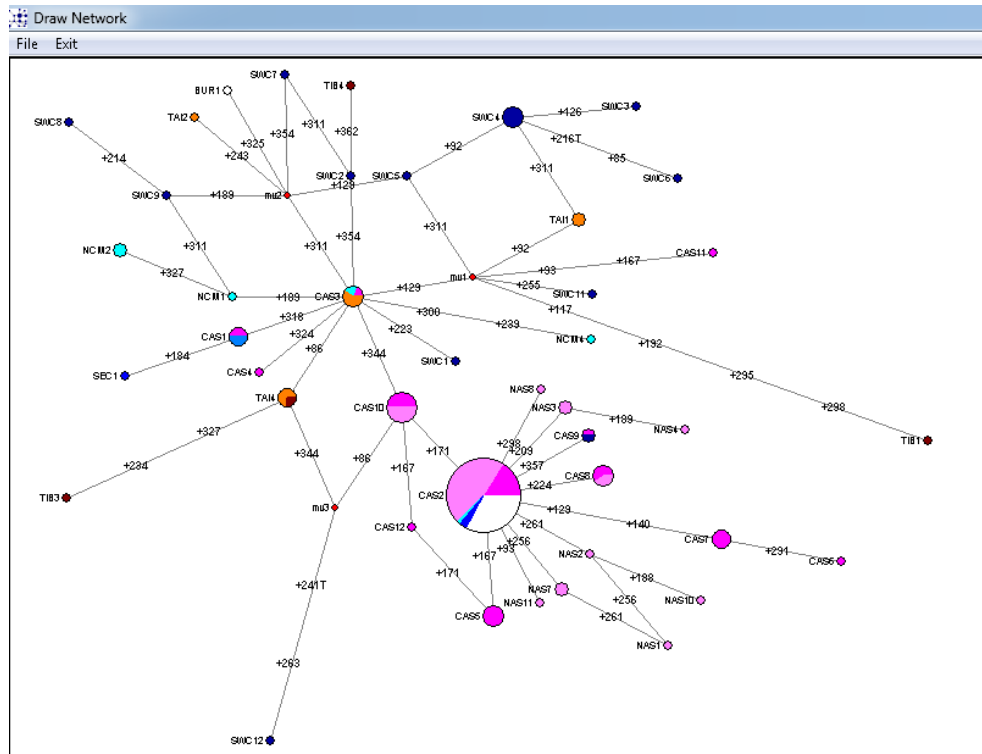


Figura A.8: Rede filogenética gerada.

Na Figura A.9, retirada do guia de utilizador do Network 4.6, é demonstrado um diagrama do *work flow* da aplicação.

Utilização do Network

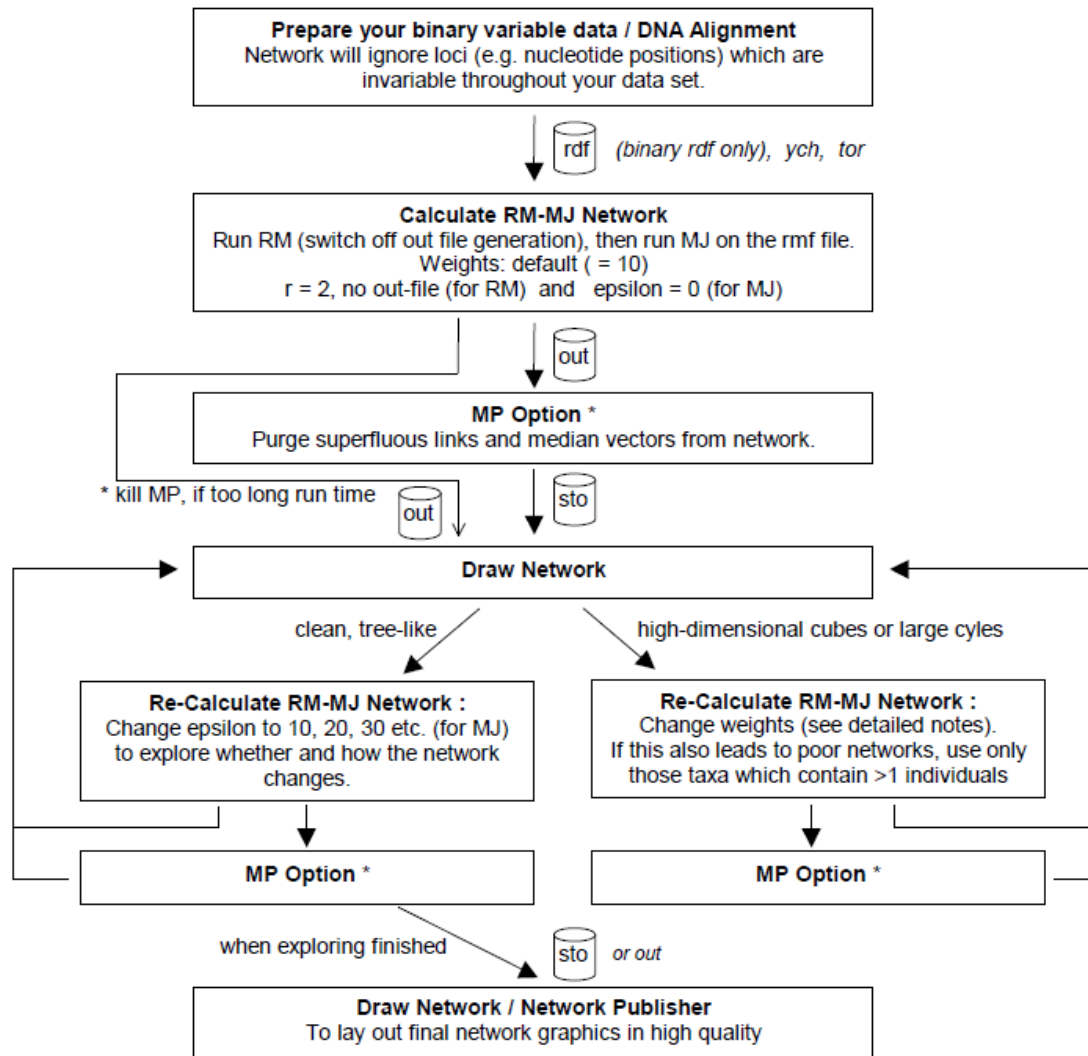


Figura A.9: Work flow do Network 4.6. ¹⁵

A partir deste diagrama acima apresentado é possível compreender melhor o fluxo de trabalho do Network 4.6. Resta acrescentar que no final de todo este processo, é possível guardar o resultado final em formato fdi, que é assim que este é criado, e ainda é possível exportar para formatos de imagem, tais como pdf ou bmp.

¹⁵ Fluxus Technology Ltd. 2011. Network 4.6 User Guide. Disponível em http://www.fluxus-engineering.com/Network4600_user_guide.pdf Última consulta em 9 de Fevereiro de 2011.

A.2 Formato RDF

Apresenta-se de seguida o ficheiro no formato rdf utilizado para este exemplo.

[illegible]

```

>SEC2;2;SEC;;;;;
000000001000000010000000000010000011010
>SWC1;1;SWC;;;;;
000000000000000000000000000010000010010
>SWC2;1;SWC;;;;;
000000000000000010000000000010000010110
>SWC3;1;SWC;;;;;
001001100000000010000000000010100010010
>SWC4;5;SWC;;;;;
001000100000000010000000000010100010010
>SWC5;1;SWC;;;;;
000000100000000010000000000010100010010
>SWC6;1;SWC;;;;;
101000100000000011000000000010100010010
>SWC7;1;SWC;;;;;
0000000000000000000010000000000010100010110
>SWC8;1;SWC;;;;;
000000000000100101000000000010100010010
>SWC9;1;SWC;;;;;
000000000000100001000000000010100010010
>SWC10;1;SWC;;;;;
000000001000000010000000000010000011000
>SWC11;1;SWC;;;;;
0000001000000000010000010000010000010010
>SWC12;1;SWC;;;;;
0100000000000000010001000010010000011010
>TAI1;2;TAI;;;;;
001000100000000001000000000010000010010
>TAI2;1;TAI;;;;;
0000000000000000010000100000010100010010
>TAI3;3;TAI;;;;;
000000000000000001000000000010000010010
>TAI4;3;TAI;;;;;
010000000000000001000000000010000010010
>TIB1;1;TIB;;;;;
000010100000010001000000000100000010010
>TIB2;1;TIB;;;;;
010000000000000001000000000010000010010
>TIB3;1;TIB;;;;;
010000000000000001010000000010000000010
>TIB4;1;TIB;;;;;
000000000000000001000000000010000010111

```

A.3 Formato OUT

Apresenta-se de seguida o ficheiro no formato out utilizado para este exemplo.

Results of file F:\New FA\Networks\Haplogroup M\CZ\C5\C5.rdf with reduction threshold=1.40000000000000E+0000

Equivalent taxa :

Tax.	14 with tax.	15
Tax.	11 with tax.	16
Tax.	1 with tax.	18
Tax.	1 with tax.	26
Tax.	1 with tax.	32
Tax.	7 with tax.	19
Tax.	7 with tax.	34
Tax.	7 with tax.	41
Tax.	8 with tax.	21
Tax.	4 with tax.	28
Tax.	2 with tax.	29
Tax.	2 with tax.	31
Tax.	2 with tax.	39

Final links :

Link	1 1 Seq.	1 with Seq.	7, mutations at 1
Link	2 1 Seq.	1 with Seq. mv	1, mutations at 14
Link	3 0 Seq.	2 with Seq.	4, mutations at 4
Link	4 1 Seq.	3 with Seq.	4, mutations at 1
Link	5 1 Seq.	3 with Seq.	14, mutations at 3
Link	6 0 Seq.	4 with Seq.	9, mutations at 22
Link	7 1 Seq.	4 with Seq.	11, mutations at 3
Link	8 0 Seq.	4 with Seq.	37, mutations at 19
Link	9 0 Seq.	5 with Seq.	11, mutations at 12
Link	10 1 Seq.	6 with Seq.	14, mutations at 5
Link	11 1 Seq.	6 with Seq.	17, mutations at 1
Link	12 0 Seq.	7 with Seq.	12, mutations at 10
Link	13 1 Seq.	7 with Seq.	20, mutations at 7
Link	14 1 Seq.	7 with Seq. mv	2, mutations at 14
Link	15 1 Seq.	8 with Seq.	14, mutations at 26
Link	16 1 Seq.	8 with Seq.	30, mutations at 1
Link	17 0 Seq.	8 with Seq.	35, mutations at 18
Link	18 0 Seq.	10 with Seq.	14, mutations at 17
Link	19 1 Seq.	11 with Seq.	14, mutations at 1
Link	20 1 Seq.	11 with Seq.	17, mutations at 5
Link	21 1 Seq.	11 with Seq.	30, mutations at 26
Link	22 0 Seq.	11 with Seq.	38, mutations at 21
Link	23 0 Seq.	11 with Seq.	40, mutations at 23
Link	24 1 Seq.	11 with Seq. mv	1, mutations at 24
Link	25 0 Seq.	13 with Seq.	38, mutations at 27
Link	26 0 Seq.	14 with Seq.	33, mutations at 6
Link	27 1 Seq.	14 with Seq. mv	2, mutations at 24
Link	28 0 Seq.	20 with Seq.	22, mutations at 2
Link	29 0 Seq.	20 with Seq.	23, mutations at 15
Link	30 0 Seq.	20 with Seq.	24, mutations at 13
Link	31 0 Seq.	20 with Seq.	25, mutations at 9
Link	32 1 Seq.	20 with Seq.	27, mutations at 14
Link	33 1 Seq.	27 with Seq. mv	2, mutations at 7
Link	34 0 Seq.	36 with Seq.	38, mutations at 8
Link	35 1 Seq. mv	1 with Seq. mv	2, mutations at 1

25

Frequ. of taxon	1 : 12
Frequ. of taxon	2 : 2
Frequ. of taxon	3 : 2
Frequ. of taxon	4 : 30
Frequ. of taxon	5 : 1
Frequ. of taxon	6 : 1
Frequ. of taxon	7 : 4
Frequ. of taxon	8 : 7
Frequ. of taxon	9 : 1
Frequ. of taxon	10 : 3
Frequ. of taxon	11 : 1
Frequ. of taxon	12 : 3

Utilização do Network

```

Frequ. of taxon      13 :   1
Frequ. of taxon      14 :   2
Frequ. of taxon      15 :   1
Frequ. of taxon      16 :   1
Frequ. of taxon      17 :   1
Frequ. of taxon      18 :   1
Frequ. of taxon      19 :   3
Frequ. of taxon      20 :  41
Frequ. of taxon      21 :  17
Frequ. of taxon      22 :   2
Frequ. of taxon      23 :   2
Frequ. of taxon      24 :   1
Frequ. of taxon      25 :   1
Frequ. of taxon      26 :   6
Frequ. of taxon      27 :   1
Frequ. of taxon      28 :   8
Frequ. of taxon      29 :  15
Frequ. of taxon      30 :   1
Frequ. of taxon      31 :   1
Frequ. of taxon      32 :   1
Frequ. of taxon      33 :   1
Frequ. of taxon      34 :   1
Frequ. of taxon      35 :   1
Frequ. of taxon      36 :   1
Frequ. of taxon      37 :   2
Frequ. of taxon      38 :   1
Frequ. of taxon      39 :   1
Frequ. of taxon      40 :   1
Frequ. of taxon      41 :   1
Frequ. of taxon      mv  1 :   0
Frequ. of taxon      mv  2 :   0
*****
Weight of character    1 :   5
Weight of character    2 :  10
Weight of character    3 :  16
Weight of character    4 :  10
Weight of character    5 :  10
Weight of character    6 :  10
Weight of character    7 :   5
Weight of character    8 :  10
Weight of character    9 :  10
Weight of character   10 :  10
Weight of character   12 :  10
Weight of character   13 :  10
Weight of character   14 :  16
Weight of character   15 :  10
Weight of character   17 :  10
Weight of character   18 :  10
Weight of character   19 :  10
Weight of character   21 :  10
Weight of character   22 :  10
Weight of character   23 :   5
Weight of character   24 :  10
Weight of character   25 :  10
Weight of character   26 :  10
Weight of character   27 :   5
*****
Mapping of character;   1;+93
Mapping of character;   2;+127
Mapping of character;   3;+148
Mapping of character;   4;+164
Mapping of character;   5;+174
Mapping of character;   6;+179
Mapping of character;   7;+189
Mapping of character;   8;+193
Mapping of character;   9;+197
Mapping of character;  10;+214
Mapping of character;  11;+223
Mapping of character;  12;+234
Mapping of character;  13;+260
Mapping of character;  14;+261
Mapping of character;  15;+266
Mapping of character;  16;+288
Mapping of character;  17;+291

```

Utilização do Network

Mapping of character; 18;+294
 Mapping of character; 19;+297A
 Mapping of character; 20;+298
 Mapping of character; 21;+300
 Mapping of character; 22;+301
 Mapping of character; 23;+311
 Mapping of character; 24;+327
 Mapping of character; 25;+357
 Mapping of character; 26;+390
 Mapping of character; 27;+189
 Mapping of taxon; 1;CAS1
 Other information of taxon;
 1;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 2;CAS2
 Other information of taxon;
 2;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 3;CAS3
 Other information of taxon;
 3;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 4;CAS4
 Other information of taxon;
 4;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 5;CAS5
 Other information of taxon;
 5;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 6;CAS6
 Other information of taxon;
 6;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 7;CAS7
 Other information of taxon;
 7;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 8;CAS8
 Other information of taxon;
 8;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 9;CAS9
 Other information of taxon;
 9;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 10;CAS10
 Other information of taxon;
 10;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 11;CAS11
 Other information of taxon;
 11;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 12;CAS12
 Other information of taxon;
 12;Phenotype;CAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 13;CCH1
 Other information of taxon;
 13;Phenotype;CCH;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 14;CCH2
 Other information of taxon;
 14;Phenotype;CCH;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 15;JAP1
 Other information of taxon;
 15;Phenotype;JAP;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 16;KOR1
 Other information of taxon;
 16;Phenotype;KOR;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 17;KOR2
 Other information of taxon;
 17;Phenotype;KOR;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 18;KOR3
 Other information of taxon;
 18;Phenotype;KOR;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 19;NAS1
 Other information of taxon;
 19;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 20;NAS2
 Other information of taxon;
 20;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 21;NAS3
 Other information of taxon;
 21;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 22;NAS4

Utilização do Network

Other information of taxon;
 22;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 23;NAS5
 Other information of taxon;
 23;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 24;NAS6
 Other information of taxon;
 24;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 25;NAS7
 Other information of taxon;
 25;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 26;NAS8
 Other information of taxon;
 26;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 27;NAS9
 Other information of taxon;
 27;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 28;NAS10
 Other information of taxon;
 28;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 29;NAS11
 Other information of taxon;
 29;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 30;NAS12
 Other information of taxon;
 30;Phenotype;NAS;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 31;NCM1
 Other information of taxon;
 31;Phenotype;NCM;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 32;NCM2
 Other information of taxon;
 32;Phenotype;NCM;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 33;NCM3
 Other information of taxon;
 33;Phenotype;NCM;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 34;NCM4
 Other information of taxon;
 34;Phenotype;NCM;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 35;NCM5
 Other information of taxon;
 35;Phenotype;NCM;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 36;SEC1
 Other information of taxon;
 36;Phenotype;SEC;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 37;SEC2
 Other information of taxon;
 37;Phenotype;SEC;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 38;SEC3
 Other information of taxon;
 38;Phenotype;SEC;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 39;SEC4
 Other information of taxon;
 39;Phenotype;SEC;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 40;TAI1
 Other information of taxon;
 40;Phenotype;TAI;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon; 41;TIB1
 Other information of taxon;
 41;Phenotype;TIB;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon;mv 1;mv 1
 Other information of taxon;mv
 1;Phenotype;;Geography;;Lineage;;Group1;;Group2;;Group3;;
 Mapping of taxon;mv 2;mv 2
 Other information of taxon;mv
 2;Phenotype;;Geography;;Lineage;;Group1;;Group2;;Group3;;

A.4 Formato FDI

Apresenta-se de seguida um exemplo de um ficheiro no formato fdi.

```

SHOW_NODENAMES;TRUE
NODES_PROPORTIONAL;TRUE
SHOW_ONLY_TORSO;FALSE
SHOW_MEDIANS;TRUE
SHOW_CHARACTERS;TRUE
MIN_CIRC_RADIUS;4
MAX_CIRC_RADIUS;50
MED_RADIUS;3

SPECIAL_FILE_TYPE;FALSE
DIAGRAM_CENTER;585;389
IMAGE_CENTER;585;389
NUMBER_OF_TAXA;22
TAXON_NAME;BUR1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;2;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;BUR;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;111;TAXON_Y;490;TAXON_COLOR_PIE1;16777215;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_COLOR_PIE2;32768;TAXON_PIE_FREQUENCY2;1;TAXON_STYLE_PIE2;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE
TAXON_NAME;CCH1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;1;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;CCH;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;818;TAXON_Y;173;TAXON_COLOR_PIE1;16744448;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE
TAXON_NAME;CCH2;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;1;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;CCH;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;978;TAXON_Y;147;TAXON_COLOR_PIE1;16744448;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE
TAXON_NAME;ECH1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;1;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;ECH;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;522;TAXON_Y;305;TAXON_COLOR_PIE1;65535;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE
TAXON_NAME;IND1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;4;TAXON_ORIG_FREQUENCY;2;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;IND;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;270;TAXON_Y;424;TAXON_COLOR_PIE1;0;TAXON_PIE_FREQUENCY1;2;TAXON_STYLE_PIE1;SOLID;TAXON_COLOR_PIE2;8454016;TAXON_PIE_FREQUENCY2;2;TAXON_STYLE_PIE2;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE
TAXON_NAME;IND2;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;104;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;IND;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;431;TAXON_Y;439;TAXON_COLOR_PIE1;0;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_COLOR_PIE2;16711680;TAXON_PIE_FREQUENCY2;5;TAXON_STYLE_PIE2;SOLID;TAXON_COLOR_PIE3;33023;TAXON_PIE_FREQUENCY3;98;TAXON_STYLE_PIE3;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE
TAXON_NAME;JAP1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;1;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;JAP;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;980;TAXON_Y;188;TAXON_COLOR_PIE1;255;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE
TAXON_NAME;MAL1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;2;TAXON_ORIG_FREQUENCY;2;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;MAL;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;0;TAXON_Y;0;TAXON_COLOR_PIE1;8454016;TAXON_PIE_FREQUENCY1;2;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;FALSE
TAXON_NAME;NCM1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;1;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;NCM;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;417;TAXON_Y;181;TAXON_COLOR_PIE1;16776960;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE
TAXON_NAME;NMS1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;1;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;NMS;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;0;TAXON_Y;0;TAXON_COLOR_PIE1;32768;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;FALSE

```

Utilização do Network

TAXON_NAME;PHI1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;3;TAXON_ORIG_FREQUENCY;2;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;PHI;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;282;TAXON_Y;503;TAXON_COLOR_PIE1;8421504;TAXON_PIE_FREQUENCY1;2;TAXON_STYLE_PIE1;SOLID;TAXON_COLOR_PIE2;33023;TAXON_PIE_FREQUENCY2;1;TAXON_STYLE_PIE2;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE

TAXON_NAME;SEC1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;5;TAXON_ORIG_FREQUENCY;5;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;SEC;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;0;TAXON_Y;0;TAXON_COLOR_PIE1;16711680;TAXON_PIE_FREQUENCY1;5;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;FALSE

TAXON_NAME;SEC2;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;1;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;SEC;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;308;TAXON_Y;544;TAXON_COLOR_PIE1;16711680;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE

TAXON_NAME;SEC3;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;2;TAXON_ORIG_FREQUENCY;2;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;SEC;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;740;TAXON_Y;347;TAXON_COLOR_PIE1;16711680;TAXON_PIE_FREQUENCY1;2;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE

TAXON_NAME;SEC4;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;2;TAXON_ORIG_FREQUENCY;2;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;SEC;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;902;TAXON_Y;343;TAXON_COLOR_PIE1;16711680;TAXON_PIE_FREQUENCY1;2;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE

TAXON_NAME;SEC5;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;1;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;SEC;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;580;TAXON_Y;375;TAXON_COLOR_PIE1;16711680;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE

TAXON_NAME;TAI1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;98;TAXON_ORIG_FREQUENCY;98;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;TAI;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;0;TAXON_Y;0;TAXON_COLOR_PIE1;33023;TAXON_PIE_FREQUENCY1;98;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;FALSE

TAXON_NAME;TAI2;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;2;TAXON_ORIG_FREQUENCY;2;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;TAI;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;281;TAXON_Y;377;TAXON_COLOR_PIE1;33023;TAXON_PIE_FREQUENCY1;2;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE

TAXON_NAME;TAI3;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;1;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;TAI;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;0;TAXON_Y;0;TAXON_COLOR_PIE1;33023;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;FALSE

TAXON_NAME;TAI4;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;2;TAXON_ORIG_FREQUENCY;2;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;TAI;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;351;TAXON_Y;580;TAXON_COLOR_PIE1;33023;TAXON_PIE_FREQUENCY1;2;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE

TAXON_NAME;TAI5;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;1;TAXON_ORIG_FREQUENCY;1;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;TAI;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;312;TAXON_Y;330;TAXON_COLOR_PIE1;33023;TAXON_PIE_FREQUENCY1;1;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE

TAXON_NAME;mv
1;TAXON_NAME_HEIGHT;6;TAXON_NAME_COLOR;0;TAXON_FREQUENCY;0;TAXON_ORIG_FREQUENCY;0;TAXON_GEOGRAPHY;;TAXON_PHENOTYPE;;TAXON_LINEAGE;;TAXON_GROUP1;Group1;;TAXON_GROUP2;Group2;;TAXON_GROUP3;Group3;;TAXON_X;671;TAXON_Y;241;TAXON_COLOR_PIE1;255;TAXON_PIE_FREQUENCY1;0;TAXON_STYLE_PIE1;SOLID;TAXON_LINE_WIDTH;1;TAXON_LINE_COLOR;0;TAXON_LINE_STYLE;SOLID;TAXON_ACTIVE;TRUE

NUMBER_OF_LINKS;17
LINK_TAXON1;BUR1;LINK_TAXON2;IND2;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;2;MUTATION1_NAME;+289
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;MUTATION2_NAME;+311
;MUTATION2_NAME_HEIGHT;6;MUTATION2_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;CCH1;LINK_TAXON2;CCH2;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+354
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE

Utilização do Network

```

LINK_TAXON1;CCH1;LINK_TAXON2;JAP1;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+162
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;CCH1;LINK_TAXON2;mv
1;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+311
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;ECH1;LINK_TAXON2;IND2;LINK_IN_TORSO;TRUE;LINK_TORSO_PART;1;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+172
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;16777215;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;16777215;LINK_ACTIVE;TRUE
LINK_TAXON1;ECH1;LINK_TAXON2;NCM1;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+264
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;ECH1;LINK_TAXON2;mv
1;LINK_IN_TORSO;TRUE;LINK_TORSO_PART;1;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+129
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;IND1;LINK_TAXON2;IND2;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+170
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;IND2;LINK_TAXON2;PHI1;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+241
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;IND2;LINK_TAXON2;SEC2;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+266
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;IND2;LINK_TAXON2;SEC5;LINK_IN_TORSO;TRUE;LINK_TORSO_PART;1;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+129
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;IND2;LINK_TAXON2;TAI2;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+69
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;IND2;LINK_TAXON2;TAI4;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+304
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;IND2;LINK_TAXON2;TAI5;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+274
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;SEC3;LINK_TAXON2;SEC4;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+355
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;SEC3;LINK_TAXON2;SEC5;LINK_IN_TORSO;FALSE;LINK_TORSO_PART;0;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+265
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE
LINK_TAXON1;SEC5;LINK_TAXON2;mv
1;LINK_IN_TORSO;TRUE;LINK_TORSO_PART;1;NUMBER_OF_MUTATIONS;1;MUTATION1_NAME;+172
;MUTATION1_NAME_HEIGHT;6;MUTATION1_NAME_COLOR;255;LINK_LINE_WIDTH;1;LINK_LINE_STYLE;SOLID;LINK_LINE_COLOR;8421504;LINK_ACTIVE;TRUE

NUMBER_OF_HELP_POINT_LINKS;0

LONG_EDGES;0
TOTAL_TAXA;129
TOTAL_HAPLOTYPES;16
NUMBER_EQUIVALENT_TAXA;5
EQUIVALENT_TAXA;IND1;MAL1
EQUIVALENT_TAXA;BUR1;NMS1
EQUIVALENT_TAXA;IND2;SEC1
EQUIVALENT_TAXA;IND2;TAI1
EQUIVALENT_TAXA;PHI1;TAI3
COLOR_SCHEME;SCHEME_ACTIVE;TRUE;SCHEME_TYPE;Phenotype;SCHEME_ITEM;BUR;ITEM_COLOR;16777215;SCHEME_ITEM;CCH;ITEM_COLOR;16744448;SCHEME_ITEM;ECH;ITEM_COLOR;65535;SCHEME_ITEM;IND;I

```

Utilização do Network

ITEM_COLOR;0;SCHEME_ITEM;JAP;ITEM_COLOR;255;SCHEME_ITEM;MAL;ITEM_COLOR;8454016;SCHEME_ITEM;NCM;ITEM_COLOR;16776960;SCHEME_ITEM;NMS;ITEM_COLOR;32768;SCHEME_ITEM;PHI;ITEM_COLOR;8421504;SCHEME_ITEM;SEC;ITEM_COLOR;16711680;SCHEME_ITEM;TAI;ITEM_COLOR;33023;COLOR_SCHEME;SCHEME_ACTIVE;FALSE;SCHEME_TYPE;Geography;COLOR_SCHEME;SCHEME_ACTIVE;FALSE;SCHEME_TYPE;Lineage;COLOR_SCHEME;SCHEME_ACTIVE;FALSE;SCHEME_TYPE;Group1;COLOR_SCHEME;SCHEME_ACTIVE;FALSE;SCHEME_TYPE;Group2;COLOR_SCHEME;SCHEME_ACTIVE;FALSE;SCHEME_TYPE;Group3;

Anexo B

Detalhes da Aplicação

B.1 Guia de Utilização

Nesta secção apresenta-se um guia de utilização da aplicação desenvolvida. Começa-se por apresentar o menu inicial, na Figura B.1, com as quatro ferramentas principais: obtenção de árvores a partir de redes filogenéticas; correcções nas árvores obtidas (escolha de arestas alternativas); eliminar *median vectors* desnecessários (com menos de três arestas); cálculo de estatísticas.

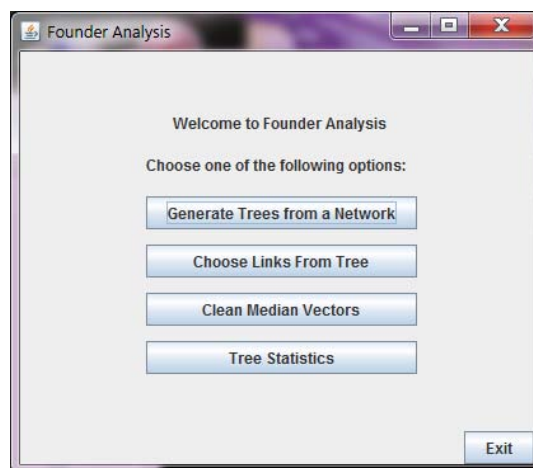


Figura B.1: Menu inicial da aplicação.

Para utilizar a primeira ferramenta, de geração de árvores a partir de uma rede, basta clicar no primeiro botão. Abrir-se-á uma nova janela, representada na Figura B.2, que permitirá escolher um ficheiro, no formato fdi ou out, que contenha uma rede filogenética.

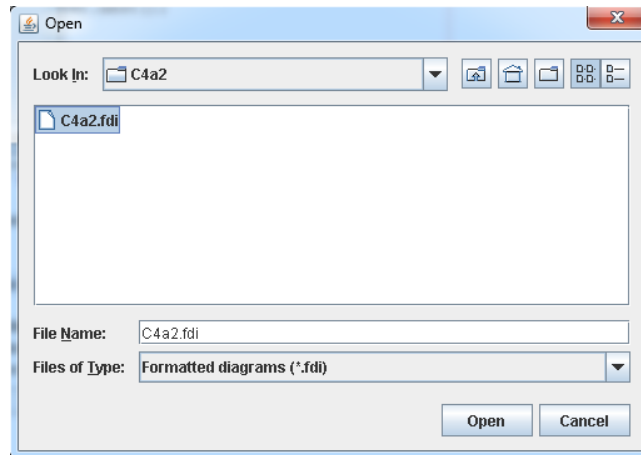


Figura B.2: Selecção do ficheiro que contém a rede.

Escolhido o ficheiro, a ferramenta irá obter os dados da rede e na janela seguinte, representada na Figura B.3, é possível escolher a raiz da rede, o algoritmo a utilizar para calcular as árvores, o número de árvores pretendido e o custo total da árvore permitido.

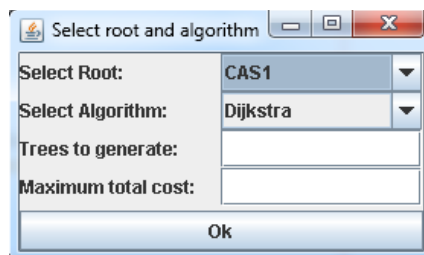


Figura B.3: Menu de parâmetros da ferramenta de geração de árvores.

Concluído este processo, serão gerados vários ficheiros – out ou fdi, dependendo do original – com as árvores obtidas, a janela fecha-se e estaremos de volta ao menu inicial, representado na Figura B.1.

Com a segunda ferramenta é possível o especialista escolher manualmente quais as arestas que se mantêm na árvore e as que deverão ser retiradas. Para tal, após carregar no botão “Choose Links From Tree” terá de seleccionar os ficheiros fdi ou out da árvore e da rede original, através de uma interface semelhante à da Figura B.2. De seguida aparecerá uma janela apresentada na Figura B.4, onde aparecem todas as arestas da rede original e uma *checkbox* à frente de cada. As arestas já existentes na árvore terão a *checkbox* activa, enquanto as restantes terão a *checkbox* inactiva. Após seleccionar e carregar no botão “Ok” o ficheiro no formato fdi ou out será alterado conforme as escolhas.

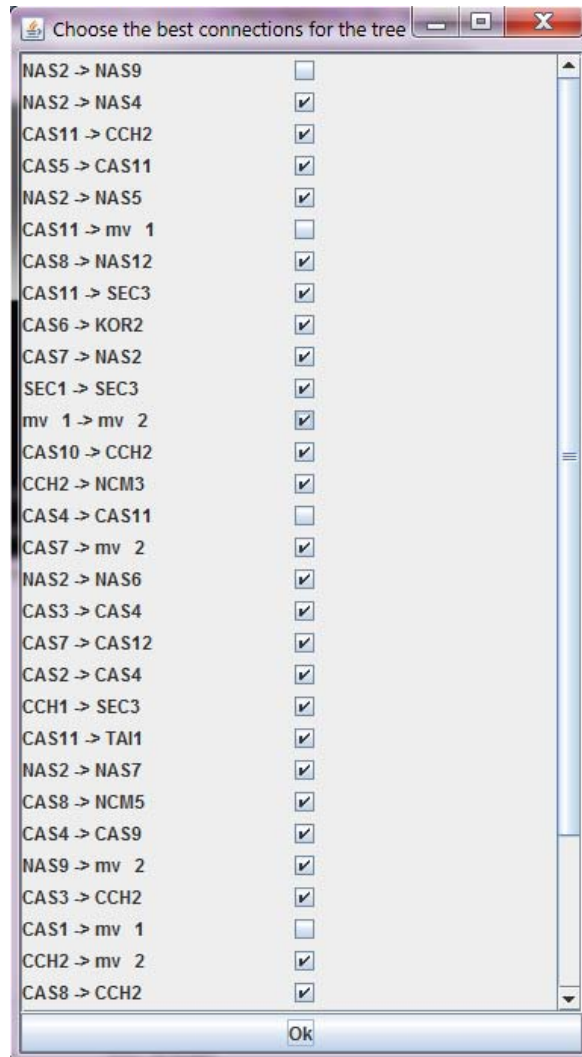


Figura B.4: Escolha das arestas da árvore de forma manual.

A terceira ferramenta serve para eliminar de um ficheiro fdi ou out os *median vectors* desnecessários, ou seja, tal como abordado na Secção 4.4, *median vectors* com menos de três arestas, que não têm razão de existir numa árvore filogenética. Após ser escolhido o ficheiro, e ser escolhida a raiz da árvore, é criado um novo ficheiro sem os *median vectors* desnecessários.

Para utilizar a quarta ferramenta, de cálculo de estatísticas, basta clicar no segundo botão do menu inicial e de seguida aparecerá uma janela, semelhante à da Figura B.2, mas desta vez, tem de ser introduzido um ficheiro (fdi ou out) com uma árvore e não com uma rede filogenética (Figura B.5).

Detalhes da Aplicação

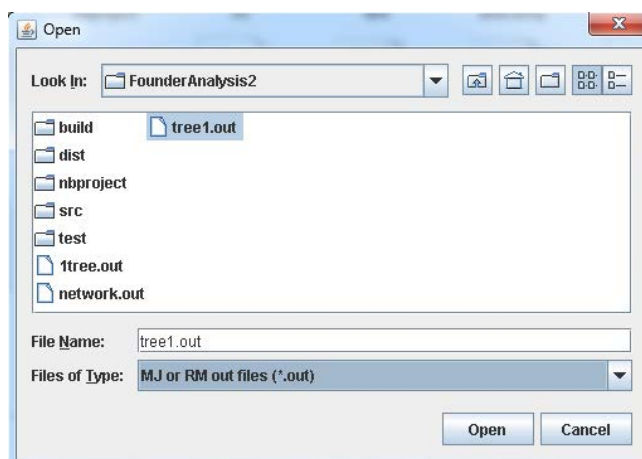


Figura B.5: Selecção do ficheiro que contém a árvore.

De seguida, aparecerá uma janela para serem especificados os parâmetros necessários (Figura B.6), ou seja, regiões ascendentes, descendentes e indeterminadas, critério f a utilizar e raiz da árvore. Na Secção 5.2 foram apresentadas várias captações desta janela para cada caso de teste.

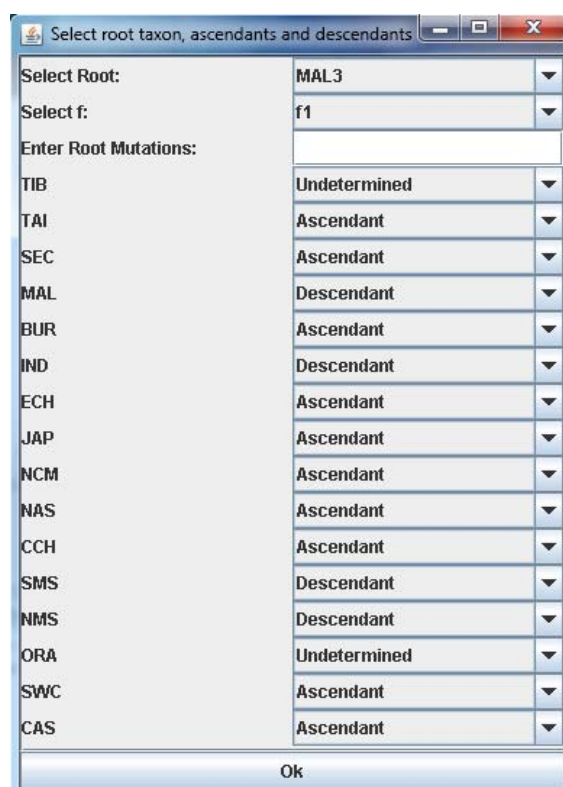
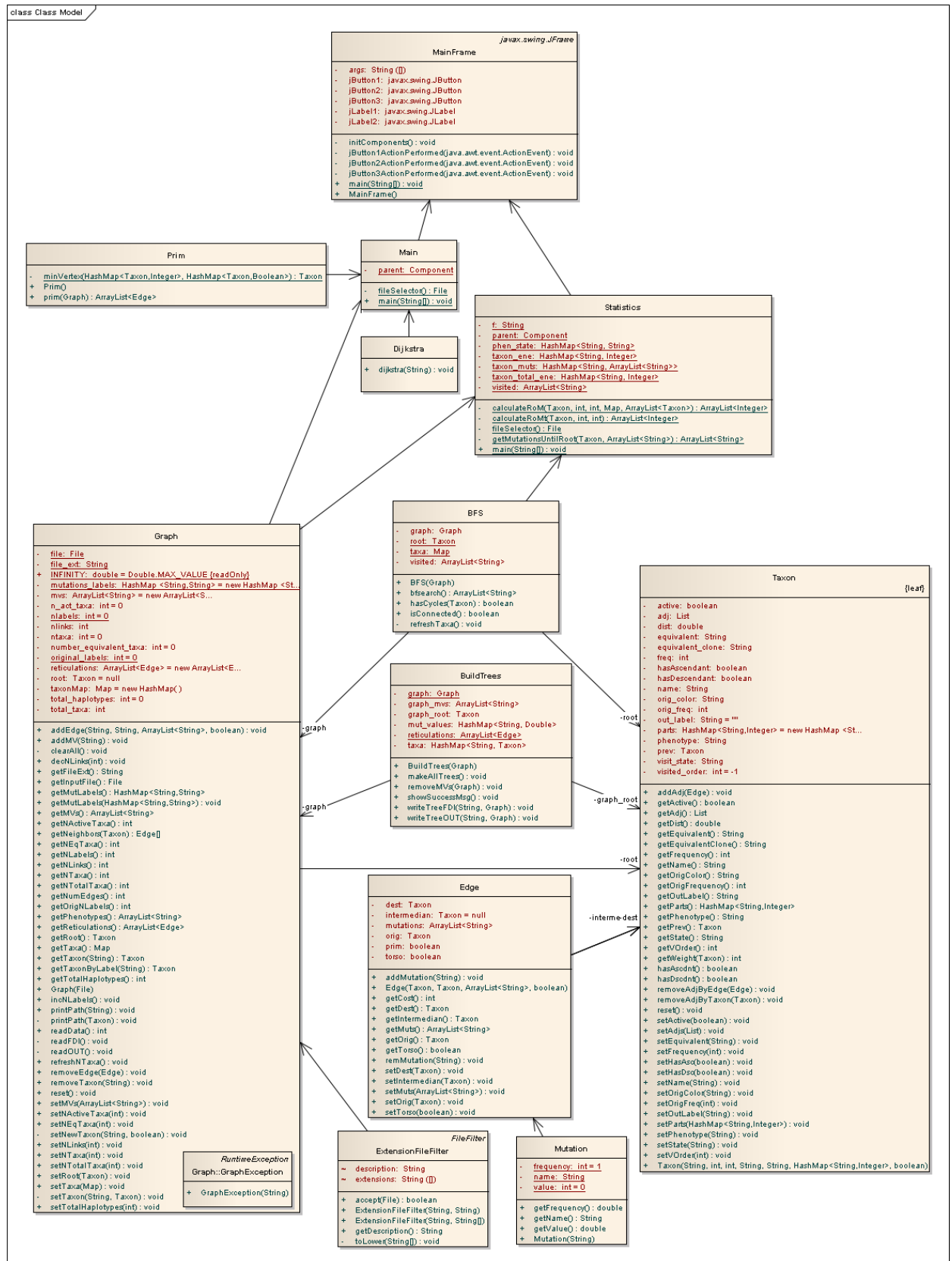


Figura B.6: Menu de parâmetros da ferramenta de cálculo de estatísticas.

B.2 Diagrama de Classes



B.3 Ficheiro Auxiliar com Frequências das Mutações

O ficheiro com os valores da frequência de cada uma das mutações é aqui apresentado. Em cada linha tem o nome da mutação e a seguir o valor da sua frequência. Quanto maior for este valor, mais frequente é a mutação, logo no grafo terá menor custo em relação a outras com menor frequência. Caso exista em alguma rede filogenética uma mutação que não exista neste ficheiro, considera-se que o seu valor de frequência é 1. Este ficheiro foi extraído da Tabela S3 de Soares et al., 2009 [SET09], mas como apenas são utilizadas as mutações presentes na Região de Controlo (CR), mais concretamente na Região Hipervariável I (HVS-I) nas posições entre 16051 até 16400, as restantes foram eliminadas deste ficheiro. Para simplificação, removeu-se o prefixo 16 da numeração, assim logo nos primeiros dois casos, +311 representa a mutação presente na posição 16311 e +189 a mutação presente na posição 16189. Algumas posições têm uma letra à frente (ex: +318T) porque pode ocorrer mais do que um tipo de mutação nessa posição. O ficheiro é apresentado em três colunas, para não ocupar demasiado espaço.

+311 120	+288 11	+343 5
+189 90	+287 11	+317 5
+129 86	+260 11	+297 5
+93 79	+213 11	+293C 5
+362 67	+193 11	+286 5
+278 43	+185 11	+284 5
+172 42	+324 10	+265 5
+223 37	+295 10	+245 5
+319 35	+290 10	+241 5
+291 34	+270 10	+222 5
+192 33	+235 10	+219 5
+390 31	+188 10	+173 5
+209 31	+168 10	+154 5
+86 30	+271 9	+153 5
+294 29	+263 9	+147 5
+256 29	+239 9	+391 4
+111 27	+140 9	+366 4
+399 26	+360 8	+359 4
+274 26	+318T 8	+342 4
+234 26	+265C 8	+316 4
+304 24	+243 8	+301 4
+261 22	+124 8	+296 4
+145 22	+352 7	+254 4
+51 21	+344 7	+244 4
+249 20	+327 7	+231 4
+126 20	+264 7	+215 4
+356 19	+255 7	+157 4
+320 19	+242 7	+150 4
+187 18	+227 7	+114A 4
+176 18	+218 7	+400 3
+293 17	+174 7	+381 3
+266 17	+335 6	+354 3
+92 17	+300 6	+302 3
+325 16	+299 6	+272 3
+148 16	+259 6	+269 3
+309 14	+257 6	+265T 3
+292 14	+240 6	+258C 3
+526 13	+217 6	+248 3
+298 13	+214 6	+230 3
+368 12	+179 6	+212 3
+357 12	+167 6	+186 3
+355 12	+162 6	+180 3
+224 12	+136 6	+166 3
+184 12	+75 6	+164 3
+169 12	+66 6	+146 3

Detalhes da Aplicação

+144 3	+346 1	+220C 1
+134 3	+343C 1	+220 1
+104 3	+337 1	+216C 1
+95 3	+336 1	+215T 1
+398 2	+330 1	+210 1
+327A 2	+326 1	+207T 1
+318C 2	+323 1	+206C 1
+291G 2	+320A 1	+204 1
+266G 2	+318A 1	+203 1
+250 2	+318 1	+194 1
+247 2	+317T 1	+190 1
+246T 2	+312 1	+188G 1
+233 2	+310 1	+184A 1
+216 2	+305 1	+182 1
+207 2	+303 1	+176G 1
+201 2	+292G 1	+171 1
+195 2	+289 1	+170C 1
+183 2	+286G 1	+170 1
+182T 2	+275 1	+169A 1
+181 2	+268 1	+164T 1
+178 2	+266A 1	+160 1
+175 2	+262 1	+156 1
+166C 2	+259A 1	+155 1
+163 2	+258T 1	+144A 1
+159 2	+258 1	+142 1
+158 2	+257A 1	+133 1
+137 2	+256A 1	+131 1
+111A 2	+254C 1	+129C 1
+94 2	+252C 1	+119 1
+69 2	+245A 1	+114 1
+67 2	+242A 1	+109C 1
+380 1	+241T 1	+108 1
+374C 1	+240C 1	+102 1
+371 1	+239G 1	+91 1
+365 1	+232A 1	+83 1
+362A 1	+232 1	+82 1
+360A 1	+229 1	+77T 1
+355A 1	+221A 1	+71 1
+353 1	+221 1	

B.4 Ficheiro Auxiliar com Informação das Populações

O ficheiro auxiliar phenotypes.txt é aqui apresentado. Em cada linha está a sigla da região de cada população e de seguida a informação sobre se é uma população ascendente (asc), descendente (dsc) ou indeterminada (und).

AEA -und
ANA -asc
ANC -und
ANE -und
ANW -und
ARC -asc
ARE -asc
ARM -asc
ARS -asc
BED -und
CAN -dsc
CAS -dsc
EAE -und
EAL -dsc
EBR -dsc
EIC -dsc
EMC -dsc
EME -dsc
EMW -dsc
ENC -dsc
ENE -dsc
ENW -dsc
ESC -dsc
ESE -dsc
JAZ -und
JNA -und
NEI -asc
NEL -asc
NEM -asc
SAS -und