Roberta Vilhena Vieira

Sistema Neurosimbólico para Construção de Árvores Filogenéticas

Campina Grande
1997

Roberta Vilhena Vieira

Sistema Neurosimbólico para Construção de Árvores Filogenéticas

Dissertação submetida ao curso de Pós-Graduação em Informática do Centro de Ciências e Tecnologia da Universidade Federal da Paraíba, como requisito parcial para a obtenção do grau de mestre em Informática.

Área de concentração: Inteligência Artificial

Orientador: Prof. José Homero Feitosa Cavalcanti

Campina Grande, Maio de 1997

V675s Vieira, Roberta Vilhena
Sistema neurosimbolico para construcao de arvores filogeneticas / Roberta Vilhena Vieira. - Campina Grande : 1997.
84 f. : il.

Dissertacao (Mestrado em Informatica) - Universidade Federal da Paraiba, Centro de Ciencias e Tecnologia.

1. Inteligencia Artificial 2. Redes Neurais de Hopfield 3. Sistema Hibrido 4. Dissertacao I. Cavalcanti, Jose Homero Feitosa, Dr. II. Universidade Federal da Paraiba - Campina Grande - (PB) III. Título

CDU 004.8(043)

SISTEMA NEURO-SIMBÓLICO PARA A CONSTRUÇÃO DE ÁRVORES FILOGENÉTICAS

ROBERTA VILHENA VIEIRA

DISSERTAÇÃO APROVADA EM 26.05.1997

PROF. JOSÉ HOMERO FEITOSA CAVALCANTI, D. SC.
Presidente

PROF. GUILHERME BITTENCOURT, DR.
Examinador

PROF. MANOEL AGAMEMNON LOPES, D. SC.
Examinador

PROF. PABLO JAVIER ALSINA, D. SC.
Examinador

CAMPINA GRANDE - PB

A meus pais e irmãos.

Agradecimentos

Agradeço ao Prof. Homero Cavalcanti, que com sua competência, disponibilidade e amizade conquistou o meu respeito e admiração. Obrigada Homerinho pelos seus valorosos ensinamentos.

Agradeço ao Prof. Martin Christoffersen, que cedeu sua sala e seu tempo para acompanhar os meus estudos no campo da evolução morfológica.

Agradeço ao Prof. Horacio Schneider, que acompanhou os meu primeiros estudos no campo da evolução genética.

Agradeço ao Prof. Agamemnon pelas suas sugestões.

Agradeço ao DSC pela aceitação no curso de mestrado e pelo espaço físico cedido para o desenvolvimento deste trabalho.

Agradeço aos funcionários da COPIN e Miniblio pela ajuda, em especial a Verinha, Manuela e Zeneide.

Agradeço aos professores da COPIN pelos ensinamentos.

Agradeço as pessoas que conheci em Campina Grande com quem compartilhei momentos agradáveis, em especial a Vana, a Adna, a Sônia, a Dandara, ao Mário Ernesto, ao Waltércio e ao Juracy.

Esta dissertação foi financiada pelas instituições do CNPq e CAPES.

Lista de Figuras

Lista de Tabelas

Simbologia

Caracteres latinos gregos:

F - estrutura quaternária de uma árvore filogenética

$\mathbb{X}$ - conjunto finito de espécies

$\mathbb{Y}$ - conjunto finito das características apresentadas pelos elementos de $\mathbb{X}$.

Φ - uma relação de ordem

Γ - função que mapeia um elemento de $\mathbb{X}$ em uma elemento do conjunto $\mathcal{P}(\mathbb{Y})$

$\mathcal{P}(\mathbb{Y})$ - é o conjunto das partes do conjunto $\mathbb{Y}$

G - estrutura binária de um grafo da árvore filogenética

a_k - arco de um grafo

Σ_k - cadeia de características

d_{ij} - elemento da matriz distância da i linha e j coluna

c_{ij} - elemento da matriz característica da i linha e j coluna

s_{ij} - elemento da matriz similaridade da i linha e j coluna

$\varnothing$ - conjunto vazio

AF_i - árvore filogenética construída pelo algoritmo da regra de inclusão e exclusão

Dist - é a medida de distância entre duas espécies calculada pelo algoritmo de Wagner

Comp - é a medida do comprimento de um ramo que liga duas espécies da árvore filogenética construída pelo algoritmo de Wagner

$Caract_A$ - vetor característica da espécie A

V - vetor que descreve a rede neural de Hopfield

V_j - valor da saída do neurônio i de uma rede neural de Hopfield

V^s - conjunto de estados que a rede neural de Hopfield pode aprender

Δ - pequeno incremento

L(.) - função lógica

T_{ij} - peso da conexão entre o neurônio i e o neurônio j

U_i - nível de ativação do neurônio i da rede neural de Hopfield

E - função de energia da rede neural de Hopfield

ΔE - variação da energia da rede neural de Hopfield

ΔV_i - variação da saída do neurônio i

I - vetor de entradas externas da rede neural de Hopfield

θ - vetor de entradas internas da rede neural de Hopfield

W - vetor de peso para as entradas externas da rede neural de Hopfield

Y_i - saída linear do neurônio i de uma rede neural de Hopfield

k - é o tempo discreto

1/RC - é a constante de tempo do circuito RC

h - é o valor do incremento de tempo

τ - é o valor de $\frac{1}{RC}$

β - é o valor de $h \times \frac{1}{RC}$

S(.) - é a função sigmóide

η - é a constante de declividade da função sigmóide

O_i - é igual ao valor da sigmóide de U_i

T(.) - é a função tangente hiperbólica

Abreviações:

DNA - Ácido Dexonucleico

RNA - Ácido Ribonucleico

SINCA - Sistema Neurosimbólico para Construição de Árvores filogenéticas

STP - série de transformação polarizada

RC - Resistor e Capacitor

ITER - número de iteração

ANCA - Algoritmo Neural para a Construção da Árvore filogenética

Sinais/Operadores:

-(menos)

+(mais)

×(multiplicação)

=(igual)

≠(diferença)

≅(aproximadamente)

>(maior que)

≥(maior ou igual)

<(menor)

≤(menor ou igual)

∃(quantificador existêncial)

$\forall$(quantificador universal)

$\sum$(somatória)

$\cap$(interseção)

$\subset$(esta contido)

$\not\subset$(não esta contido)

$\in$(pertence)

$\notin$(não pertence)

Sumário

Resumo

Este trabalho apresenta um sistema neurosimbólico para construção de árvores filogenéticas denominado SINCA. Neste sistema as técnicas simbólicas e conexionistas trabalham de maneira cooperativa. O módulo conexionista do SINCA usa uma rede neural de Hopfield para encontrar a menor distância entre os ramos da árvore filogenética, controlando a explosão combinatorial gerada pelo número de árvores filogenéticas possíveis para o conjunto de espécies investigadas. O módulo simbolista do SINCA usa um sistema especialista para construir árvores filogenéticas a partir do conhecimento, fornecido pelo usuário, das regras de sua base de conhecimento, e do conhecimento gerado pelo seu módulo conexionista. Apresenta-se um estudo de árvores filogenéticas, os principais algoritmos para construção de árvores filogenéticas, um estudo de redes neurais de Hopfield sua estabilidade e seus pesos, a implementação de um algoritmo neural para construção de árvores filogenéticas (ANCA) e um exemplo de construção de árvores filogenéticas com o ANCA. Finalmente, apresenta-se a implementação do SINCA, alguns exemplos de construção de árvores filogenéticas com o SINCA e são feitas sugestões de trabalhos futuros.

1 Introdução

1.1 A Evolução das Espécies

O estudo da evolução das espécies é uma das áreas mais antigas e importantes das ciências biológicas e tem servido de fonte de informação a diversos campos da Biologia, tais como: Biogeografia, Biologia Molecular, Farmacologia, etc.

Na Biogeografia as informações sobre a evolução das espécies permitem, fazer uma reconstituição das regiões geográficas que sofreram separação na época de sua formação através da análise da vicariância[1] e congruência entre padrões de distribuição de diferentes grupos de espécies. Assim, se uma certa fauna e flora é encontrada na América do Sul e na África, pode-se deduzir desse fato que no passado essas regiões eram ligadas por um bloco de terra [Amorim, 1994][Lawrence, 1951][Rizzini, 1979].

Na Biologia Molecular, as informações sobre a evolução das espécies são utilizadas como um conhecimento adicional para auxiliar na compreensão da origem e diferenciação das macromoléculas (DNA e RNA), ajudando dessa forma o trabalho do geneticista na decodificação do código genético (projeto GENOMA), ou melhor na busca para determinar a seqüência de bases nucleotídicas responsável por uma determinada característica [Amorim, 1994][Clement, 1986].

Na Farmacologia, as informações sobre a evolução das espécies têm direcionado os testes de reação imunológica, ou de toxidade no combate dos sintomas apresentados por uma

[1] Vicariância - Mecanismo de evolução das espécies em que a distribuição de uma espécie ancestral é fragmentada em duas ou mais áreas, as quais se apresentam isoladas geograficamente por uma barreira. Desse modo, a disjunção observada entre as partes isoladas é o resultado do movimento de uma parte de uma população para uma área que não era habitada, superando uma barreira existente, seguida pelo aparecimento de uma barreira que isola a reprodução dessas populações. Nesses casos, a soma das áreas atualmente disjuntas deverá resultar com maior ou menor precisão na distribuição geográfica original da espécie ou população ancestral.

família de vírus, de modo que, esses testes tenham um alcance mais amplo e resultados mais previsíveis [Amorim, 1994].

1.2 Motivação da Dissertação

A confiança atribuída a uma dada informação, sobre a evolução das espécies, é diretamente proporcional ao volume de dados analisados na construção dessa informação. Essa proporcionalidade justifica o desenvolvimento de um método de construção de informação, sobre a evolução das espécies que possa trabalhar com grandes volumes de dados. Tal requisito dificultou a manipulação dos dados e da informação sobre a evolução das espécies em uma representação textual. Por esse motivo, os dados passaram a ser representados em tabelas, e a informação sobre a evolução das espécies passou a ser representada graficamente por um diagrama ramificado, denominado árvore filogenética. Apesar dessas representações serem de fácil manipulação, a construção de árvores filogenéticas por processos manuais foi se tornando cada vez mais complexa, além do que, esses processos requerem muito tempo do filogeneticista e propiciam a introdução de erros humanos na análise. Essa situação perdurou até o início da década de 60, quando se passou a utilizar os primeiros computadores de cartão perfurado como uma ferramenta destinada a auxiliar o filogeneticista a construir a árvore filogenética das espécies [Amorim, 1994].

Os primeiros programas de computadores destinados a construir a árvore filogenética, foram baseados em métodos convencionais e receberam várias críticas devido à sua incapacidade em:

- fornecer uma explicação para as decisões tomadas pelo programa no tratamento de dados ambíguos e incompletos,
- manipular dados com diversas representações ao mesmo tempo,
- permitir a priorização de uma determinada relação de parentesco entre duas espécies quaisquer, e
- conter a explosão combinatorial gerada pela presença de dados ambíguos e incompletos.

Essas críticas resultaram no desenvolvimento de métodos baseados em Inteligência Artificial. Alguns desses métodos, citados em [Wiley, 1990], foram denominados de pesquisa exaustiva, ramifica-e-limita e busca heurística.

O método pesquisa exaustiva, inicialmente gera todas as possíveis árvores filogenéticas para o conjunto de espécies investigadas. Em seguida, será contado o número de dados utilizados na construção de cada ramo das árvores filogenéticas geradas. A árvore filogenética gerada, que tiver utilizado o menor número de dados, será a árvore filogenética solução obtida com este método.

O método ramifica-e-limita, inicialmente gera uma árvore filogenética com raiz igual a cada uma das espécies analisadas, de modo que cada novo ramo adicionado à árvore corresponda ao ramo que irá minimizar o número total de dados utilizados até o momento na construção da mesma. Em seguida, será contado o número de dados utilizados na construção de cada ramo das árvores filogenéticas geradas; a árvore filogenética gerada que tiver utilizado o menor número de dados será a árvore filogenética solução obtida com este método.

O método pesquisa heurística, inicialmente solicita ao usuário para selecionar uma das espécies analisadas como a raiz da árvore filogenética solução. Após o usuário fornecer o nome da espécie raiz, este método gera um conjunto de árvores filogenéticas, de modo que, cada novo ramo adicionado a uma dessas árvores, corresponda ao ramo que irá minimizar o número de dados utilizados até o momento na construção da árvore. Em seguida, será contado o número de dados utilizados na construção de cada ramo das árvores filogenéticas geradas. A árvore filogenética gerada que tiver utilizado o menor número de dados será a árvore filogenética solução.

As principais críticas feitas a esses métodos foram:

- o método pesquisa exaustiva e ramifica-e-limita são incapazes de construir uma árvore filogenéticas para uma dada família de espécies quando o volume de dados a ser analisado é grande;
- o método pesquisa heurística é incapaz de construir uma árvore filogenéticas para uma dada família de espécies quando o volume de dados a ser analisado é pequeno.

Essas críticas levaram Wiley [Wiley, 1990] a considerar que os métodos de construção de árvores filogenéticas pesquisa exaustiva, ramifica-e-limita e pesquisa heurística são ineficientes, já que esses métodos não podem ser aplicados sobre qualquer volume de dados.

1.3 Objetivos da Dissertação

Propor uma sistema neurosimbólico para construir a árvore filogenética de um conjunto de espécies que:

- forneça uma explicação para as decisões tomadas pelo sistema no tratamento de dados ambíguos e incompletos;
- manipule dados com diferentes representações;
- permita ao filogeneticista priorizar uma determinada relação de parentesco entre duas espécies quaisquer;
- gerencie a explosão combinatorial de todas as possíveis soluções geradas pela presença de dados ambíguos e incompletos;
- permita ao filogeneticista acompanhar o raciocínio desenvolvido pelo sistema;
- permita a construção de árvores filogenéticas pela análise de pequenos e grandes volume de dados.

1.4 Descrição da Dissertação

Neste trabalho será apresentado o desenvolvimento de um Sistema Neurosimbólico para Construção de Árvores filogenéticas (SINCA). No SINCA, a árvore filogenética será construída pelo trabalho cooperativo dos seus módulos simbólico e conexionista que se comunicam de modo bidirecional, como pode ser visto na Figura 1.1.

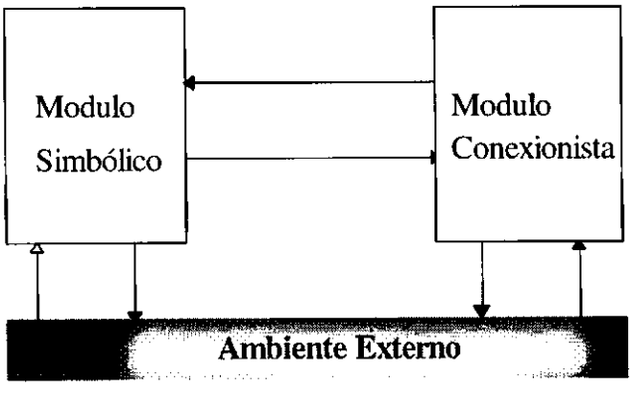


Figura 1.1 - Arquitetura do SINCA

O módulo simbólico do SINCA, é formado por um sistema especialista que tem a responsabilidade de construir a árvore filogenética pela combinação do conhecimento proveniente do módulo conexionista do SINCA, das regras contidas na base de conhecimento do SINCA, e da interação com o usuário. O conhecimento do módulo conexionista do SINCA é gerado a partir das informações fornecidas pelo módulo simbólico ou fornecidas pelo usuário sob a supervisão do módulo simbólico. O conhecimento proveniente das regras do sistema foi desenvolvido a partir das informações obtidas na literatura especializada dessa área. O conhecimento proveniente do usuário, diz respeito à experiência adquirida por este durante as suas investigações sobre a evolução do grupo de espécies em análise.

O módulo conexionista do SINCA é formado por uma rede neural de Hopfield, que é usada para construir todos os ramos possíveis da árvore filogenética a partir das características das espécies analisadas.

1.5 Organização da Dissertação

Para cumprir os objetivos definidos na Seção 1.3 deste capítulo, organizou-se este trabalho em cinco capítulos, incluindo esta introdução.

No capítulo 2, serão analisadas algumas matrizes utilizadas pelos filogeneticistas para armazenar os dados disponíveis sobre a evolução das espécies investigadas. A seguir, serão apresentados alguns algoritmos de construção de árvores filogenéticas e resultados experimentais obtidos com esses algoritmos. Por último, serão mostrados alguns problemas gerados pela presença de dados incompletos e ambíguos, e os critérios utilizados para solucionar esses problemas.

No capítulo 3, será proposta a implementação de um sistema conexionista desenvolvido para construir árvores filogenéticas segundo o algoritmo das médias. Após a descrição dessa implementação, será mostrado uma aplicação do sistema proposto.

No capítulo 4, será mostrado a implementação e os resultados experimentais de um sistema neurosimbólico desenvolvido para construir árvores filogenéticas pela análise de pequenos e grandes volumes de dados com e sem ambigüidades.

No capítulo 5, será mostrada a conclusão dessa dissertação e as perspectivas de trabalhos futuros a serem desenvolvidos.

2 Árvore Filogenética

2.1 Introdução

Neste capítulo, após uma breve definição da árvore filogenética, será mostrado como os dados sobre a evolução das características apresentadas pelas espécies investigadas podem ser representados e manipulados para se obter a árvore filogenética ótima[1] a partir do conjunto de todas as possíveis árvores filogenéticas construídas para essas espécies.

2.2 Descrição Formal da Árvore Filogenética

Definição 2.3.1 Uma *árvore filogenética* é uma estrutura $F = (\mathbb{X}, \mathbb{Y}, \Phi, \Gamma)$, onde:

- $\mathbb{X}$ é um conjunto finito de espécies,
- $\mathbb{Y}$ é um conjunto finito das características apresentadas pelos elementos de $\mathbb{X}$,
- Φ é uma relação de ordem sobre os elementos do conjunto $\mathbb{X}$, tal que:
  - i. $\forall x \in \mathbb{X}, x \not\Phi x$;
  - ii. $\forall x_1, x_2 \in \mathbb{X}$, se $x_1 \Phi x_2$, então $x_2 \not\Phi x_1$;
  - iii. $\forall x_1, x_2, x_3 \in \mathbb{X}$, se $x_1 \Phi x_2$ e $x_2 \Phi x_3$, então $x_1 \Phi x_3$.
- $\Gamma: \mathbb{X} \rightarrow \mathcal{P}(\mathbb{Y})$,

a qual é representada pelo gráfico $G = (\mathbb{X}, \Phi)$ com cada arco $a_k = (x_{k,1}, x_{k,2})$ de G rotulado pela seqüência dos elemento do conjunto $\Sigma_k = \Gamma(x_{k,2}) - \Gamma(x_{k,1})$.

---

[1] Árvore filogenética ótima - É o elemento do conjunto de todas as possíveis árvores filogenéticas para o conjunto de espécies investigadas que "melhor" reflete as suposições do filogeneticista sobre a evolução das espécies desse conjunto.

Por exemplo, seja F = ($\mathbb{X}$, $\mathbb{Y}$, gerou, Γ), onde:

$\mathbb{X}$ = { esp_1, esp_2, esp_3, esp_4, esp_5, esp_6, esp_7};

$\mathbb{Y}$ = { c_1, c_2, c_3, $c_{1'}$, $c_{2'}$, $c_{3'}$, $c_{2''}$};

esp_1 gerou esp_2;

esp_1 gerou esp_3;

esp_3 gerou esp_4;

esp_3 gerou esp_5;

esp_5 gerou esp_6;

esp_5 gerou esp_7;

$\Gamma(esp_1)$ = { c_1, c_2, c_3};

$\Gamma(esp_2)$ = { $c_{1'}$, c_2, c_3};

$\Gamma(esp_3)$ = { c_1, $c_{2'}$, c_3};

$\Gamma(esp_4)$ = { $c_{1'}$, $c_{2''}$, c_3};

$\Gamma(esp_5)$ = { c_1, $c_{2'}$, $c_{3'}$};

$\Gamma(esp_6)$ = { $c_{1'}$, $c_{2'}$, $c_{3'}$};

$\Gamma(esp_7)$ = { c_1, $c_{2'}$, $c_{3'}$};

que pode ser representada graficamente pela Fig. 2.1.

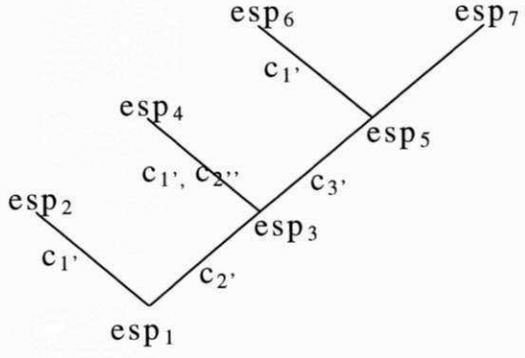


Fig. 2.1 - Árvore filogenética

A análise da árvore filogenética da Fig. 2.1 pode ser assim descrita: " existia inicialmente uma espécie esp_1 que apresentava as características c_1, c_2 e c_3 a qual deu origem as espécies esp_2 e esp_3 que apresentam as características $c_{1'}$, c_2, c_3 e c_1, $c_{2'}$, c_3. A espécie esp_3 deu origem as espécies esp_4 e esp_5 que apresentam as características $c_{1'}$, $c_{2''}$, c_3 e c_1, $c_{2'}$, $c_{3'}$. A

espécie esp_5 deu origem as espécies esp_6 e esp_7 que apresentam as características $c_{1'}$, $c_{2'}$, $c_{3'}$ e c_1, $c_{2'}$, $c_{3'}$."

As características analisadas pelos processos filogenéticos correspondem às modificações ocorridas em qualquer expressão fenotípica de um conjunto de espécie com base genética. Por exemplo, a expressão fenotípica "asa" no conjunto de insetos holometabólicos da Fig. 2.2 apresenta as características: asas posteriores semelhante a das asas anteriores (A,B) e halter[2] (C,D).

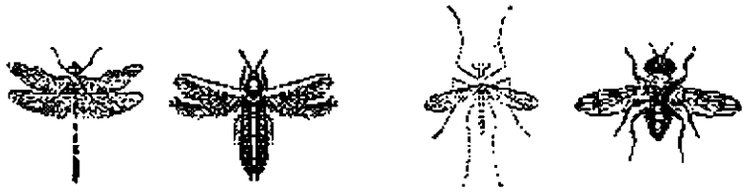


Fig. 2.2 - Insetos da família holometabólicos

No caso dos vírus, por exemplo, as expressão fenotípicas "presença de envoltório", "ausência de envoltório", fita "dupla", fita "simples", e natureza do ácido nucleico viral é "RNA" ou "DNA" são analisadas na construção da árvore filogenética dessa espécie. A Tabela 2.1 apresenta alguns vírus que apresentam estas características.

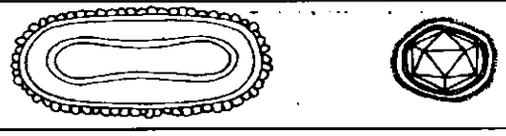
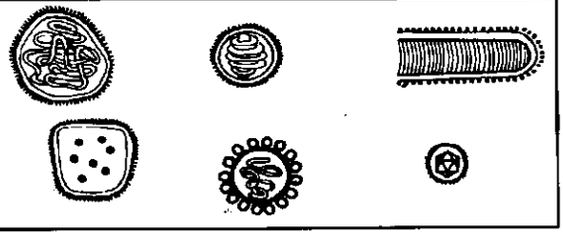
Vírus com fita dupla e DNA	
Sem envoltório	Com envoltório
	
Vírus com fita dupla, RNA e sem envoltório	
	
Vírus com fita simples e RNA	
Sem envoltório	Com envoltório
	

Tabela 2.1 - Representação esquemática de alguns vírus

[2] Halter- Característica que corresponde a presença só das asas posteriores.

Os processos da natureza responsáveis pela alteração das expressões fenotípicas das espécies durante a sua evolução recebem o nome de *processos filogenéticos fundamentais* [Bernardi, 1981] e compreende os processos de:

- *anagênese*, processo responsável pela transmissão das expressões fenotípicas da espécie ancestral para a espécie descendente com algumas modificações;
- *cladogenêse*, processo responsável pela divisão de uma espécie em duas ou mais espécies;
- *estasigênese*, processo responsável pela transmissão das expressões fenotípicas da espécie ancestral para a espécie descendente sem modificações;
- *extinção*, processo responsável pela morte de uma espécie.

As relações de parentescos entre as espécies presentes em uma árvore filogenética são:

- relação *ancestral direto*, ocorre entre duas espécies esp_i e esp_j onde esp_i gerou esp_j, ou vice-versa (por exemplo, na Fig. 2.1, esp_1 é ancestral direto de esp_2 e esp_3);
- relação *ancestral indireto*, ocorre entre duas espécies esp_i e esp_j onde a espécie esp_i gerou a espécies esp_1, a espécie esp_1 gerou a espécies esp_2, ..., a espécie esp_n gerou a espécies esp_j, ou vice-versa (por exemplo, na Fig. 2.1, esp_1 é ancestral indireto de esp_4);
- relação *ancestral comum*, ocorre entre duas espécies esp_i e esp_j que não apresentam uma relação ancestral direto ou indireto entre si, mas possuem uma espécie esp_k que é ancestral direto ou indireto à ambos (por exemplo, na Fig. 2.1, esp_2 e esp_3 tem como ancestral comum esp_1).

2.3 Matriz Polarizada

Geralmente, os dados sobre a evolução das características apresentadas pelas espécies investigadas, são armazenados em uma matriz polarizada que relaciona a espécie à série de transformação polarizada (STP) de cada uma das expressões fenotípicas apresentadas pelas espécies investigadas [Amorim, 1994][Bernardi, 1981][Lawrence, 1951][Wiley,1990].

A série de transformação polarizada de uma expressão fenotípica, informa a ordem cronológica na qual as características dessa expressão fenotípica ocorreram ao longo do tempo. Por exemplo, sejam θ, ε e ς características de uma mesma expressão fenotípica, dependendo da ordem na qual essas características surgiram ao longo da evolução das espécies na Terra, pode-se ter as seguintes séries de transformação polarizadas:

i.θ → ε → ς (lê-se θ gerou ε e ε gerou ς);

ii.θ → ς → ε (lê-se θ gerou ς e ς gerou ε);

iii.θ → ε (lê-se θ gerou ε e ς);
↘ ς

iv.ε → θ → ς (lê-se ε gerou θ e θ gerou ς);

v.ε → ς → θ (lê-se ε gerou ς e ς gerou θ);

vi.ε → θ (lê-se ε gerou θ e ς);
↘ ς

vii.ς → θ → ε (lê-se ς gerou θ e θ gerou ε);

viii.ς → ε → θ (lê-se ς gerou ε e ε gerou θ);

ix.ς → θ (lê-se ς gerou θ e ζ).
↘ ζ

As séries de transformação polarizadas lineares do tipo,

a característica θ *gerou a característica* ε *e a característica* ε *gerou a característica* ς,

expressam uma relação ancestral direto entre as espécies que apresentam a característica θ e ε, uma relação ancestral direto entre as espécies que apresentam a característica ε e ς, e uma relação ancestral indireto entre as espécies que apresentam a característica θ e ς. Enquanto que as séries de transformação polarizadas paralelas do tipo,

a característica θ *gerou a característica* ε *e a característica* ς,

expressam uma relação ancestral direto entre as espécies que apresentam a característica θ e ε, uma relação ancestral direto entre as espécies que apresentam a característica θ e ς, e uma relação ancestral comum entre as espécies que apresentam a característica ε e ς.

Mas, para que uma série de transformação polarizada possa ser armazenada em uma matriz, é necessário primeiro codificá-la de modo que, de posse desse código, qualquer pessoa possa reconstituir a série de transformação polarizada de uma expressão fenotípica a partir das informações contidas nessa matriz. Se as características de uma expressão fenotípica, em uma série de transformação polarizada, são transmitidas de maneira linear como nas séries de transformação i, ii, iv, v, vii e viii mostradas anteriormente, a sua codificação será processada

pela enumeração das características ocorridas nessa expressão fenotípica em ordem cronológica (Tabela 2.2).

STP	θ→ ε→ ς	θ→ ς→ ε	ε→ θ→ ς	ε→ς→ θ	ς→ θ→ ε	ς→ ε→ θ
espécie(θ)	0	0	1	2	1	2
espécie(ε)	1	2	0	0	2	1
espécie(ς)	2	1	2	1	0	0

Tabela 2.2 - Codificação em uma matriz polarizada de séries de transformação polarizadas linear

Porém, se as características de uma expressão fenotípica são transmitidas de maneira polarizada paralela, como nas séries de transformação iii, vi e ix mostradas anteriormente, então deve-se primeiro decompor cada série desse tipo, em tantas séries de transformação polarizadas lineares quanto for o número de características ocorridas simultaneamente na série de transformação polarizada paralela em decomposição. Segundo, codifica-se separadamente cada uma dessas séries de transformação polarizadas lineares (Tabela 2.3).

STP	θ→ ε, θ→ ς		ε→ θ, ε→ ς		ς→ θ, ς→ ε	
Decomposição	θ→ε	θ→ ς	ε →θ	ε→ ς	ς →θ	ς→ε
espécie(θ)	0	0	1	0	1	0
espécie(ε)	1	0	0	0	0	1
espécie(ς)	0	1	0	1	0	0

Tabela 2.3 - Codificação em uma matriz polarizada de séries de transformação polarizadas via decomposição

A utilização da matriz polarizada restringe o tipo de dados analisados aos dados de natureza morfológica, deixando de lado, dados de outras naturezas, tais como: genética, citológica, etc., que são utilizados hoje em dia para melhorar a compreensão do processo de evolução das espécies [Bachmann, 1995].

2.4 Matriz Característica

Algumas vezes as informações sobre a evolução das características apresentadas por um conjunto de espécies são representadas em uma matriz característica. Uma matriz característica é uma versão simplificada da matriz polarizada que relaciona cada espécie às características terminais do conjunto de espécies investigadas. Diz-se que, x é uma característica terminal do conjunto de espécies investigadas, se nenhuma das características

apresentadas pelas espécies do conjunto forem geradas a partir de x. Por exemplo, sejam θ, ω, ε, ρ, τ, κ, π, α, β, σ e δ características das expressões fenotípicas do conjunto de espécies $\{esp_1, esp_2, esp_3, esp_4\}$ que possuem as seguintes séries de transformação polarizadas:

i. $\theta \longrightarrow \omega$ (lê-se θ gerou ω);

ii. $\delta \longrightarrow \varepsilon$ (lê-se δ gerou ε);

iii. $\rho \longrightarrow \tau$ (lê-se ρ gerou τ);

iv. $\alpha \longrightarrow \beta$ (lê-se α gerou β);

v. $\kappa \longrightarrow \pi$ (lê-se κ gerou π e σ).
$\searrow$
σ

A espécie esp_1 apresenta as características θ, δ, ρ, α e κ, a espécie esp_2 apresenta as características ω, δ, ρ, α e κ, a espécie esp_3 apresenta as características ω, ε, τ, α e π e a espécie esp_4 apresenta as características ω, ε, τ, β e σ. A Tabela 2.4 abaixo apresenta a matriz polarizada para essas espécies.

STP	θ→ω	δ→ε	ρ→τ	α→β	κ→ π, s	
Decomposição	θ→ω	δ→ε	ρ→τ	α→β	κ→π	κ→σ
esp_1	0	0	0	0	0	0
esp_2	1	0	0	0	0	0
esp_3	1	1	1	0	1	0
esp_4	1	1	1	1	0	1

Tabela 2.4 - Matriz polarizada usada como exemplo no capítulo 2

A partir da Tabela 2.4, constrói-se a matriz característica das espécies investigadas (Tabela 2.5). Primeiro coloca-se na segunda linha e na primeira coluna a letra romana maiúscula A para representar a espécie esp_1, na terceira linha e na primeira coluna a letra romana maiúscula B para representar a espécie esp_2, e assim sucessivamente até que, todas as espécies da Tabela 2.4 estejam representados por uma letra romana maiúscula. Em seguida, coloca-se na primeira linha e na segunda coluna o número 1 para representar a característica terminal ω, na primeira linha e na terceira coluna o numero 2 para representar a característica terminal ε, e assim sucessivamente até que, todas as características terminais das séries de transformação polarizadas da Tabela 2.4 estejam representados por um número. Os elementos dessa matriz, das i linhas e j colunas, com $2 \leq i \leq$ número de linhas da matriz polarizada

analisada e $2 \leq j \leq$ número de colunas da matriz polarizada analisada, são representados por $c_{i,j}$ e tem o mesmo valor do elemento $p_{i+1,j}$ da matriz polarizada analisada [Abe,1991].

	1	2	3	4	5	6
A	0	0	0	0	0	0
B	1	0	0	0	0	0
C	1	1	1	0	1	0
D	1	1	1	1	0	1

Tabela 2.5 - Matriz característica

2.5 Matriz Similaridade

A matriz similaridade é uma matriz quadrada de dimensão n que relaciona duas espécies a uma medida de similaridade, onde n é igual ao número de espécies investigadas mais 1. Os elementos dessa matriz são obtidos a partir da análise dos elementos $c_{u,v}$ da matriz característica e são representados por $s_{i,j}$. Os elementos da primeira linha e coluna contém uma letra romana maiúscula que corresponde ao nome das espécies investigadas. O valor do elemento $s_{i,j}$ é igual ao número de dados $c_{i,k} = c_{j,k}$, com $2 \leq i,j \leq$ número de linhas da matriz característica analisadas e $2 \leq k \leq$ número de colunas da matriz característica analisadas. Por exemplo, o valor dos elementos da segunda linha da matriz similaridade da Tabela 2.6 obtida a partir da Tabela 2.5, é o seguinte:

$s_{2,2} = (c_{2,1}=c_{2,1}) + (c_{2,2}=c_{2,2}) + (c_{2,3}=c_{2,3}) + (c_{2,4}=c_{2,4}) + (c_{2,5}=c_{2,5}) + (c_{2,6}=c_{2,6})$

$= (0=0) + (0=0) + (0=0) + (0=0) + (0=0)+(0=0$

$= 6$

$s_{2,3} = (c_{2,1}=c_{3,1}) + (c_{2,2}=c_{3,2}) + (c_{2,3}=c_{3,3}) + (c_{2,4}=c_{3,4}) + (c_{2,5}=c_{3,5}) + (c_{2,6}=c_{3,6})$

$= (0=1) + (0=0) + (0=0) + (0=0) + (0=0)+(0=0)$

$= 5$

$s_{2,4} = (c_{2,1}=c_{4,1}) + (c_{2,2}=c_{4,2}) + (c_{2,3}=c_{4,3}) + (c_{2,4}=c_{4,4}) + (c_{2,5}=c_{4,5}) + (c_{2,6}=c_{4,6})$

$= (0=1) + (0=1) + (0=1) + (0=0) + (0=1)+(0=0)$

$= 2$

$s_{2,5} = (c_{2,1}=c_{5,1}) + (c_{2,2}=c_{5,2}) + (c_{2,3}=c_{5,3}) + (c_{2,4}=c_{5,4}) + (c_{2,5}=c_{5,5}) + (c_{2,6}=c_{5,6})$

$= (0=1) + (0=1) + (0=1) + (0=1) + (0=0)+(0=1)$

$= 1$

	A	B	C	D
A	6	5	2	1
B	5	6	3	2
C	2	3	6	3
D	1	2	3	6

Tabela 2.6 - Matriz similaridade

Observação: No Apêndice A é mostrado um exemplo de como dados genéticos e morfológicos podem ser combinados em uma única matriz similaridade.

2.6 Matriz Distância

A matriz distância é uma matriz quadrada de dimensão n que relaciona duas espécies a uma medida de distância, onde n é igual ao número de espécies investigadas mais 1. Os elementos dessa matriz são obtidos a partir da análise dos elementos $c_{u,v}$ da matriz característica e são representados por $d_{i,j}$. Os elementos da primeira linha e coluna contém uma letra romana maiúscula que corresponde ao nome das espécies investigadas. O valor do elemento $d_{i,j}$ é igual ao número de dados $c_{i,k} \neq c_{j,k}$, com $2 \leq i,j \leq$ número de linhas da matriz característica analisada e $2 \leq k \leq$ número de colunas da matriz característica analisada. Por exemplo, o valor dos elementos da segunda linha da matriz distância da Tabela 2.7 obtida a partir da Tabela 2.5, é o seguinte:

$$d_{2,2} = (c_{2,1} \neq c_{2,1}) + (c_{2,2} \neq c_{2,2}) + (c_{2,3} \neq c_{2,3}) + (c_{2,4} \neq c_{2,4}) + (c_{2,5} \neq c_{2,5}) + (c_{2,6} \neq c_{2,6})$$

$$= (0 \neq 0) + (0 \neq 0) + (0 \neq 0) + (0 \neq 0) + (0 \neq 0) + (0 \neq 0)$$

$$= 0$$

$$d_{2,3} = (c_{2,1} \neq c_{3,1}) + (c_{2,2} \neq c_{3,2}) + (c_{2,3} \neq c_{3,3}) + (c_{2,4} \neq c_{3,4}) + (c_{2,5} \neq c_{3,5}) + (c_{2,6} \neq c_{3,6})$$

$$= (0 \neq 1) + (0 \neq 0) + (0 \neq 0) + (0 \neq 0) + (0 \neq 0) + (0 \neq 0)$$

$$= 1$$

$$d_{2,4} = (c_{2,1} \neq c_{4,1}) + (c_{2,2} \neq c_{4,2}) + (c_{2,3} \neq c_{4,3}) + (c_{2,4} \neq c_{4,4}) + (c_{2,5} \neq c_{4,5}) + (c_{2,6} \neq c_{4,6})$$

$$= (0 \neq 1) + (0 \neq 1) + (0 \neq 1) + (0 \neq 0) + (0 \neq 1) + (0 \neq 0)$$

$$= 4$$

$d_{2,5} = (c_{2,1} \neq c_{5,1}) + (c_{2,2} \neq c_{5,2}) + (c_{2,3} \neq c_{5,3}) + (c_{2,4} \neq c_{5,4}) + (c_{2,5} \neq c_{5,5}) + (c_{2,6} \neq c_{5,6})$

$= (0 \neq 1) + (0 \neq 1) + (0 \neq 1) + (0 \neq 1) + (0 \neq 0) + (0 \neq 1)$

$= 5$

	A	B	C	D
A	0	1	4	5
B	1	0	3	4
C	4	3	0	3
D	5	4	3	0

Tabela 2.7 - Matriz distância

Observação: No Apêndice B é mostrado um exemplo de como dados genéticos e morfológicos podem ser combinados em uma única matriz distância.

2.7 Alguns Algoritmos de Construção de Árvores Filogenéticas

O modelo Henianno tenta construir a árvore filogenética de um conjunto de espécies, através da reconstrução da história evolutiva de suas características [Amorim, 1994][Wiley, 1990][Meidanis, 1994]. Nesta seção serão apresentados três algoritmos de construção de árvores filogenéticas segundo o modelo Henianno. Para facilitar o entendimento desses algoritmos, será mostrada a árvore filogenética construída a partir da matriz característica da Tabela 2.5 e da matriz distância da Tabela 2.7.

A distância entre a espécie A e a espécie B é o complemento da similaridade entre a espécie A e a espécie B [Meidanis, 1994]. Logo se um algoritmo constrói a árvore filogenética para um conjunto de espécies através da mínima distância entre as espécies, ele pode ser usado para construir uma árvore filogenética através da máxima similaridade entre as espécies. Por esse motivo, nesta seção, não será apresentado qualquer algoritmo de construção da árvore filogenética de um conjunto de espécies a partir da matriz similaridade.

2.7.1 A Regra de Inclusão e Exclusão

A regra de inclusão e exclusão é um algoritmo de construção da árvore filogenética de um conjunto de espécies, que recebe a matriz característica dessas espécies e devolve o conjunto de todas as possíveis árvores filogenéticas dessas espécies. A seguir serão apresentados os passos desse algoritmo[Wiley,1990].

1º passo - Para cada característica terminal N considerada no estudo, será construída uma árvore filogenética AF_i (Fig. 2.3a, 2.3b, 2.3c, 2.3d, 2.3e e 2.3f), com $1 \leq i \leq$ número de características terminais consideradas;

2º passo - Combine todas as relações de parentescos presentes nas AF_i construídas no passo anterior, de modo que, as relações de parentesco presentes na AF_i sejam preservadas pela adição das relações de parentesco presentes na AF_j com $i < j$ (Fig. 2.3g).

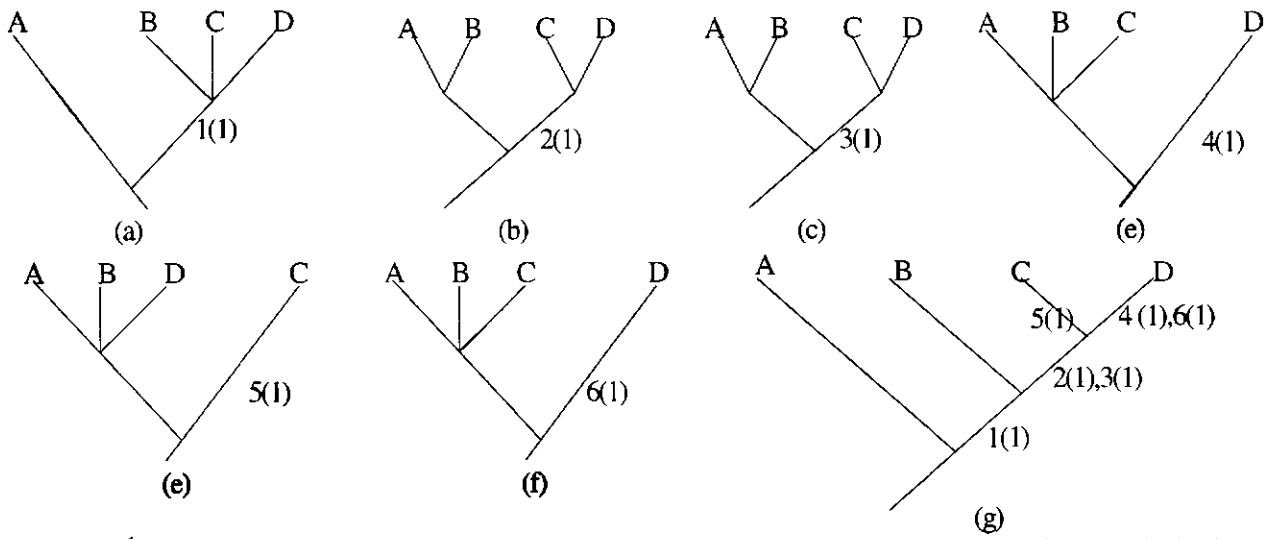


Fig. 2.3 - Árvore filogenética para as espécies da Tabela 2.5 obtida pela análise das características terminais: 1 (a), 2(b), 3(c), 4(d), 5(e) e 6(f). A combinação dessas árvores é mostrada em (g).

2.7.2 O Algoritmo de Wagner

O algoritmo de Wagner, é um algoritmo de construção da árvore filogenética de um conjunto de espécies, que recebe a matriz característica dessas espécies e devolve uma das árvores filogenéticas possíveis para essas espécies. A seguir serão apresentados os passos desse algoritmo [Amorim, 1994][Wiley,1990].

1º passo - Especifique a espécie raiz;

2º passo - Construa a matriz distância para as espécies da matriz característica fornecida;

3º passo - Selecione a espécie que tiver a menor distância para a espécie raiz. Esta será a atual espécie selecionada;

4º passo - Crie um ramo ligando a atual espécie selecionada a espécie raiz com comprimento igual à distância entre essas duas espécies;

5º passo - Selecione a próxima espécie que apresente a menor distância para a espécie raiz, esta será agora a atual espécie selecionada;

6º passo - Calcule a distância, usando a eq.2.1 abaixo, entre a atual espécie selecionada e todas as outras espécies já selecionadas;

$$Dist\left(Esp_x, Esp_y\right) = \sum_{x=1}^{n} \left| \text{Caract}_x - \text{Caract}_y \right|^3 \qquad \text{(eq.2.1)}$$

onde $Caract_i$ é o vetor característica $(c_{i,1},\ldots, c_{i,n})$ da espécie Esp_i, $c_{i,j}$ é o elemento da *i* linha e *j* coluna da matriz característica *C* de dimensão *n*.

7º passo - Selecione a espécie que apresenta a menor distância no passo anterior a atual espécie selecionada. Esta será a espécie irmã selecionada;

8º passo - Crie um ramo ligando a atual espécie selecionada S ao meio do ramo que chega na espécie irmã selecionada I de comprimento *Comp* calculado por (eq.2.2) (Fig. 2.4a, 2.4b e 2.4c);

$$Comp(S,I) = \frac{\left(Dist(S,I) + Dist(S, Ancestral(I))\right) - Dist(I, Ancestral(I))}{2} \qquad \text{(eq.2.2)}$$

onde a função Ancestral(X) retorna a espécie ancestral direto da espécie X.

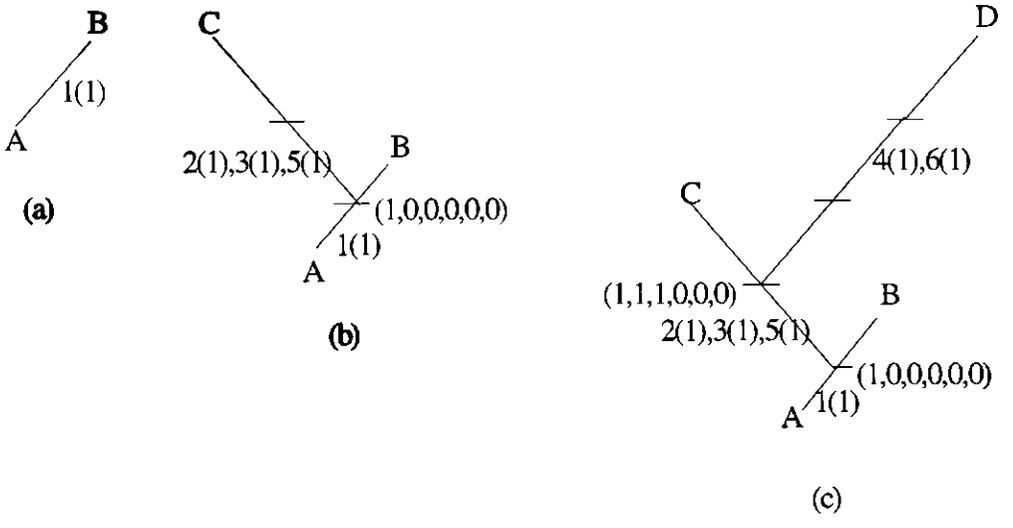


Fig. 2.4 - Árvores filogenéticas construídas pela primeira (a), segunda (b) e terceira (c) execução do 5º passo do algoritmo de Wagner para as espécies da Tabela 2.5.

[3] $|caract_x - caract_y|$ - É o valor absoluto da diferença entre os valores $caract_x$ e $caract_y$.

9º passo - Determine o vetor característica[4] do ancestral comum *A* $Carct_A = (c_{A,1},\ldots,c_{A,n})$ entre a atual espécie selecionada S e a espécie irmã selecionada I, onde $c_{A,i}$ = menor($c_{O,i}, c_{I,i}$) e $c_{x,y}$ é o elemento da x linha e y coluna da matriz característica C;

10º passo - Enquanto existir uma espécie que ainda não foi selecionada volte ao passo 5.

2.7.3 O Algoritmo das Médias (UPGMA)

O algoritmo das médias é um algoritmo de construção da árvore filogenética de um conjunto de espécies que recebe a matriz distância dessas espécies e devolve uma das árvores filogenéticas possíveis para essas espécies. Note que os ramos da árvore filogenética construída por esse algoritmo, não contém rótulos. A seguir serão apresentado os passos deste algoritmo [Meidanis,1994].

1º passo - Tome o par de espécies com menor distância entre si e agrupe-os numa super-espécie. Este par de espécies terá um ancestral comum direto (Fig. 2.5a B e A, 2.5b C e S_1 e 2.5c D e S_2);

2º passo - Recalcule a distância de cada uma das demais espécies S_i para a super-espécie recém criado como sendo a média das distâncias de S_i para cada uma das espécies que constituem a super-espécie (Tabelas 2.8 e 2.9);

3º passo - Repita os passos 1 e 2 enquanto houver dois ou mais (super-)espécies não visitadas.

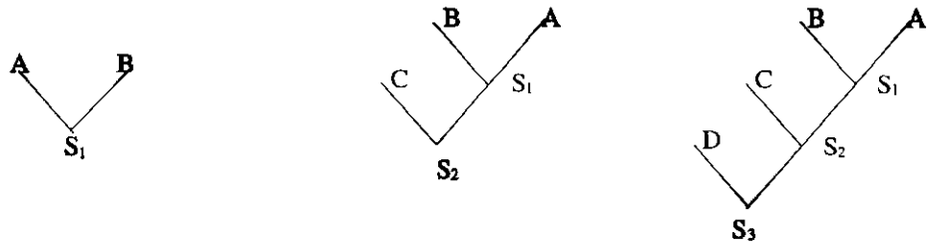
	S_1	C	D
S_1	0	2,5	5,5
C	2,5	0	3
D	5,5	3	0

Tabela 2.8 - Matriz distância obtida a partir dos dados da Tabela 2.7 considerando que as espécies A e B tem a super-espécie S_1 como ancestral direto.

	S_2	D
S_2	0	4,25
D	4,25	0

Tabela 2.9 - Matriz distância obtida a partir dos dados da Tabela 2.8 considerando que as espécies S_1 e C tem a super-espécie S_2 como ancestral direto

[4] Nota - Na versão original, este algoritmo só é aplicado para matrizes polarizadas binárias, assim o cálculo do vetor característica do ancestral, pode ser definida como a média do valor do vetor característica entre a atual espécie selecionada S e a espécie irmã selecionada *I*.



(c) (b) (c)

Fig. 2.5 - Árvore filogenética construída pelo primeiro (a), segundo (b) e terceiro (c) execução do 1º passo do algoritmo das médias para as espécies da Tabela 2.7.

2.8 Problemas Encontrados com os Algoritmos

O algoritmo da regra de inclusão e exclusão e o algoritmo de Wagner servem muito bem para construir a árvore filogenética ótima, quando os dados sobre a evolução das características apresentadas pelas espécies, não apresentam conflitos[5]. Pois se esses dados forem conflitantes, então a matriz característica de dimensão *nxm* terá duas colunas *i* e *j*, tal que:

$\forall x \in \{1,...,m\}, \exists S_x = \{n \mid c_{n,x} \neq 0\}$ com $S_i \cap S_j \neq \emptyset$, $S_i \not\subset S_j$ e vice-versa.

Por exemplo, considere a matriz característica da Tabela 2.10.

	1	2	3	4	5	6
A	0	0	0	0	0	0
B	1	1	0	0	1	1
C	1	1	1	1	1	1
D	1	1	1	1	0	0

Tabela 2.10 - Matriz característica com conflito

Inicialmente constrói-se os conjunto $\mathcal{S}_1$ = {B,C,D}, $\mathcal{S}_2$ = {B,C,D}, $\mathcal{S}_3$ = {C,D}, $\mathcal{S}_4$ = {C,D}, $\mathcal{S}_5$ = {B,C}, $\mathcal{S}_6$ = {B,C},note que $\mathcal{S}_1 = \mathcal{S}_2$, $\mathcal{S}_3 = \mathcal{S}_4$, e $\mathcal{S}_5 = \mathcal{S}_6$, então tém-se a seguinte análise dos dados:

- $\mathcal{S}_1 \cap \mathcal{S}_3 = \mathcal{S}_3$ e $\mathcal{S}_1 \cap \mathcal{S}_5 = \mathcal{S}_5$, logo as características 1 e 2 não apresentam conflito com as características 3, 4, 5 e 6;
- $\mathcal{S}_3 \cap \mathcal{S}_5 = \{C\}$, $\mathcal{S}_3 \not\subset \mathcal{S}_5$ e $\mathcal{S}_5 \not\subset \mathcal{S}_3$, logo as características 3 e 4 apresentam conflito com as características 5 e 6.

[5] Conflito - O conflito ocorre quando as relações de parentesco expressas por uma característica contradizem as relações de parentesco expressas por uma outra característica.

Nesse caso, o algoritmo da regra de inclusão e exclusão irá construir um conjunto de árvores filogenéticas para as espécies analisadas com cardinalidade maior que 1 (Fig. 2.6). A árvore filogenética ótima, para as espécies de uma matriz característica com conflito construída pelo algoritmo da regra de inclusão e exclusão, será a árvore filogenética obtida pela aplicação de um critério de otimização ao conjunto de árvores filogenéticas construído. Enquanto que o algoritmo de Wagner, irá construir uma árvore filogenética que será dependente da ordem na qual as espécies estão dispostas na matriz característica com conflito analisada (Fig. 2.6(a)). Para se obter a árvore filogenética ótima, com o algoritmo de Wagner, deve-se primeiro alternar as posições das espécies na matriz analisada, para se obter o conjunto de todas as possíveis árvores filogenéticas para essas espécies. E depois, aplica-se sobre esse conjunto um critério de otimização.

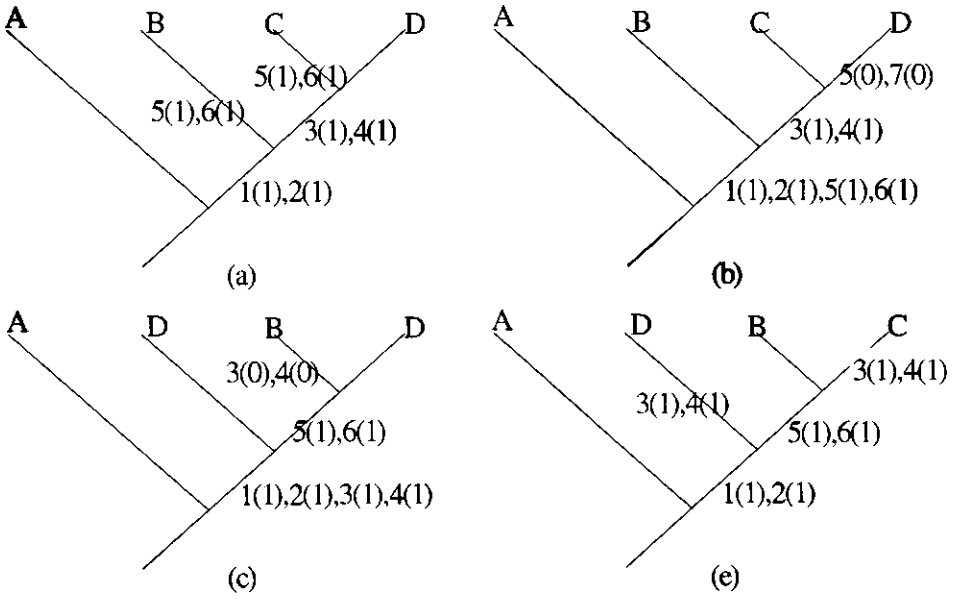


Fig. 2.6 - Árvores filogenéticas construídas pelo algoritmo de inclusão e exclusão a partir da Tabela 2.10

O algoritmo das médias constrói a árvore filogenética ótima, quando os dados sobre a evolução das características apresentadas pelas espécies não apresenta conflito. Pois, se os dados sobre a evolução das características apresentadas pelas espécies forem conflitantes, então a matriz distância de dimensão *nxn*, com θ igual ao menor valor armazenado, terá dois ou mais elementos $d_{i,j} = d_{u,v} = \theta$, tal que:

$$i, j \neq u, v \text{ e } i, j \neq v, u.$$

Por exemplo, considere a matriz distância da Tabela 2.11 construída a partir da Tabela 2.10.

	A	B	C	D
A	0	4	6	4
B	4	0	2	4
C	6	2	0	2
D	4	4	2	0

Tabela 2.11 - Matriz distância com conflito

O menor valor armazenado nesta Tabela é 2. O valor 2 corresponde a distância entre as espécies B e C, e entre as espécies C e D o que é um conflito.

Quando a matriz distância analisada pelo algoritmo das médias apresenta conflito, a árvore filogenética construída irá depender da ordem na qual as espécies estão dispostas na matriz distância (Fig. 2.7). A árvore filogenética ótima, construída pelo algoritmo das médias a partir de uma matriz distância com conflito, será obtida da mesma forma que a árvore filogenética ótima construída pelo algoritmo de Wagner a partir de uma matriz característica com conflito.

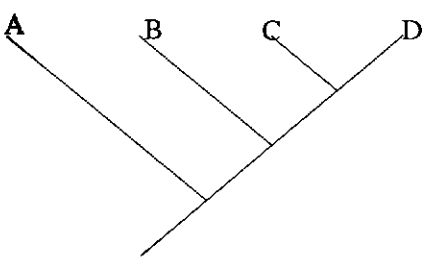


Fig. 2.7 - Árvores filogenéticas construídas pelo algoritmo das médias a partir da Tabela 2.11

2.9 Critérios de Otimização

Nesta seção, serão apresentados dois critérios de otimização utilizados para se obter à árvore filogenética ótima do conjunto de todas as árvores filogenéticas possíveis para as espécies investigadas. Os critérios são denominados de consenso [Amorim, 1994] [Wiley, 1990] e de parcimônia [Amorim, 1994][Meidanis, 1994][Wiley, 1990].

2.9.1 Consenso

O consenso, é um tipo de critério de otimização usado para construir uma nova árvore filogenética para o conjunto de espécies analisadas, a partir das árvores filogenéticas

construídas por um dado algoritmo. A nova árvore filogenética deverá apresentar as informações comuns às árvores filogenéticas do conjunto $\mathbb{A}$, onde $\mathbb{A}$ é o conjunto das árvores filogenéticas construídas que apresentam o menor número de características em seus ramos. A nova árvore não apresenta rótulos em seus ramos. Os tipos de consenso propostos são:

- *Consenso de Adams,* onde a nova árvore filogenética é formada só pelas relações de parentesco que não são conflitantes com as relações de parentesco expressas em todas as árvores filogenéticas do conjunto $\mathbb{A}$ [Amorim, 1994][Wiley, 1990];
- *Consenso Estrito*, onde a nova árvore filogenética é formada só pelas relações de parentesco que aparecem em todas as árvores filogenéticas do conjunto $\mathbb{A}$ [Amorim, 1994][Wiley, 1990];
- *Consenso de Maioria,* onde a nova árvore filogenética é formada só pelas relações de parentescos que são expressas na maioria das árvores filogenéticas do conjunto $\mathbb{A}$ [Amorim, 1994][Wiley, 1990]. Por exemplo, considere as três árvores filogenéticas da Fig. 2.8 cada uma com 8 características em seus ramos.

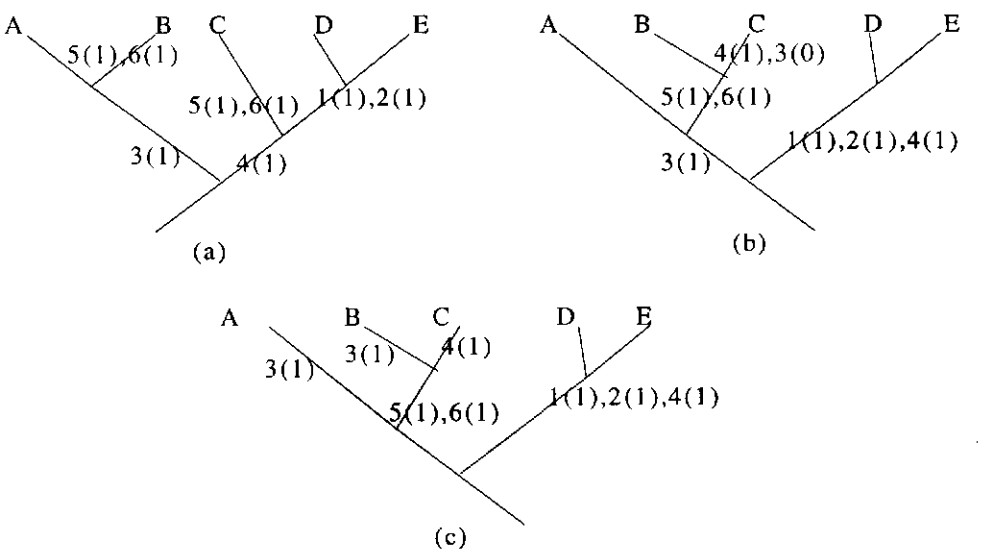


Fig. 2.8 - Conjunto de todas as árvores filogenéticas construídas pela regra de inclusão e exclusão para as espécies da Tabela 2.12.

	1	2	3	4	5	6
A	0	0	1	0	0	0
B	0	0	1	0	1	1
C	0	0	0	1	1	1
D	1	1	0	1	0	0
E	1	1	0	1	0	0

Tabela 2.12 - Matriz característica com conflito usada para os exemplos dos critérios de otimização

Analisando as árvores filogenéticas da Fig. 2.9, obtém-se a árvore filogenética da Fig. 2.9(a) com o consenso de Adams, Fig. 2.9(b) com o consenso Estrito e Fig. 2.9(c) com o consenso da Maioria.

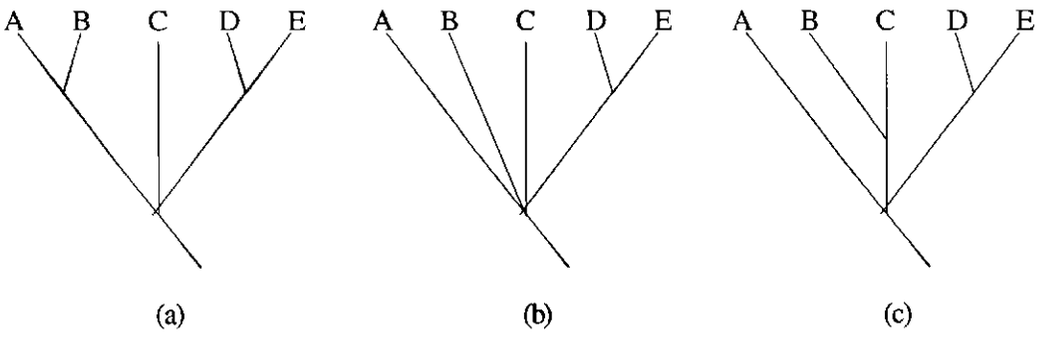


Fig. 2.9 - As árvores filogenéticas construídas com os critérios de consenso

2.9.2 Parcimônia

A parcimônia é um tipo de critério de otimização que relaciona o número de características contidas nos rótulos dos ramos da árvore filogenética e à existência ou não de uma seqüência necessariamente ordenada das alterações ocorridas nas características que conduzem ao surgimento de novas espécies e também a admissão ou não de ocorrer tanto reversão[6] como convergência[7]. Os tipos de parcimônia são:

- *Parcimônia de Wagner*, busca minimizar o número de transições de estados, admite convergências, mas só admite reversões quando a ordem entre as alterações ocorridas nas características é preservada [Amorim, 1994][Wiley, 1990] (Fig. 2.7 (a), (b) e (c));
- *Parcimônia de Fitch*, busca minimizar o número de transições de estados e admite tanto reversões como convergência [Amorim, 1994][Wiley, 1990] (Fig. 2.7 (a), (b) e (c));
- *Parcimônia de Dollo*, busca minimizar o número de transições de estados, admite reversões, mas não admite convergências [Amorim, 1994][Meidanis, 1994][Wiley, 1990] (Fig. 2.10);

[6] Reversão - É quando um descendente *X* apresenta a expressão fenotípica *C* com característica *n* e seu ancestral direto *Y* apresenta *C* com característica *n'*, sendo *n'* uma das características geradas a partir da característica *n*.

[7] Convergência - É quando a mesma expressão fenotípica aparece em mais de um ramo da árvore filogenética com a mesma característica.

- *Parcimônia de Camin-Sokal,* busca minimizar o número de transições de estados, admite convergência, mas não admite reversão [Amorim, 1994][Meidanis, 1994] [Wiley, 1990] (Fig. 2.7 (a) e (c)).

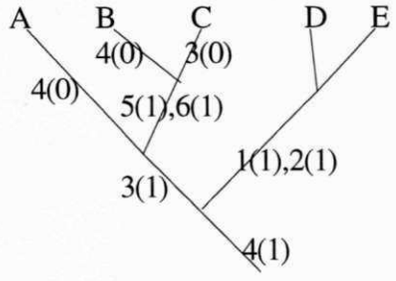


Fig. 2.10 - Árvores filogenéticas construídas pela regra de inclusão e exclusão para as espécies da Tabela 2.12 obtida pelo critério de parcimônia de Dollo.

No próximo capítulo será apresentado o desenvolvimento de uma abordagem conexionista de construção de árvores filogenéticas, inspirado no algoritmo das médias. O novo algoritmo irá tratar os problemas de conflito com o auxílio do usuário (filogeneticista).

3 Construção de Árvore Filogenética Usando Redes Neurais de Hopfield

3.1 Introdução

Neste capítulo apresenta-se o projeto e os resultados experimentais de uma rede neural de Hopfield usada para construção de árvores filogenéticas, com e sem filogenia perfeita[1].

3.2 Motivação

A construção de árvores filogenéticas com e sem filogenia perfeita é um problema da classe NP-completo [Meidanis, 1994]. Os métodos desenvolvidos para resolver problemas NP-completos devem ser capazes também de resolver todos os outros problemas NP-completos, isto em tempo polinomial [Campello, 1994]. Esta característica do método justifica a sua aplicação ao problema de construção de uma árvore filogenética com e sem filogenia perfeita. A rede neural de Hopfield pode ser usada para resolver problemas NP-completo [Hopfield, 1986] .

3.3 A Rede Neural de Hopfield

Uma das primeiras aplicações da rede neural de Hopfield, usando neurônios com função de ativação com dois estados, foi como memórias associativas [Kovács, 1996][Hopfield, 1982]. Mais tarde, essa rede foi alterada para trabalhar com estados intermediários, recebeu um elemento de armazenamento, e passou a ser aplicada ao problema NP-completo de otimização combinatória [Aiyer, 1990][Ali, 1993][Ansari, 1995][Haykin, 1994].

[1] Filogenia Perfeita - Diz-se que uma árvore filogenética tem filogenia perfeita se ela não apresenta nem reversão nem convergência.

3.3.1 Rede Neural de Hopfield Usada como Memória Associativa

Hopfield, baseado no sistema nervoso humano, propôs uma rede neural com um certo número de pontos estáveis em um espaço de estados [Hopfield, 1982]. O estado da rede neural de Hopfield da Fig. 3.1 foi descrito como um vetor $V = (V_1, V_2, ..., V_n)$, e pode ser usada para aprender um conjunto de estados $V^s = \{V^s_1, V^s_2, ..., V^s_m\}$. A dinâmica de evolução dos estados da rede é assíncrona.

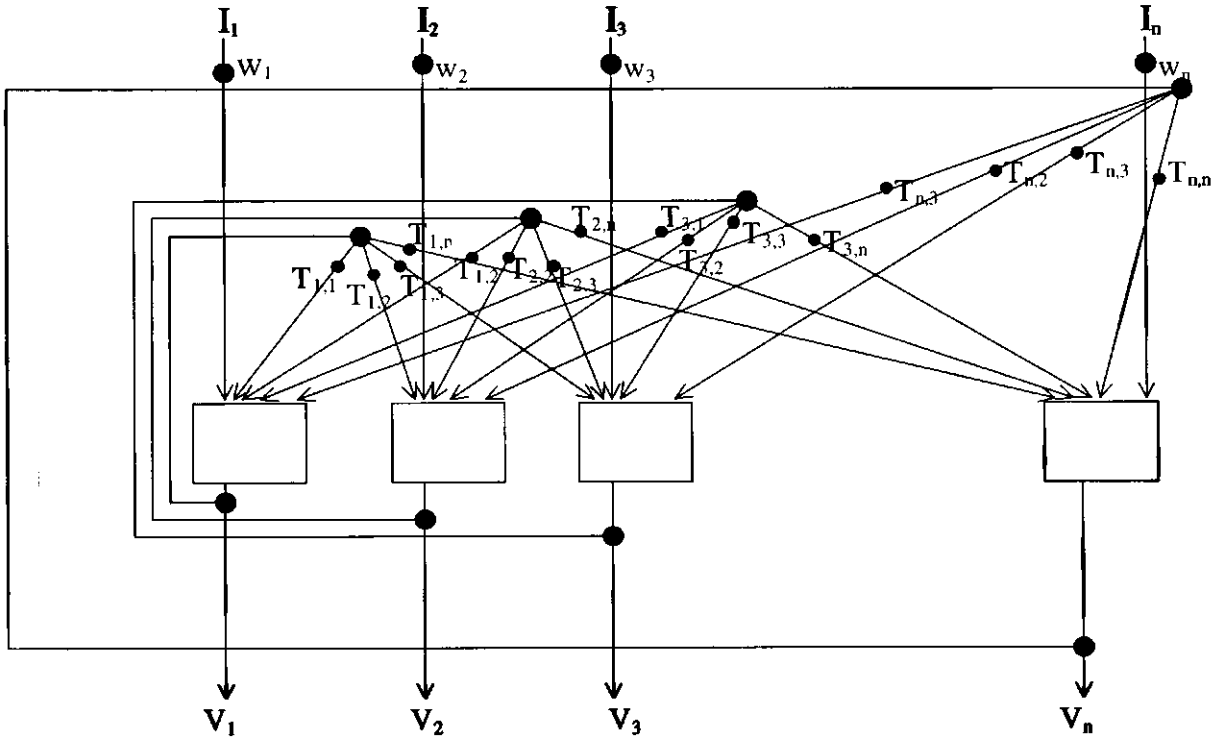


Fig. 3.1 - Rede Neural de Hopfield

Supondo que a rede é inicializada no estado não estável V^s_i, então a dinâmica de evolução dos estados da rede evoluirá até que a rede atinja o estado estável V^s_j, com $1 \leq j \leq m$ e $V^s_j = V^s_i + \Delta$. Em outras palavras, quando o vetor V^s_i contém um conhecimento parcial do item V^s_j, a rede neural evoluirá até o estado V^s_j. Os neurônios da rede neural de Hopfield (Fig. 3.2) tem função de ativação lógica (dois estados, V_1 = "0" e V_1 = "1") como mostrado na Fig. 3.3.

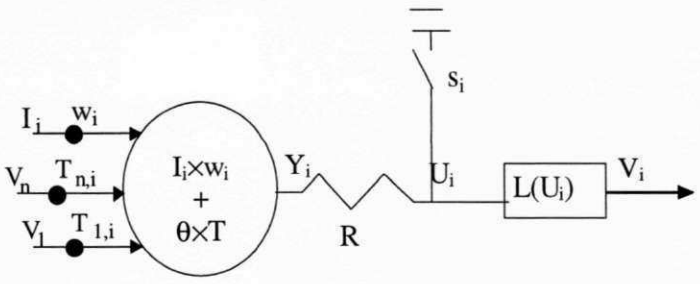


Fig. 3.2 - Neurônio de Hopfield com função de ativação lógica L(.)

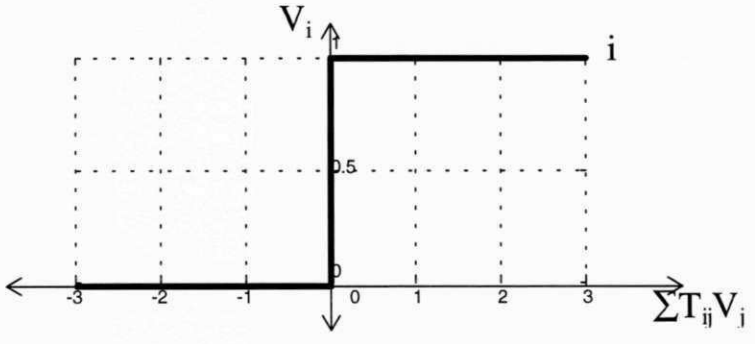


Fig. 3.3 - Função Lógica

Todos os neurônios são interconectados pelas suas saídas através de pesos $T_{i,j}$. $T_{i,j} = 0$ quando $i = j$ e $T_{i,j} = T_{j,i} \neq 0$ quando $i \neq j$. O comportamento do neurônio i dependerá do valor do limiar desse neurônio U_i e pode ser descrito pela regra abaixo:

R3.1 Função lógica de ativação do neurônio

se $\sum_{j \neq i} T_{j,i} \times V_j < U_i$

então V_i = "0"

senão V_i = "1"

onde V_i é a saída do neurônio i.

Hopfield [Hopfield, 1982] sugeriu uma função de Lyapunov, denominada por ele de função de energia, para representar os estados dos neurônios com função de ativação lógica da sua rede eq.3.1.

$$E = -\frac{1}{2} \sum \sum_{i \neq j} T_{i,j} \times V_i \times V_j \qquad \text{(eq.3.1)}$$

Usando a eq.3.1, Hopfield mostrou que a sua rede neural possui pontos limites estáveis. Para a função da eq.3.1, pode-se definir a variação da energia da rede em relação à variação da saída do neurônio V_i, como:

$$\frac{\Delta E}{\Delta V_i} = -\sum_{i \neq j} T_{i,j} \times V_j \qquad \text{(eq.3.2)}$$

Hopfield usou um algoritmo para alterar V_j na direção de menor energia. Na eq.3.2 $\Sigma T_{i,j} \times V_j$ é positivo. Hopfield observou que as saídas da sua rede neural, com função de ativação lógica, ao se dirigirem para o estado de menor energia atingem um estado estável.

3.3.2 Rede Neural de Hopfield Usada para Problemas NP-completo

Hopfield também propôs que os neurônios de sua rede recorrente usassem, além de um elemento de armazenamento (circuito elétrico RC), uma função de ativação monotônica crescente, tal como a função sigmóide (Fig. 3.4a) e tangente hiperbólica (Fig. 3.4b), para que essa rede possa ser aplicada a classe de problemas NP-completo.

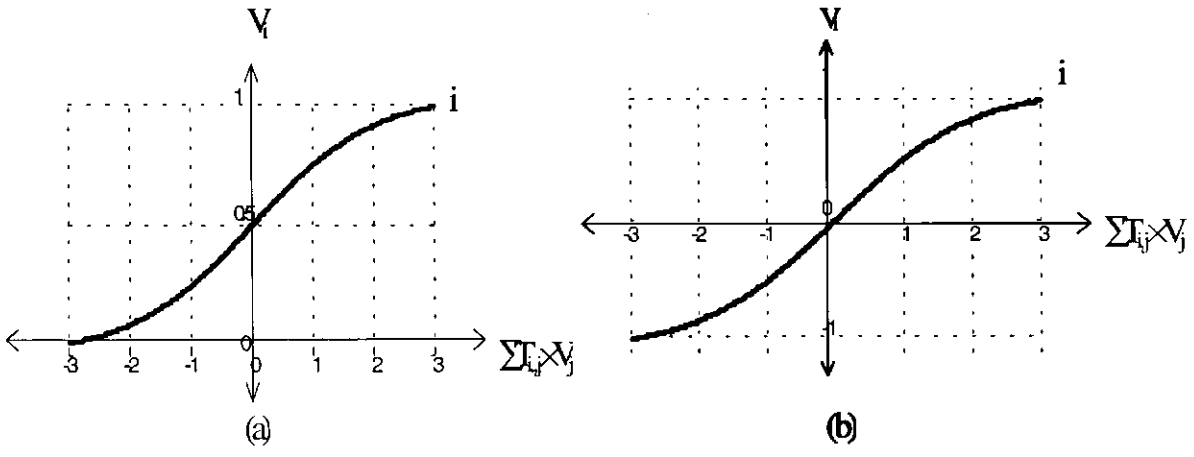


Fig. 3.4 - Função sigmóide (a) e função tangente hiperbólica (b)

A Fig. 3.1 mostra a arquitetura de uma rede neural de Hopfield recorrente [Tagliarini, 1991] formada por n neurônios de Hopfield todos interconectados, usada para resolver problemas NP-completos. A rede dessa figura possui um vetor de entradas externas $I = (I_1,..., I_n)$, um vetor de entradas internas $\theta = (V_1,..., V_n)$, um vetor de peso para as entradas externas $W = (w_1,...,w_n)$, um vetor de peso para as entradas internas $T = (T_{1,1},..., T_{n,n})$ e um vetor de saída $V = (V_1,..., V_n)$.

O comportamento do neurônio i de Hopfield é caracterizado pelo seu nível de ativação U_i (eq.3.3) [Hopfield, 1984][Tagliarini, 1991]. Hopfield [Hopfield, 1984] sugeriu uma função de Lyapunov (eq.3.4) [Hagedorn, 1984], denominada por ele de função de energia, para representar os estados dos neurônios com função de ativação tangente hiperbólica da sua rede considerando que $\tau = \frac{1}{RC}$ é muito grande.

$$\frac{dU_j}{dt} = -\tau U_i + \sum_{j=1}^{n} T_{j,i} V_i + I_i w_i \qquad \text{(eq.3.3)}$$

$$E = -\frac{1}{2}\sum_{i=1}^{n}\sum_{\substack{j=1 \\ j\neq i}}^{n} T_{i,j} V_i V_j - \sum_{i=1}^{n} I_i w_i V_i + \sum_{i=1}^{n} U_i V_i \qquad \text{(eq.3.4)}$$

Usando a eq.3.4, Hopfield mostrou que a sua rede neural possuí pontos limites estáveis. Para a função da eq.3.4, pode-se definir a variação da energia da rede em relação a variação da saída do neurônio V_i, como:

$$\frac{\Delta E}{\Delta V_i} = -\left[\sum_{j=1}^{n} T_{j,i} V_j + I_i w_i - U_i\right] \qquad \text{(eq.3.5)}$$

Hopfield usou um algoritmo para alterar V_i na direção de menor energia. Na eq.3.5 o termo entre colchetes e a variação de estado do neurônio ΔV_i devem ter o mesmo sinal. Usando a derivada da função de energia, Hopfield provou que a sua rede neural com função de ativação tangente hiperbólica também possui estados estáveis. Na sua demonstração, ele adicionou a integral da inversa da função hiperbólica à função de energia mostrada na eq.3.4. Como a inversa da função tangente hiperbólica é sempre crescente, Hopfield provou que a variação no tempo da função energia E tenderá para um mínimo, isto é, a sua rede neural tenderá para um estado estável [Hopfield, 1984].

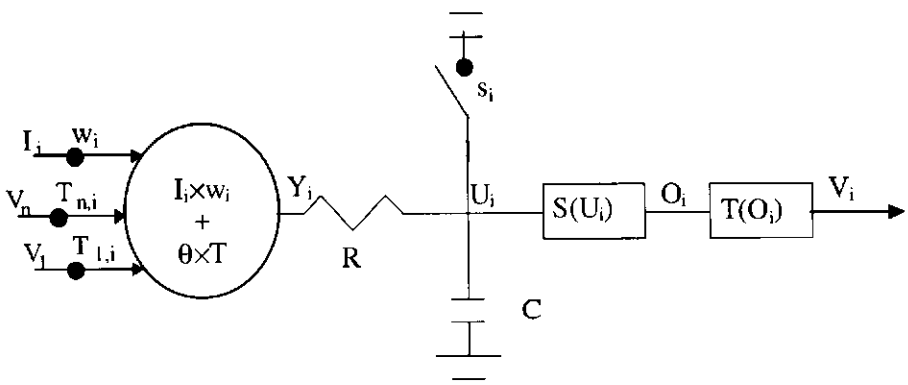


Fig. 3.5 - Neurônio de Hopfield com função de ativação tangente hiperbólica T(.)

Na forma discreta, o neurônio i da rede neural de Hopfield (Fig. 3.5) possui uma saída linear, Y_i, igual a:

$$Y_i = I_i w_j + \sum_{\substack{j=1 \\ j \neq i}}^{n} T_{j,i} V_j \qquad \text{(eq.3.6)}$$

O valor da integração do neurônio i, U_i, é obtido a partir da equação contínua eq.3.7 pelo método de Euler, método numérico para solução de equações diferenciais [Kopchenova, 1975]. Na eq.3.8 é mostrado a forma discreta da eq.3.7, com k representando o tempo discreto, 1/RC representando a constante de tempo do circuito RC (Resistor e Capacitor), h representando o valor do incremento de tempo e substituindo $h \times \frac{1}{RC}$ por β.

$$\frac{dU_i}{dk} = \frac{(Y_i - U_i)}{RC} \qquad \text{(eq.3.7)}$$

$$U_{k,i} = U_{k-1,i} + \beta(Y_{k-1,i} - U_{k-1,i}) \qquad \text{(eq.3.8)}$$

A saída intermediária do neurônio i, O_i, é igual ao valor da sigmóide da integração do neurônio i como mostra a eq.3.9, com η igual a valor da constante de declividade da curva.

$$O_i = S(U_i) = \frac{1}{1 + e^{-\eta U_i}} \qquad \text{(eq.3.9)}$$

A saída do neurônio i, V_i, é igual ao valor da tangente hiperbólica de O_i (eq.3.10).

$$V_i = T(O_i) = 2 \times O_i - 1 \qquad \text{(eq.3.10)}$$

Define-se como neurônio vencedor, aquele que após a competição tem $V_i \cong 1$. Em uma rede neural de Hopfield com dois neurônios, a competição ocorre quando dois neurônios têm o mesmo estado inicial e durante um certo tempo as saídas dos dois neurônios continuam com o mesmo sinal [McClelland, 1989].

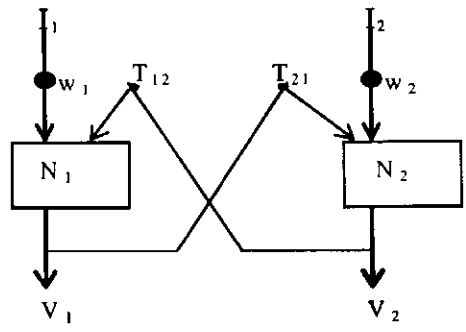


Fig. 3.6 - Rede Neural de Hopfield com 2 neurônios

A Fig. 3.7(a) apresenta os resultados obtidos durante a competição entre os neurônios da rede da Fig. 3.6 simulada em um microcomputador IBM PC (i486), usou-se $w_1 = w_2 = +1$, $T_{1,2} = T_{2,1} = -1$, $\beta = 0.02$, $\eta = 8$ e número de iteração ITER = 100. Observa-se que quando $I_1 > I_2$ o neurônio N_1 é o neurônio vencedor da competição. As curvas das saídas dos dois neurônios foram obtidas para $I_1 = 1$ e diferentes valores de I_2 (0.1, 0.5 e 0.9). Considerando-se o número de iterações menor do que 40, o efeito da competição ($I_1 > 0$ quando $I_2 > 0$) é evidente quando $I_1 = 1.0$ e $I_2 > 0.4$. Este efeito é negligenciável para $I_1 = 1.0$ e $I_2 < 0.3$. Observa-se que quando $I_1 < I_2$, o neurônio N_2 é o neurônio vencedor e a curva de saída é a mesma da Fig.3.7(a) considerando N_1 a linha pontilhada e N_2 a linha contínua. Quando o número de iterações é maior do que 200 e $I_1 > I_2$, a saída de N_1 é aproximadamente igual a 1. Quando β é constante, o valor final de saída de N_1 é diretamente proporcional ao número de iterações.

Na Fig.3.7(b) são mostrados os resultados obtidos com $w_1 = w_2 = 0.1$, $T_{1,2} = T_{2,1} = -1$, $\beta = 0.02$, $\eta = 8$, ITER = 50, $I_1 = 1.0$ e I_2 igual a 0.9, 0.5 e 0.1. Observou-se que a competição é negligenciável quando I_1w_1 e I_2w_2 estão próximos de zero.

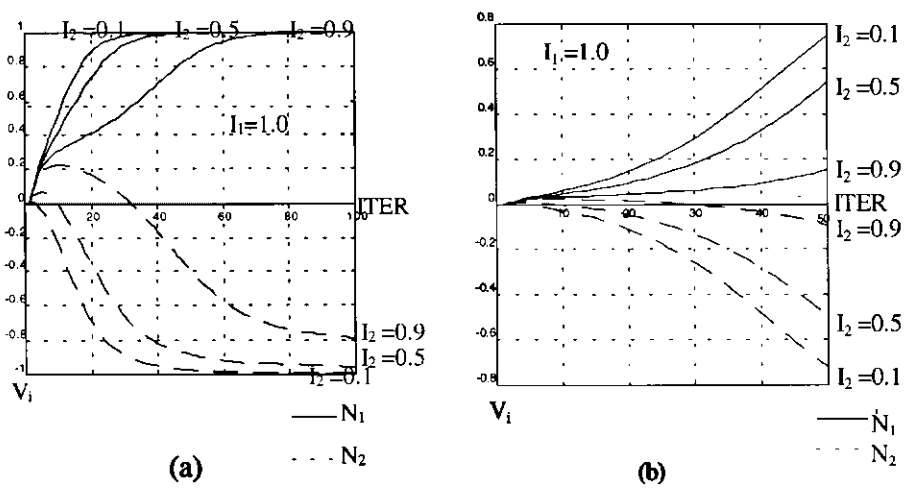


Fig. 3.7 - Competição dos neurônios da rede da Fig. 3.6

Se $I_2 > I_1$, e o número de iterações é maior que 100, o valor de saída N_2 se aproxima de um. Por outro lado, se $I_2 < I_1$ o valor de saída de N_2 se aproxima de menos um. Mas, se o número de iterações é menor que 100, o módulo do valor final de N_2 é menor que um.

A rede neural de Hopfield na forma k-winer é usada para solucionar problemas NP-completos [Tagliarini, 1991]. Os maiores valores do vetor de entrada correspondem às soluções de um problema NP-completo e são identificados através dos neurônios vencedores.

A rede neural de Hopfield usada no SINCA é emulada num microcomputador IBM PC (i486) DX4. A seguir apresenta-se um teorema, baseado na forma discreta da rede neural de Hopfield, que prova que a rede neural de Hopfield da Fig. 3.6 é conduzida a um estado estável, para um número de iterações adequado.

Teorema 3.3.1 O neurônio vencedor é o que possuí o maior valor U.

Demonstração: Suponha que na rede neural de Hopfield da Fig. 3.6 $I_1 > I_2$, $w_1 = w_2 = +1$, $T_{1,2} = T_{2,1} = -1$, e que inicialmente as chaves s_1 e s_2 são fechadas, o que implica o fato de $U_{0,1} = U_{0,2} = 0$. Durante a competição desses neurônios o valor das integrações $U_{n,1}$ e $U_{n,2}$, com $n > 0$, irão assumir diferentes valores. O valor de $U_{n,1}$ e $U_{n,2}$ para $n = 1$ será calculado a partir da eq.3.8 como mostram a eq.3.11 e a eq.3.12.

$$U_{1,1} = U_{0,1} + \beta \times \left(Y_{1,1} - U_{0,1}\right) \qquad \text{(eq.3.11)}$$

$$U_{1,2} = U_{0,2} + \beta \times \left(Y_{1,2} - U_{0,2}\right) \qquad \text{(eq.3.12)}$$

substituindo $Y_{1,1}$ na eq.3.11 e $Y_{1,2}$ na eq.3.12

$$U_{1,1} = U_{0,1} + \beta \times \left(I_1 \times w_1 + T(O_2) \times T_{2,1} - U_{0,1}\right) \qquad \text{(eq.3.13)}$$

$$U_{1,2} = U_{0,2} + \beta \times \left(I_2 \times w_2 + T(O_1) \times T_{1,2} - U_{0,2}\right) \qquad \text{(eq.3.14)}$$

mas como as funções sigmóide e tangente hiperbólica são crescentes, e $I_1 > I_2$, $w_1 = w_2 = +1$, $T_{1,2} = T_{2,1} = -1$, então $U_{1,1} > U_{1,2}$. O valor de $U_{n,1} > U_{n,2}$ para $n = 2$ será calculado a partir da eq.3.8, da mesma forma que foi calculado esses valores para $n = 1$, e levando em consideração que $U_{1,1} > U_{1,2}$, então $U_{2,1} > U_{2,2}$. O valor de $U_{n,1} > U_{n,2}$ para $n = 3$ será calculado a partir da eq.3.8, da mesma forma que foi calculado esses valores para $n = 1$, e levando em consideração que $U_{2,1} > U_{2,2}$, então $U_{3,1} > U_{3,2}$. Por indução o valor das integrações $U_{n,1}$ e $U_{n,2}$ para $n > 3$ será $U_{n,1} > U_{n,2}$. A variação dos valores de $U_{n,1} > U_{n,2}$ irá se repetir até que $T(U_{n+1,1}) = T(U_{n,1})$ e $T(U_{n+1,2}) = T(U_{n,2})$, ou seja, até que a saída do neurônio N_1 atinja o ponto limite +1 da função tangente hiperbólica ou a saída do neurônio N_2 atinja o limite -1 da função hiperbólica. Nesse instante o processo de competição dos neurônios N_1 e N_2 terá terminado, sendo o valor de $V_1 > 0$ e $V_2 < 0$.■

Os resultados do **Teorema 3.3.1** pode ser facilmente estendido para um número n de neurônios, desde que cuidados sejam tomados com os valores de w_i. Isto é, os valores de w_i devem ser tais que $I_i w_i$ não pode assumir valores próximos de zero. Quando $I_i w_i$ assume valores próximos de zero, a competição dos neurônios da rede será negligenciável.

3.4 Implementação do Algoritmo das Médias em uma Rede Neural de Hopfield

Para a construção da árvore filogenética a partir de informações com e sem conflito, usou-se uma rede neural de Hopfield formada por uma série de *n* neurônios interligados, onde *n* é a dimensão da matriz distância *D* a ser analisada. Na rede neural de Hopfield implementada, o neurônio i, $1 \leq i \leq n$, possui uma função de ativação tangente hiperbólica. Os pesos das entradas externas w_i são fixados em 0.1 e os pesos das entradas internas T_{ij} são calculados a partir do número de neurônios da rede, como é mostrado na eq.3.15. Na eq.3.15 float é uma função que recebe um número real e devolve a parte fracionária deste número.

$$T_{ij} = \text{float}\left(\frac{n}{10}\right) \qquad \text{(eq.3.15)}$$

A entrada externa I_i do neurônio i da rede é igual ao inverso do elemento d_{ji} eq.3.16. Dessa forma, garante-se que $I_i w_i \leq 0.1$ para o maior I_i. O maior I_i corresponde ao menor valor armazenado na linha j da matriz analisada.

$$I_i = \begin{cases} \dfrac{1}{d_{ji}} & , \text{se } i \neq j \\ 0.0 & , \text{se } i = j \end{cases} \qquad \text{(eq.3.16)}$$

Para garantir que ao final da competição dos neurônios da rede só um neurônio seja o vencedor, usou-se uma entrada adicional em cada neurônio com valor constante igual a -0.5, denominada excitação. O valor da excitação irá depender do número de neurônios da rede, mas se o número de neurônios for menor ou igual a 10 então pode-se usar excitação igual a -0.5.

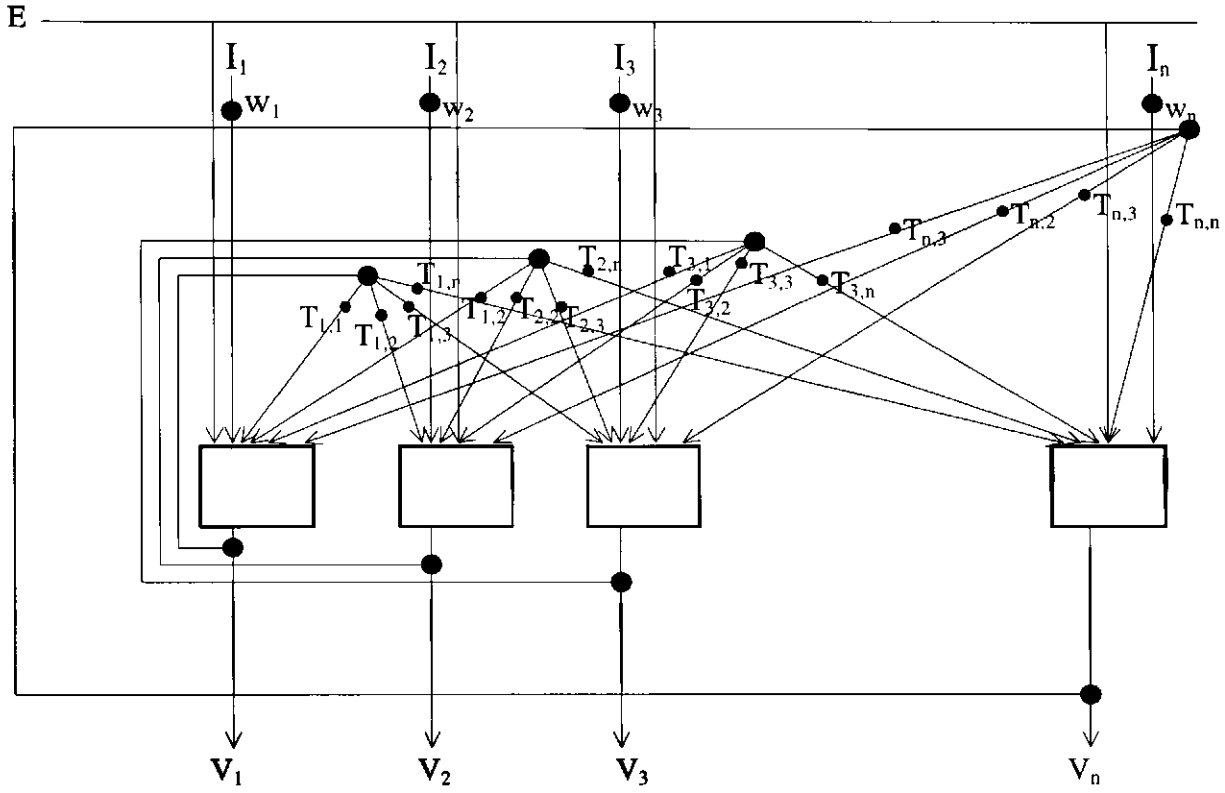


Fig. 3.8 - Rede implementada para construir árvores filogenéticas com e sem filogenia perfeita

A árvore filogenética construída com a rede neural de Hopfield (Fig.3.8) é formada por um conjunto de super-espécies encontradas pela rede e confirmadas pelo usuário. Uma super-espécie (j, i) é recomendado pela rede neural de Hopfield se o neurônio i for um dos neurônios vencedores da competição entre os elementos da j linha e vice-versa.

O algoritmo que supervisiona a interação entre o usuário e o programa que implementa a rede neural de Hopfield é chamado de Algoritmo Neural para a Construção da Árvore filogenética (ANCA). A árvore filogenética construída com o auxílio da rede neural de Hopfield dependerá da interação entre o sistema e o usuário. Nesta interação o usuário é responsável por:

- fornecer a matriz distância que será analisada pela rede neural;
- iniciar o processo de competição dos neurônio da rede neural;
- confirmar a escolha das super-espécies encontradas pela rede neural.

A seguir, descreve-se o ANCA.

1. lê os parâmetros da rede neural de Hopfield
2. lê a matriz distância
3. ajusta o valor dos elementos da matriz distância para servirem de entrada à rede
4. para j variando de 1 à n
   inicie a competição dos elementos da linha j
   guarde os pares (j, i), onde i é um dos neurônios vencedores
5. para j variando de 1 à n
   para i variando de j à n
   se os pares (j, i) e (i, j) corresponderem a neurônios vencedores
   então crie uma super-espécie X = (j, i)
6. guarde as super-espécies criadas que possuem a menor distância
7. escreve na tela as super-espécies construídas e guarde-as em um vetor A
8. pede-se que o usuário confirme as super-espécies construídas
9. lê as super-espécies confirmadas pelo usuário
10. constrói uma nova matriz distância
11. se o número de espécies for maior que 2
    então volte ao passo 1
    senão construa a árvore filogenética com as super-espécies guardadas em A.

O valor dos parâmetros default da rede do ANCA foram determinados experimentalmente. Inicialmente, construíu-se uma rede com 2 neurônios. Atribuiu-se para o vetor de entrada externa os vetores (0.0, 0.1) e (0.1, 0.0), para o número de iterações o valor 100, e para o fator de Euler o valor 0.01. Variou-se o valor da excitação dentro do intervalo de [-1, -0.1] até que o neurônio vencedor para o vetor de entrada externa (0.0, 0.1) fosse o neurônio 2 e o neurônio vencedor para o vetor de entrada externa (0.1, 0.0) fosse o neurônio 1. Durante essa emulação observou-se que para os valores de excitação dentro do intervalo [-0.5, -0.1], o neurônio vencedor para o vetor de entrada externa (0.0, 0.1) é o neurônio 2 e o neurônio vencedor para o vetor de entrada externa (0.1, 0.0) é o neurônio 1.

O passo seguinte foi a construção de uma rede neural com n neurônios. Atribuí-se para o vetor de entrada externa os vetores (0.0, 0.1, 0.2, ..., n-1) e (n-1, ..., 0.2, 0.1, 0.0), para o número de iterações o valor 100, para o fator de Euler o número 0.01. Variou-se o valor da excitação dentro do intervalo de [-1, -0.1] até que o neurônio vencedor para o vetor de entrada externa (0.0, 0.1, ..., 0.n-1) fosse o neurônio n e o neurônio vencedor para o vetor de entrada externa (0.n-1, ..., 0.1, 0.0) fosse o neurônio 1. Durante essa emulação observou-se que quando a rede construída apresentava só o neurônio vencedor n e 1 respectivamente para os vetores de entrada externa (0.0, 0.1, 0.2, ..., n-1) e (n-1, ..., 0.2, 0.1, 0.0), com $2 \leq n \leq 10$, o valor da excitação pertence ao intervalo D_1 = [-0.5, -0.1] para n = 2, D_2 =[-0.6, -0.1] para n = 3, e assim sucessivamente. O valor da excitação da rede será então igual ao maior valor do intervalo formado pela interseção aos intervalos D_i.

O valor dos parâmetros da rede neural de Hopfield depende do número de neurônios da rede e dos valores dos elementos do vetor de entrada externa. Assim, para usar estar rede na construção de uma árvore filogética para um conjunto de n espécies, com n > 10, o usuário deverá primeiro determinar os parâmetros da rede executando o mesmo procedimento descrito acima. Após o usuário determinar o valor dos parâmetros da rede neural Hopfield para o conjunto de espécie de seu interesse, ele poderá iniciar a execução do programa que executa o algoritmo do ANCA.

3.5 Construção de uma Árvore Filogenética Usando o ANCA

Considerando que o filogeneticista está investigando o conjunto de espécies {A, B, C, D} e que a matriz distância desse conjunto de espécies é mostrada na Tabela 3.1 (que é uma cópia da Tabela 2.6). Para que o filogeneticista possa construir a árvore filogenética das espécies A, B, C e D, usando a rede neural de Hopfield descrita na Seção 3.3 ele deverá executar um programa que realiza os passos do ANCA (tree.exe) a partir do prompt do DOS, como mostrado abaixo.

C:\>tree

	A	B	C	D
A	0	1	4	5
B	1	0	3	4
C	4	3	0	3
D	5	4	3	0

Tabela 3.1 - Matriz distância fornecida pelo usuário para o programa tree.exe

A seguir, o ANCA apresentará a Tela do Menu Principal da Fig. 3.9. Digitando os número das opções do menu desta tela o usuário poderá construir uma árvore filogenética com o ANCA.

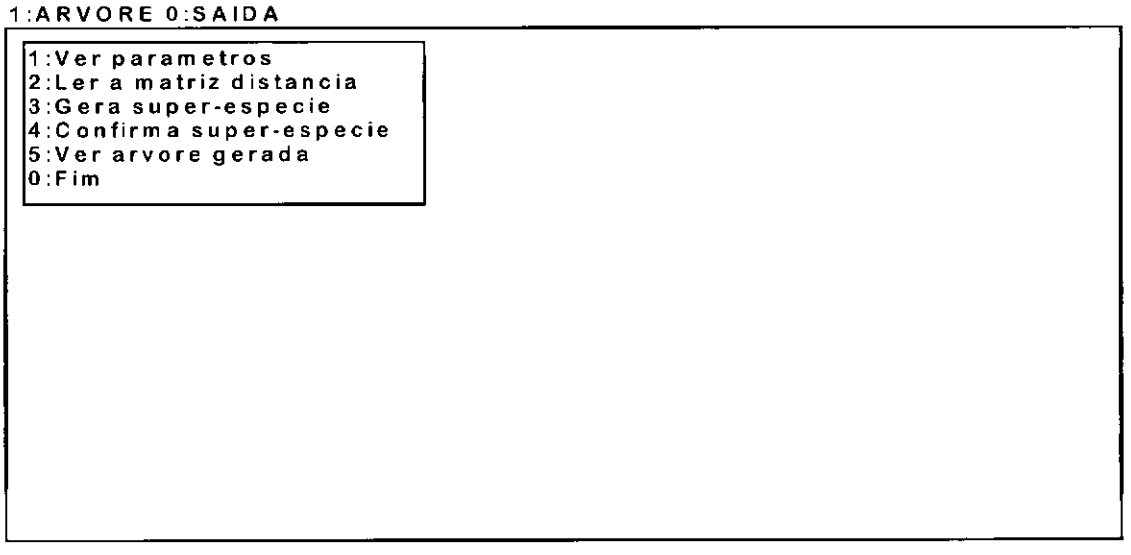


Fig. 3.9 - Tela do Menu Principal

Supondo que o usuário digite o número 1 que corresponde a opção "Ver parâmetros" da Tela do Menu Principal do ANCA. O ANCA exibirá a Tela de Entrada/Saída de Parâmetros da rede neural de Hopfield apresentada na Fig. 3.10. O usuário deverá digitar "s" ou "S" se desejar alterar os parâmetros da rede neural de Hopfield. Se o usuário digitar "s" ou "S", ele poderá alterar os valores padrões do número de iterações, do fator de Euler, de Beta e da Excitação do ANCA (ver seção 3.3.2).

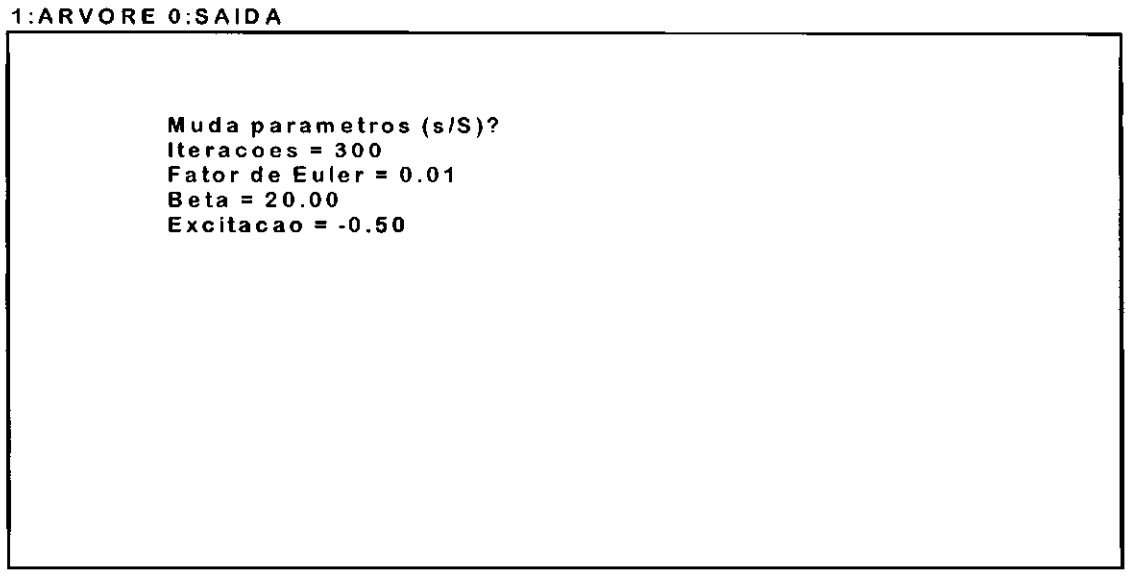


Fig. 3.10 - Tela de Entrada/Saída dos Parâmetros da rede neural de Hofield

Supondo que o usuário digite o número 2, que corresponde a opção "Ler a matriz distância", da Tela do Menu Principal do ANCA. O ANCA apresentará a Tela de Entrada do Nome do Arquivo da matriz distância (Fig. 3.11). O arquivo tem o nome distan*.dat, e * representa a dimensão da matriz distância.

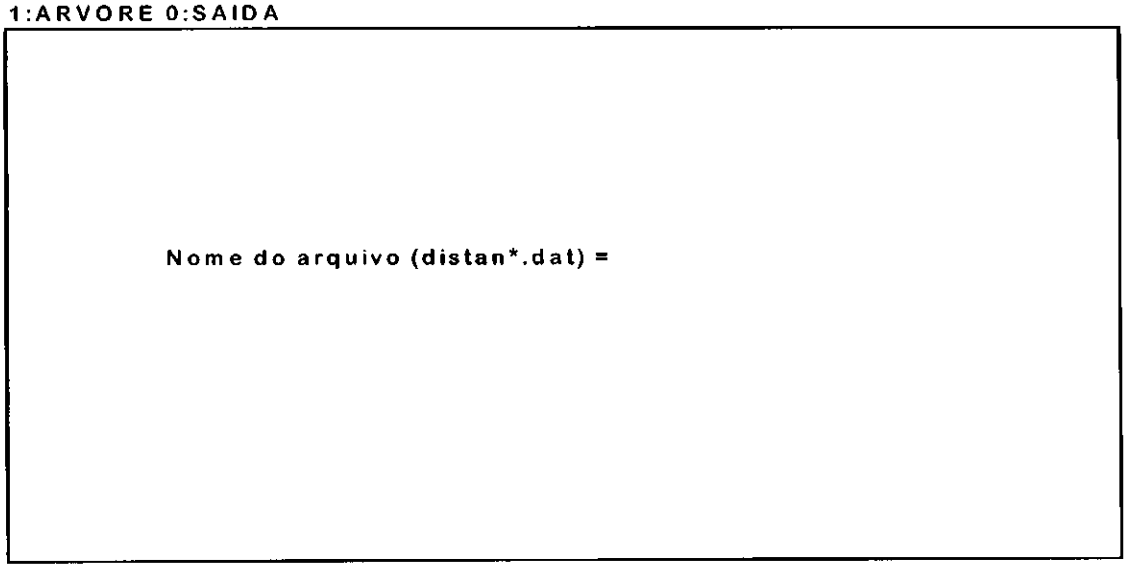


Fig. 3.11 - Tela de Entrada do Nome do Arquivo

Após o usuário teclar distan4.dat, que é o nome do arquivo que contém a matriz distância apresentada na Tabela 3.1, o ANCA lerá o arquivo distan4.dat, obtendo a matriz distância da Tabela 3.1.

A seguir, supondo-se que o usuário digite o número 3, que corresponde a opção "Gerar super-espécie", da Tela do Menu Principal do ANCA. O ANCA usará o inverso do valor dos elementos da matriz distância como entradas I_i da rede neural de Hopfield e atribuirá 0.1 aos pesos das entradas externas w_i, 0.4 aos pesos das entradas internas T_{ij}, -0.5 para o valor da excitação, 300 para o número de iterações, e 0.0 para os valores de saídas dos neurônios (ver etapa 1 a 3 do ANCA). Em seguida, nas etapas 4 a 5 do ANCA, serão analisadas todas as linhas da matriz distância.

A primeira linha a ser analisada será a linha 1 da matriz distância, o vetor de entrada externa correspondente a esta linha é (0.00, 1.00, 0.25, 0.20). Para este vetor de entrada é criada uma rede neural de Hopfield formada por 4 neurônios. A seguir, inicia-se a competição dos neurônios da rede criada durante 300 iterações. Os resultados da competição, usando o vetor de entrada externa (0.00, 1.00, 0.25, 0.20), são apresentados na Fig. 3.12 na forma de círculos e na forma de curvas.

Na Fig. 3.12 a seguir os círculos são numerados da esquerda para a direita e representam os valores das saídas dos neurônios da rede neural de Hopfield após o processo de competição. Os círculos preenchidos com a cor branca correspondem aos neurônios com valor de saída menor do que 0. Os círculos preenchidos com a cor preta correspondem aos neurônios com valor de saída maior do que 0. Os diâmetros dos círculos são diretamente proporcionais aos valores das saídas dos neurônios. Observa-se que o neurônio 2 foi o neurônio vencedor. As curvas da Fig. 3.12 são numeradas da esquerda para a direita de cima para baixo e representam a trajetória descrita pelos neurônios da rede neural de Hopfield criada durante o processo de competição. Observa-se que inicialmente todos os neurônios tem valor menor do que 0 devido a excitação (-0.5). O valor das saídas dos neurônios da rede serão alterados durante o processo de competição até a rede neural de Hopfield atingir um estado estável. Quando a rede neural de Hopfield atinge um estado estável, a saída dos neurônios V_1 = (-0.98, 0.99, -0.98, -0.98) não se alteram mais.

Após a criação das super-espécies possíveis para as linhas da matriz distância da Tabela 3.1 o ANCA verificará a menor distância mútua apresentada por estas super-espécies. As super-espécies que apresentarem a menor distância mútua serão guardados pelo ANCA como o conjunto de todas as super-espécies possíveis para a matriz distância da Tabela 3.1. A super-espécie (A,B) tem distância mútua 1 e a super-espécie (C,D) tem distância mútua 3. O ANCA escolherá a super-espécie (A,B).

Supondo que o usuário digite o número 4, que corresponde a opção "Escolhe uma super-espécie", da Tela do Menu Principal. O ANCA exibirá uma tela informando ao usuário o conjunto de todas as super-espécies construídas para a matriz distância analisada (Fig. 3.13) e solicita que o usuário forneça o número da(s) super-espécie(s) que melhor reflete suas hipóteses sobre a evolução do conjunto de espécies investigadas.

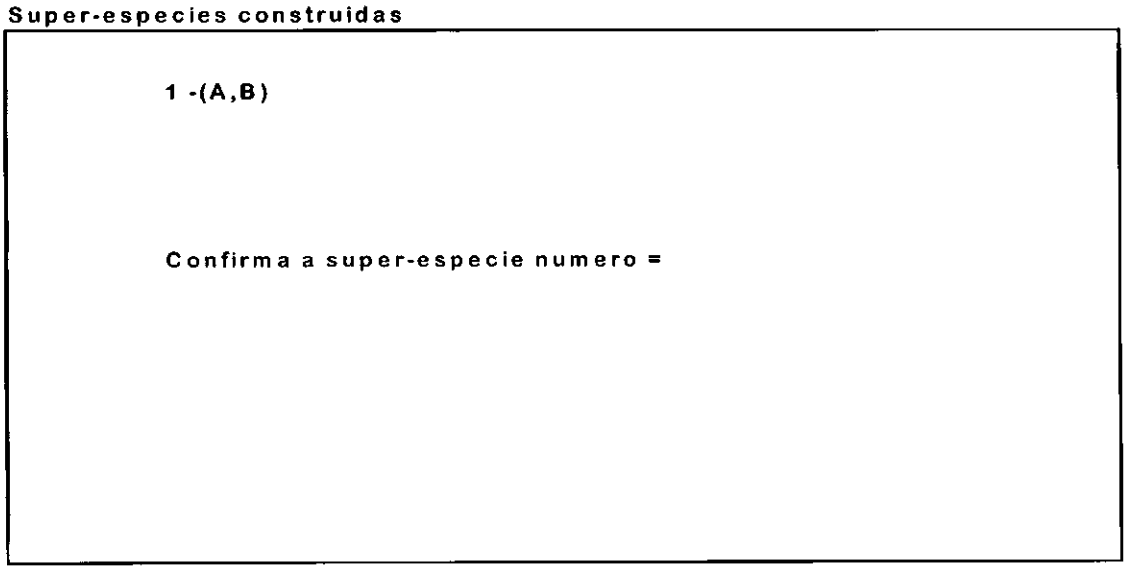


Fig. 3.13 - Tela de saída das super-espécies construídas a partir da análise da Tabela 3.1

Considerando que o usuário escolheu a super-espécie (A,B). Com base na escolha do usuário o ANCA irá gerar a próxima matriz distância (Tabela 3.3) para depois gravá-la no arquivo distan3.dat.

	S_1	C	D
S_1	0	3.5	4,5
C	3,5	0	3
D	4,5	3	0

Tabela 3.3 - Matriz distância reduzida a partir da Tabela 3.1

Supondo que o usuário digite o número 2, que corresponde a opção "Ler a matriz distância", da Tela do Menu Principal do ANCA. O ANCA apresentará a Tela de Entrada do Nome do Arquivo da matriz distância.

Após o usuário teclar distan3.dat, que é o nome do arquivo que contém a matriz distância apresentada na Tabela 3.3, o ANCA lerá o arquivo distan3.dat, obtendo a matriz distância da Tabela 3.3.

A seguir, supondo-se que o usuário digite o número 3, que corresponde a opção "Gerar super-espécie", da Tela do Menu Principal do ANCA. O ANCA usará o inverso do valor dos elementos da matriz distância como entradas I_i da rede neural de Hopfield e atribuirá 0.1 aos pesos das entradas externas w_i, 0.3 aos pesos das entradas internas T_{ij}, -0.5 para o valor da excitação, 300 para o número de iterações, e 0.0 para os valores de saídas dos neurônios (ver etapa 1 a 3 do ANCA). Em seguida, nas etapas 4 a 5 do ANCA, serão analisadas todas as linhas da matriz distância.

A Tabela 3.4 descreve os próximos passos executados pelo ANCA. A primeira coluna dessa tabela apresenta o número da linha analisada; a segunda coluna apresenta o vetor de entrada externa fornecido a rede neural de Hopfield formada por 3 neurônios; a terceira coluna apresenta o vetor de saída da rede neural de Hopfield formada por 3 neurônios após a competição entre esses neurônios durante 300 iterações; a quarta e última coluna apresenta as hipóteses construídas.

Linha analisada	Vetor de entrada externa	Vetor de saída	Hipóteses construídas
1 (S_1)	(0.00, 0.28, 0.22)	(-0.99, 0.65, -0.95)	(S_1, C)
2 (C)	(0.28, 0.00, 0.33)	(-0.99, -0.99, 0.84)	(C, D)
3 (D)	(0.28, 0.33, 0.00)	(-0.99, 0.85, -0.99)	(C, D)

Tabela 3.4 - Resumo dos passos do ANCA para a análise da Tabela 3.3

Após a criação das super-espécies possíveis para as linhas da matriz distância da Tabela 3.3 o ANCA verificará a menor distância mútua apresentada por estas super-espécies. As

super-espécies que apresentarem esta menor distância mútua serão guardados pelo ANCA como o conjunto de todas as super-espécies possíveis para a matriz distância da Tabela 3.3.

Supondo que o usuário digite o número 4, que corresponde a opção "Escolhe uma super-espécie", da Tela do Menu Principal. O ANCA exibirá uma tela informando ao usuário o conjunto de todas as possíveis super-espécies para a matriz distância analisada (Fig. 3.14) e solicita que o usuário forneça o número da(s) super-espécie(s) que melhor reflete suas hipóteses sobre a evolução do conjunto de espécies investigadas.

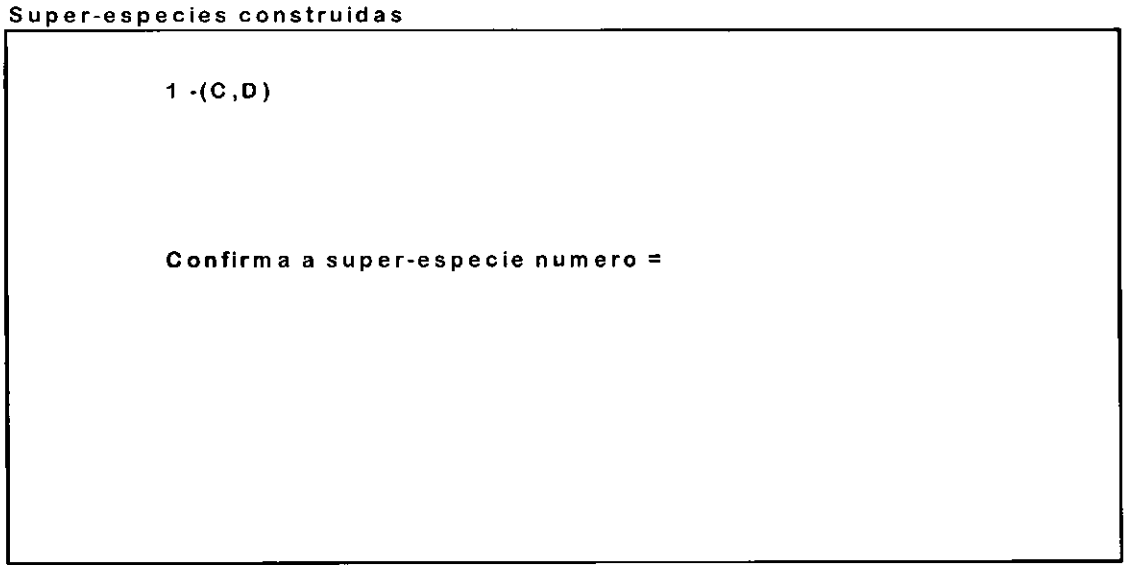


Fig. 3.14 - Tela de saída das super-espécies construídas a partir da análise da Tabela 3.4

Considerando que o usuário escolheu a super-espécie (C,D). Com base na escolha do usuário o ANCA irá gerar a próxima matriz distância (Tabela 3.5) para depois gravá-la no arquivo distan2.dat.

	S_2	S_3
S_2	0	4
S_3	4	0

Tabela 3.5 - Matriz distância reduzida a partir da Tabela 3.3

Supondo que o usuário digite o número 2, que corresponde a opção "Ler a matriz distância", da Tela do Menu Principal do ANCA. O ANCA apresentará a Tela de Entrada do Nome do Arquivo da matriz distância.

Após o usuário teclará distan2.dat, que é o nome do arquivo que contém a matriz distância apresentada na Tabela 3.5, o ANCA lerá o arquivo distan2.dat, obtendo a matriz distância da Tabela 3.5.

A seguir, supondo-se que o usuário digite o número 5, que corresponde a opção "Ver árvore gerada", da Tela do Menu Principal do ANCA. O ANCA apresentará a Tela Árvore Gerada apresentada na Fig. 3.15. A Tela Árvore Gerada do ANCA apresenta a árvore filogenética construída pelas super-espécies encontradas pela rede e confirmados pelo usuário.

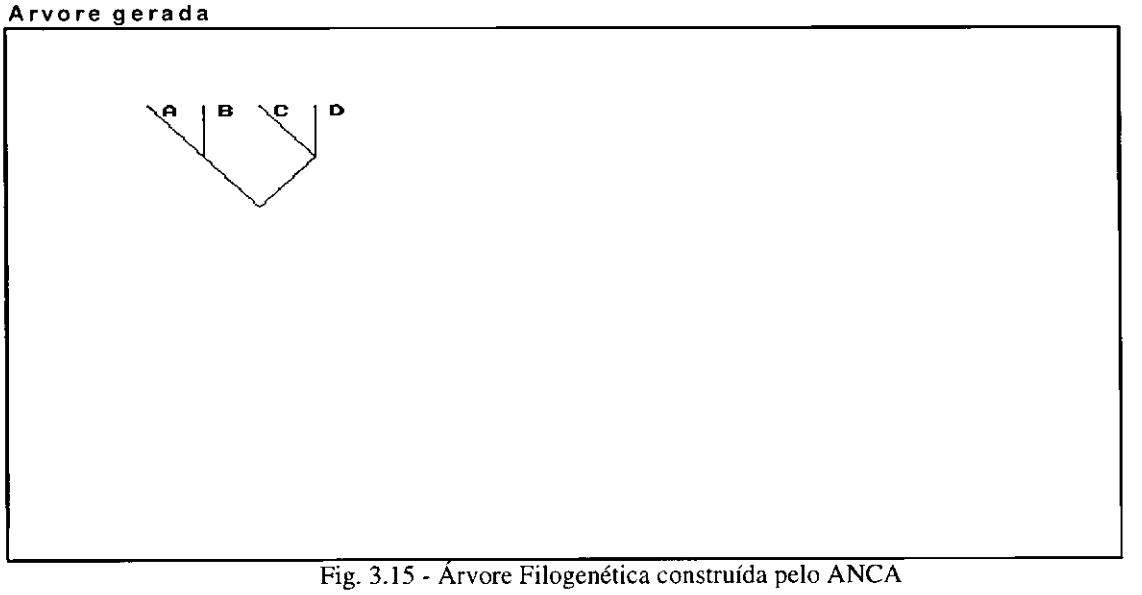


Fig. 3.15 - Árvore Filogenética construída pelo ANCA

No próximo capítulo será apresentado um sistema neurosimbólico desenvolvido para construir árvores filogenéticas com e sem filogenia perfeita.

4 Construção de Árvore Filogenética Usando um Sistema Neurosimbólico

4.1 Introdução

Neste capítulo apresentam-se o projeto e os resultados experimentais de um sistema neurosimbólico desenvolvido para construir uma árvore filogenética, com e sem filogenia perfeita, a partir do conhecimento fornecido pelo usuário.

4.2 Motivação

No capítulo anterior, usou-se uma abordagem conexionista na construção de árvores filogenéticas e se observou que a explicação fornecida por essa abordagem pode não ser compreensível para o usuário (filogeneticista). Além do que a abordagem conexionista solicita a intervenção do usuário na escolha de todas as super-espécies da árvore filogenética, mesmo quando o ANCA só construiu uma super-espécie para as espécies analisadas. Essa observação, justifica o desenvolvimento de um sistema neurosimbólico que combine a abordagem conexionista do Capítulo 3 a uma abordagem simbólica que forneça uma explicação compreensível para o usuário.

4.3 Sistema Neurosimbólico para Construção de Árvores Filogenéticas (SINCA)

Um sistema neurosimbólico é um ambiente computacional que integra técnicas simbólicas e técnicas conexionistas. O objetivo de um sistema neurosimbólico é aproveitar as vantagens das técnicas simbólicas e das técnicas conexionistas, já que as vantagens das técnicas simbólicas complementam as desvantagens das técnicas conexionistas e vice-versa [Andrade, 1997].

A Fig. 4.1 mostra a arquitetura básica do SINCA desenvolvido neste trabalho para construir árvore filogenéticas com e sem filogenia perfeita.

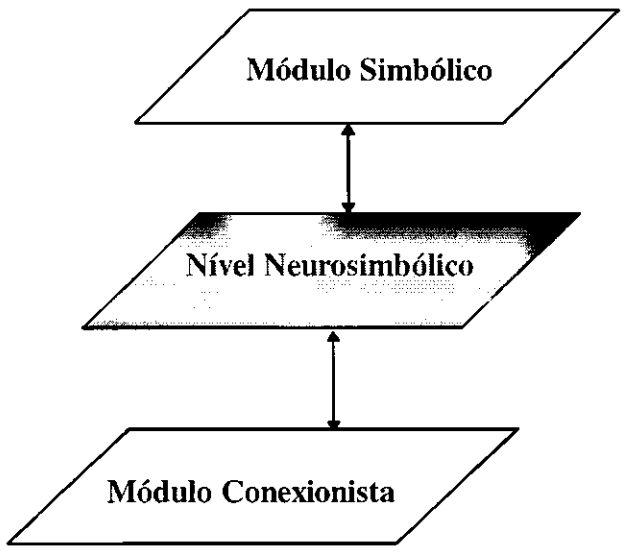


Fig. 4.1 - Arquitetura básica do SINCA

O módulo simbólico do SINCA contém um sistema especialista. O nível neurosimbólico do SINCA é composto por um módulo que envia um sinal de controle para o módulo conexionista e salva os resultados do módulo simbólico em um arquivo com extensão sim e por um módulo que envia um sinal de controle para o módulo simbólico e salva os resultados do módulo conexionista em um arquivo com extensão rnh. O módulo conexionista do SINCA é composto pela rede neural de Hopfield desenvolvida para construir árvores filogenéticas no Capítulo 3.

No SINCA os módulos simbólico e neurosimbólico foram implementados na linguagem LPA-Prolog. Enquanto que o módulo conexionista do SINCA foi implementado na linguagem C^{++}.

4.4 O Módulo Simbólico do SINCA

A Fig. 4.2 mostra a arquitetura de um sistema especialista baseado em regras [Giarratano, 1989]. O sistema especialista da Fig. 4.2, possui os seguintes módulos:

- *interface com o usuário*: é responsável pela supervisão da comunicação entre o usuário e o sistema;

- *facilidade de explanação*: é responsável pelo desenvolvimento de um encadeamento lógico do conhecimento utilizado pelo sistema na construção de uma hipótese;
- *memória de trabalho*: é um repositório dos fatos usados pelo sistema na construção de uma hipótese;
- *base de conhecimento*: é um repositório das regras usadas pelo sistema na construção de uma hipótese;
- *máquina de inferência*: é responsável pela construção da árvore filogenética usando as regras da base de conhecimento e os fatos da memória de trabalho;
- *aquisição de conhecimento*: é responsável pela entrada do conhecimento do usuário na base de conhecimento antes que o sistema construa sua hipótese.

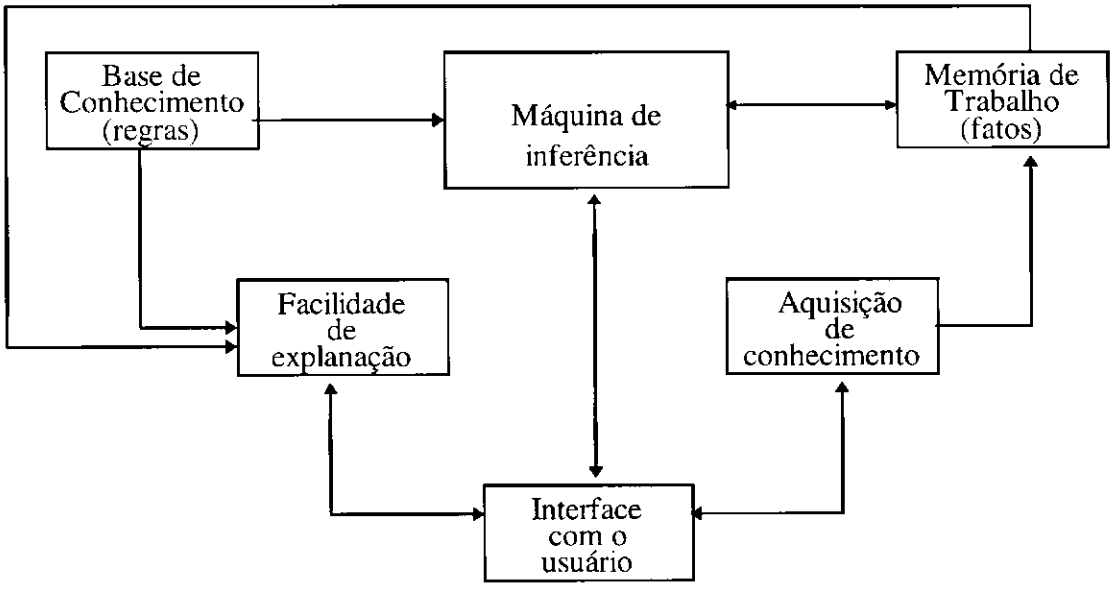


Fig. 4.2 - Arquitetura de um sistema especialista baseado em regras

4.4.1 Interface com o Usuário

Usou-se o padrão Windows para a interface entre o usuário e o SINCA. Quando o usuário, no ambiente Windows, acessa as opções do menu pull-down da janela principal (Fig. 4.3), ele poderá executar uma determinada tarefa. Se para a execução de uma tarefa, alguma informação for necessária, então o SINCA mostrará na tela uma janela de entrada de informação composta de campos para preenchimento, de listas com informações já fornecidas e de botões para que o usuário possa confirmar/cancelar a informação fornecida.

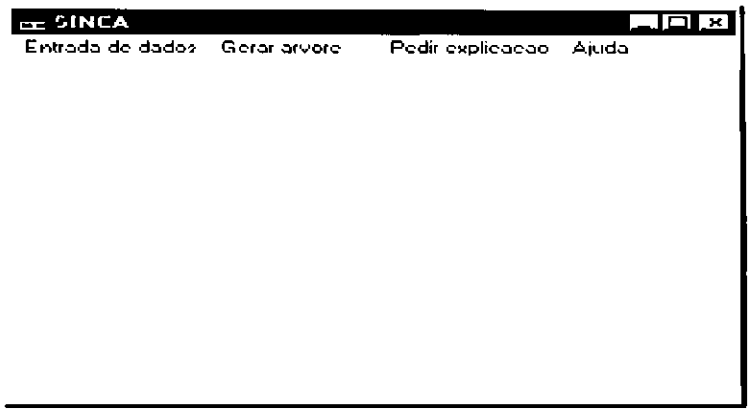


Fig. 4.3 - Tela Principal do SINCA

Mas, se o usuário estiver impossibilitado de executar a tarefa solicitada, o SINCA mostrará uma janela de alerta contendo um texto onde é explicado, ao usuário, os motivos da impossibilidade e um botão de confirmação. Quando o usuário pressionar esse botão, o sistema irá considerar que o usuário está ciente da situação.

A Fig. 4.4 mostra a hierarquia das opções e janelas do SINCA. Nessa figura adotou-se a seguinte representação:

- retângulos sombreados com cantos não arredondados para as janelas;
- retângulos sombreados com cantos arredondados para as opções;
- retângulos sem sombreado com cantos arredondados para as condições necessárias a execução de uma tarefa;
- linhas pontilhadas para as condições verdadeiras;
- linhas tracejadas para as condições falsas;
- linhas contínuas para ativação de processos.

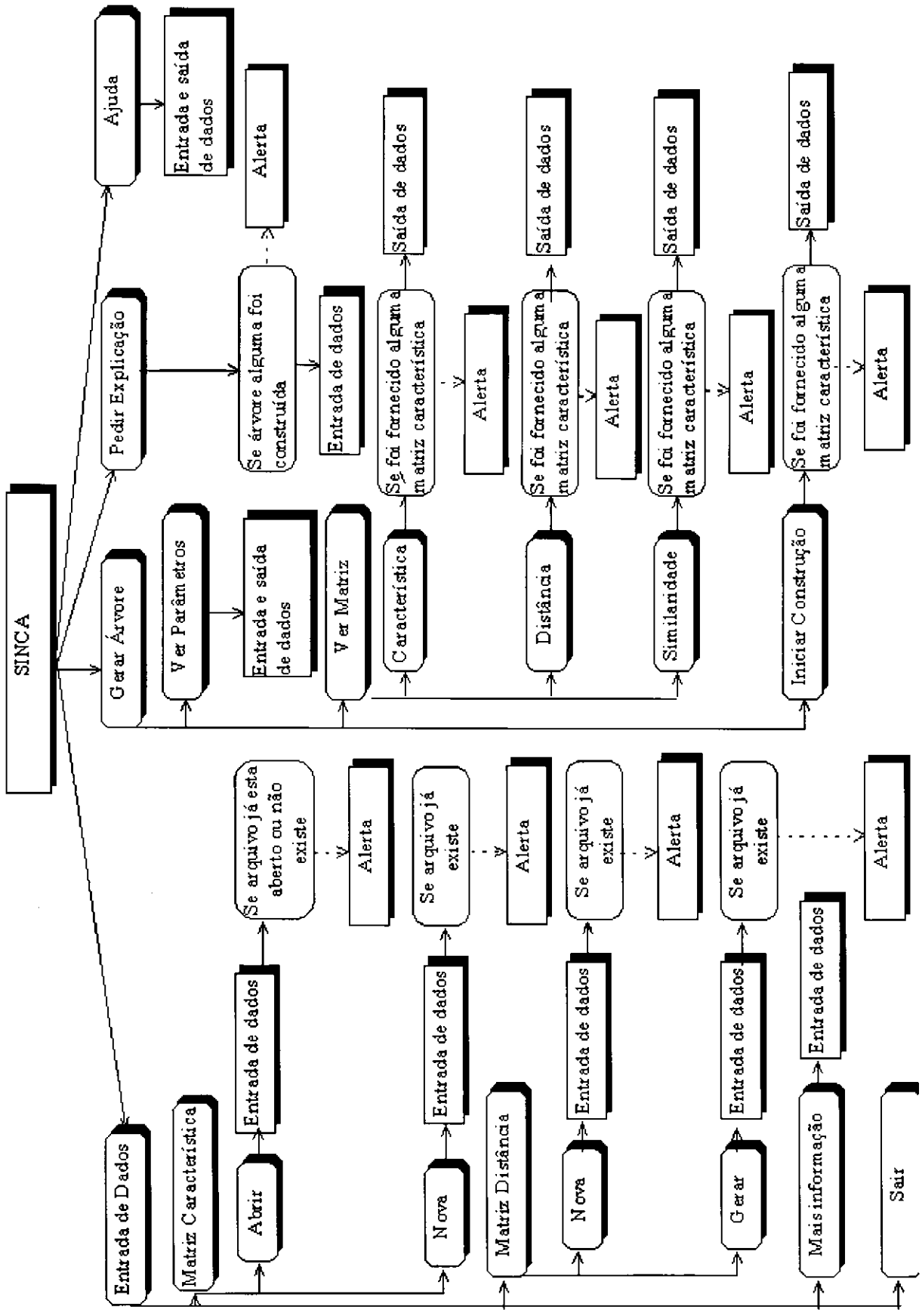


Fig. 4.4 - Hierarquia das opções e janelas do SINCA

4.4.2 Estrutura das Bases de Dados

Nesta parte do trabalho, serão apresentadas as estruturas da base de conhecimento e da memória de trabalho manipuladas pelo módulo simbólico do SINCA durante o processo de construção da árvore filogenética. As estruturas de dados utilizadas seguem o padrão do LPA-Prolog e são classificadas em:

- *fato*: é um conhecimento que expressa um acontecimento, que tem a seguinte sintaxe

$$\text{nome\_do\_fato}(\text{atributo}_1, \text{atributo}_2, ..., \text{atributo}_n).$$

onde se lê: o fato nome_do_fato é verdade se tiver os seguintes valores atributo_1, ..., atributo_n para os seus parâmetros, e

- *regra*: é um conhecimento que expressa uma condição, que tem a seguinte sintaxe

```
regra(Y):- X.
```

onde se lê: se a condição Y é verdadeira então execute X.

4.4.3 Facilidade de Explanação

Os seres humanos têm a capacidade de argumentar com outras pessoas, que dispõem aproximadamente do mesmo conhecimento acerca do problema em questão, sobre a precisão das suas soluções para um problema de maneira compreensível. A argumentação será realizada pelo módulo de facilidade de explanação do SINCA. Este módulo tem como premissa básica a seguinte assertiva: *toda decisão tomada pelo SINCA na construção de uma hipótese pode ser explicada com base nos fatos e regras disponíveis em suas bases de dados.* Assim, as explicações do SINCA ficam limitadas à estrutura dos fatos e das regras armazenados na sua memória de trabalho e na sua base de conhecimento respectivamente.

O SINCA oferece uma explicação para que o usuário tome conhecimento dos motivos que o levaram a construir uma determinada super-espécie.

4.4.4 Máquina de Inferência

A máquina de inferência do SINCA é formada por um conjunto de regras e pela rede neural de Hopfield. De modo que, se o número de espécies da matriz distância a ser analisada for menor do que 5 e se esta matriz não tiver conflito, então o SINCA utilizará o conjunto de regras de

sua base de conhecimento para construir as super-espécies que irão fazer parte da árvore filogenética correspondente a essa matriz.

O conjunto de regras do SINCA é listado abaixo. Note que as regras deste conjunto não consideram a possibilidade de existir conflito entre os dados analisados e são do tipo **if_then_else**.

R4.1 Avalia um conjunto com 2 espécies

se d_{12} = número de características analisadas

então crie uma super-espécie $S_1 = (O_1,\bullet)$ e $S_2 = (\bullet,O_2)$

senão crie uma super-espécie $S_1 = (O_1,O_2)$

R4.2 Avalia um conjunto com 3 espécies

se $d_{12} < d_{13}$ e $d_{12} < d_{23}$

então crie uma super-espécie $S_1 = (O_1,O_2)$, constrói uma nova matriz distância e aciona a rega **R4.1**

senão crie uma super-espécie $S_1 = (O_1,O_3)$, constrói uma nova matriz distância e aciona a regra **R4.1**

R4.3 Avalia um conjunto com 4 espécies

se $d_{12} < d_{13}$ e $d_{12} < d_{14}$ e $d_{12} < d_{23}$ e $d_{12} < d_{24}$ e $d_{12} < d_{34}$

então crie uma super-espécie $S_1 = (O_1,O_2)$, constrói uma nova matriz distância e aciona a regra **R4.2**

senão se $d_{13} < d_{12}$ e $d_{13} < d_{14}$ e $d_{13} < d_{23}$ e $d_{13} < d_{24}$ e $d_{13} < d_{34}$

então crie uma super-espécie $S_1 = (O_1,O_3)$), constrói uma nova matriz distância e aciona a regra **R4.2**

senão crie uma super-espécie $S_1 = (O_1,O_4)$), constrói uma nova matriz distância e aciona a regra **R4.2**

onde • indica que a super-espécie entre duas espécies está localizado em uma geração muito longe da geração dessas duas espécies.

Mas, se o número de espécies da matriz distância a ser analisada for maior ou igual à 5 ou se esta matriz tiver conflito, então o SINCA acionará o módulo conexionista para construir o conjunto de todas as possíveis super-espécies da matriz distância analisada.

Quando o módulo conexionista termina a construção do conjunto de super-espécies, ele envia um sinal de controle para o módulo simbólico do SINCA e grava no arquivo saida.rnh os elementos desse conjunto.

O módulo simbólico do SINCA ao receber o sinal de controle da rede neural de Hopfield lerá o arquivo saida.rnh. Se a cardinalidade[1] do conjunto das super-espécies for maior que 1, então o módulo simbólico do SINCA pesquisará na sua memória de trabalho à procura de um fato que justifique a escolha de um dos elementos do conjunto de super-espécies construídas. Os elementos do conjunto de super-espécies construídas pela rede neural de Hopfield são tratados internamento como fatos conhecimento_da_rede(Dimensao,(E1,E2)), onde o atributo Dimensao contém a dimensão da matriz distância analisada pela rede e o atributo (E1,E2) contém uma das super-espécies construídas pela rede neural de Hopfield. O conhecimento fornecido pelo usuário que prioriza uma relação de parentesco entre duas espécies é tratado internamente como um fato conhecimento_do_usuario((E3,E4), Justificativa), onde o atributo (E3,E4) contém uma das super-espécies cuja a relação de parentesco deve ser priorizada e o atributo Justificativa contém os motivos que justificam a priorização da relação de parentesco entre as espécies E3 e E4. O processo de pesquisa na memória de trabalho realizado pela máquina de inferência do módulo sombólico do SINCA à procura de um fato que justifique a escolha de uma das super-espécies construídas pelo módulo conexionista do SINCA nada mais é do que a busca de um fato conhecimento_da_rede(Dimensao,(E1,E2)) e conhecimento_do_usuário((E3,E4), Justificativa), onde os atributos (E1,E2) e (E3,E4) formem a mesma super-espécie, ou seja, (E1,E2)=(E3,E4).ou (E1,E2)=(E4,E3) .Se existir na memória de trabalho do módulo simbólico do SINCA algum fato que justifique a escolha de um dos elementos desse conjunto, então essa será a super-espécie construída. Senão, o módulo simbólico do SINCA mostrará na tela uma janela informando ao usuário as super-espécies possíveis com um botão de confirmação. Quando o usuário pressionar o botão de confirmação, o SINCA exibirá uma janela de entrada de dados para que o usuário faça a sua escolha e forneça o(s) motivo(s) que justifique(m) a sua escolha.

[1] Cardinalidade - Número de elementos de um conjunto.

A árvore filogenética construída pelo SINCA será então o resultado da combinação das super-espécies geradas pelo conjunto de regras e pela rede neural de Hopfield. A Fig. 4.5 descreve o comportamento da máquina de inferência do SINCA. Nessa figura adotou-se a seguinte representação:

- retângulos sombreados com cantos não arredondados para os processos;
- retângulos sem sombreado com cantos arredondados para as condições necessárias a execução de uma processo;
- linhas pontilhadas para as condições verdadeiras;
- linhas tracejadas para as condições falsas;
- linhas contínuas para ativação de processos.

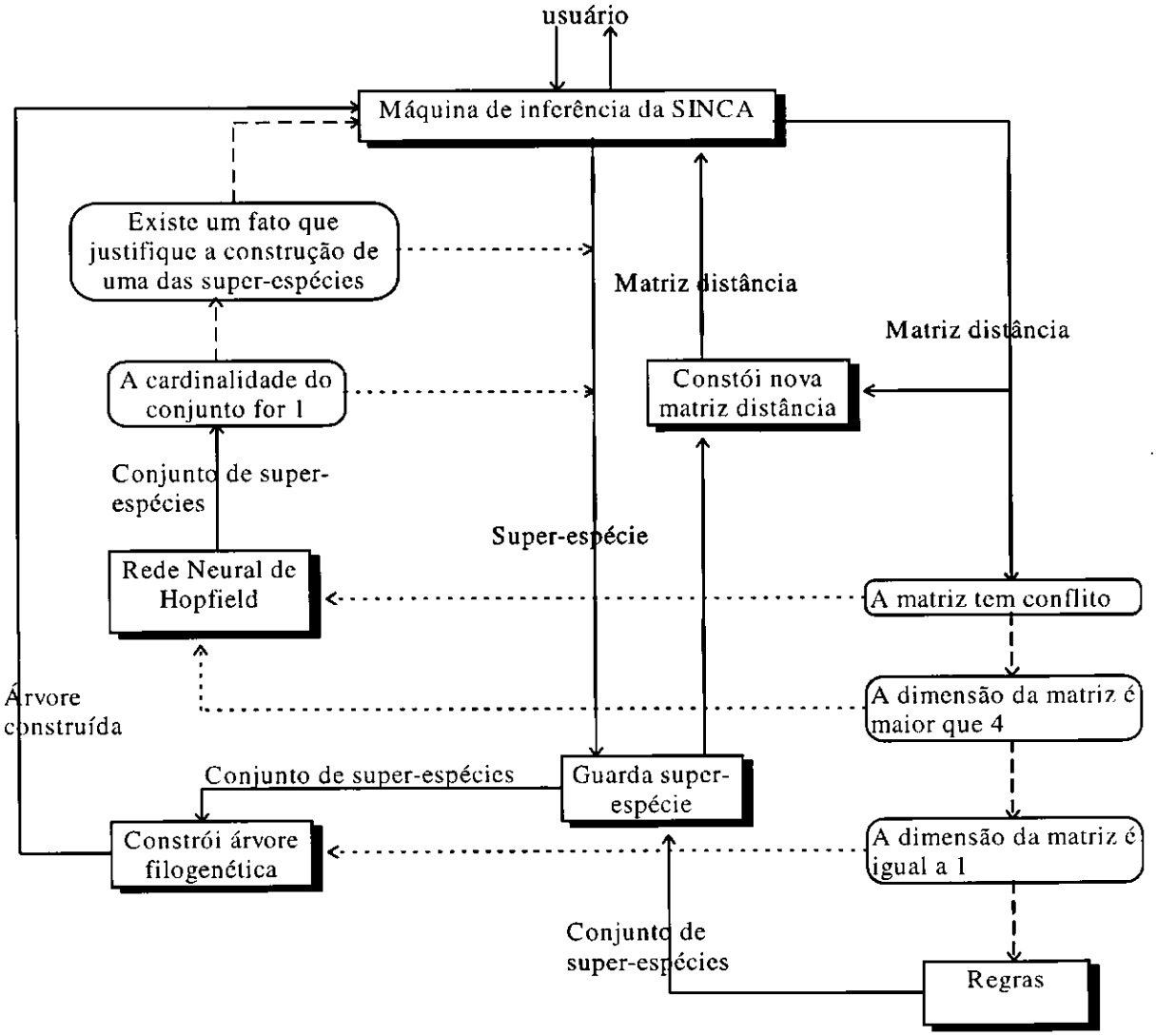


Fig. 4.5 - Comportamento da Máquina de Inferência do SINCA

4.4.5 Aquisição de Conhecimento

O módulo de aquisição de conhecimento do SINCA é usado para receber as informações necessárias às suas tomadas de decisão. No SINCA, este módulo permite a inclusão, exclusão e alteração dos fatos contidos na memória de trabalho que foram fornecidos pelo usuário.

A seguir será mostrado exemplos de construção de árvores filogenéticas com o SINCA.

4.5 Construção de uma Árvore Filogenética com o Auxílio do SINCA

Inicialmente o usuário executará a instrução de inicialização do SINCA. Após a execução dessa instrução, o módulo simbólico do SINCA apresentará a Tela Principal mostrada na Fig. 4.3. Acessando as opções desta tela, o usuário poderá fornecer os dados sobre as espécies investigadas ("Entrada de dados"), gerar uma árvore filogenética para essas espécies ("Gerar árvore"), pedir explicação sobre a geração das super-espécies presentes na árvore filogenética ("Pedir explicação") e pedir ajuda para executar uma análise ("Ajuda").

Inicialmente, usando a opção "Entrada de dados", o usuário fornecerá a matriz característica da Tabela 4.1 acessando a opção "Criar" do menu pull-down da opção "Matriz característica" (Fig. 4.6).

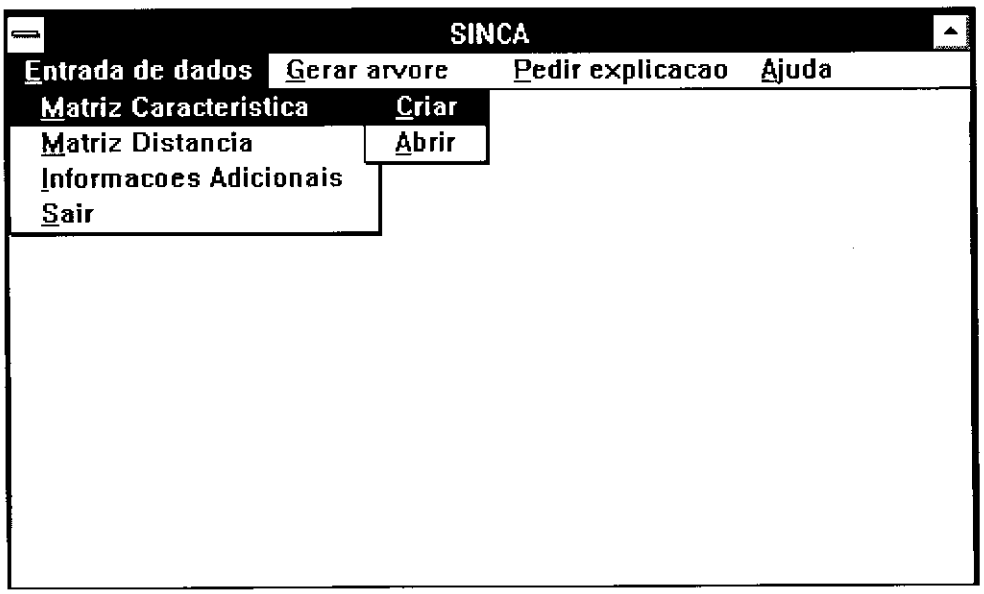


Fig. 4.6 - Menu pull-down da opção "Matriz característica"

	1	2	3	4	5	6	7
A	0	0	0	0	0	0	0
B	1	1	0	0	0	0	0
C	1	1	0	0	1	1	1
D	1	1	1	1	1	1	1

Tabela 4.1 - Matriz característica polarizada usada no primeiro exemplo do Capítulo 4

A criação de um novo arquivo para conter a matriz característica fornecida pelo usuário exige que o usuário antes de fornecer a matriz informe ao SINCA o nome do arquivo e do diretório que conterá essa matriz (Fig. 4.7).

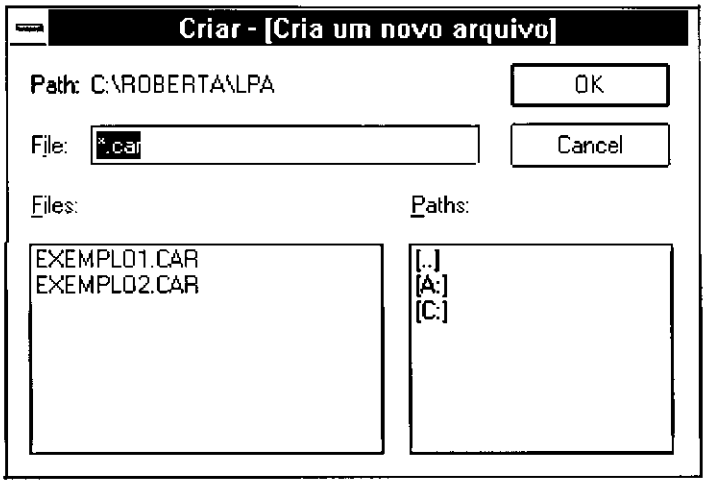


Fig. 4.7 - Entrada de dados para à matriz característica

Se o nome do arquivo já existir no diretório informado pelo usuário, o SINCA mostrará um janela de alerta informando para o usuário que já existe um arquivo com o nome fornecido e que o sistema não pode criá-lo novamente (Fig. 4.8).

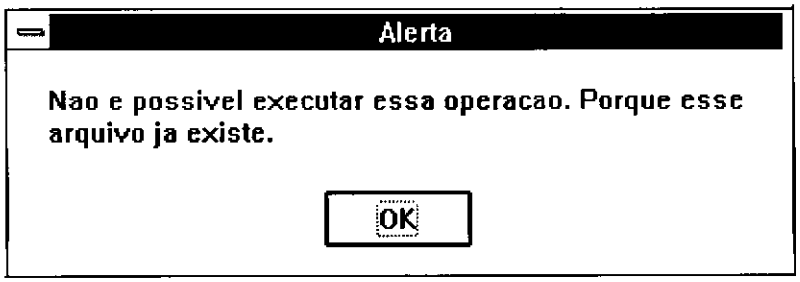


Fig. 4.8 - Janela de alerta

Quando o nome do arquivo e do diretório fornecido pelo usuário não existir no diretório informado, o módulo simbólico do SINCA criará o arquivo neste diretório e chamará o editor de texto WordPad para que o usuário entre com a matriz característica na forma de um conjunto de fatos como mostrado na Fig 4.9.

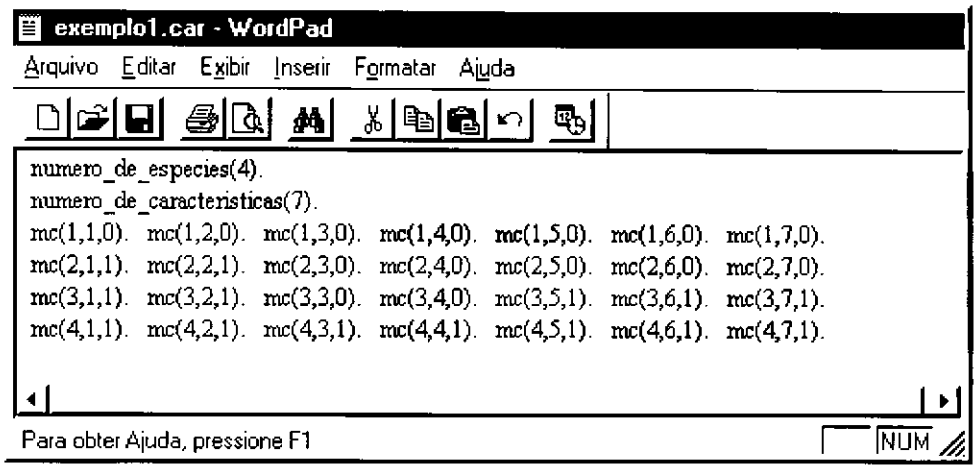


Fig. 4.9 - Entrada da matriz no editor WordPad

Após, a entrada da matriz característica o usuário deverá solicitar ao sistema que gere a matriz distância mostrada na Tabela 4.2 (Fig. 4.10).

	A	B	C	D
A	0	2	5	7
B	2	0	3	5
C	5	3	0	2
D	7	5	2	0

Tabela 4.2 - Matriz Distância gerada a partir da Tabela 4.1

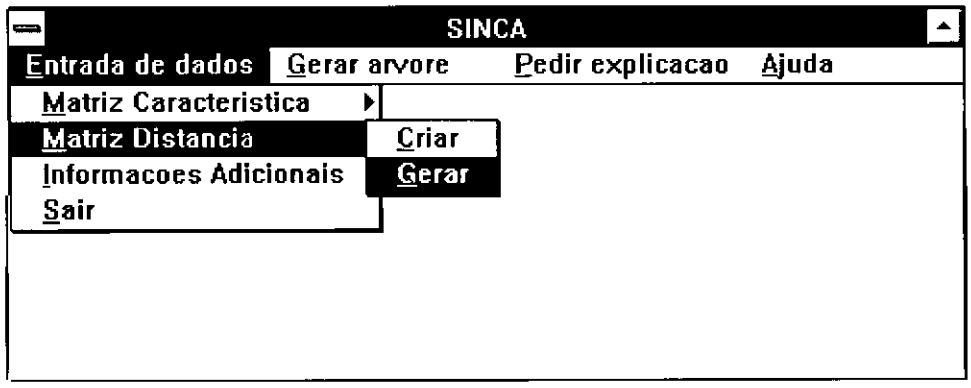


Fig. 4.10 - Menu pull-down da opção "Matriz distância"

Depois que o sistema tiver gerado a matriz distância, o usuário poderá iniciar a construção da árvore filogenética. Para isso, ele deve acessar a opção "Gerar árvore" da Tela Principal e escolher a opção "Ver resultados" do menu pull-down da opção "Gerar árvore" (Fig. 4.11).

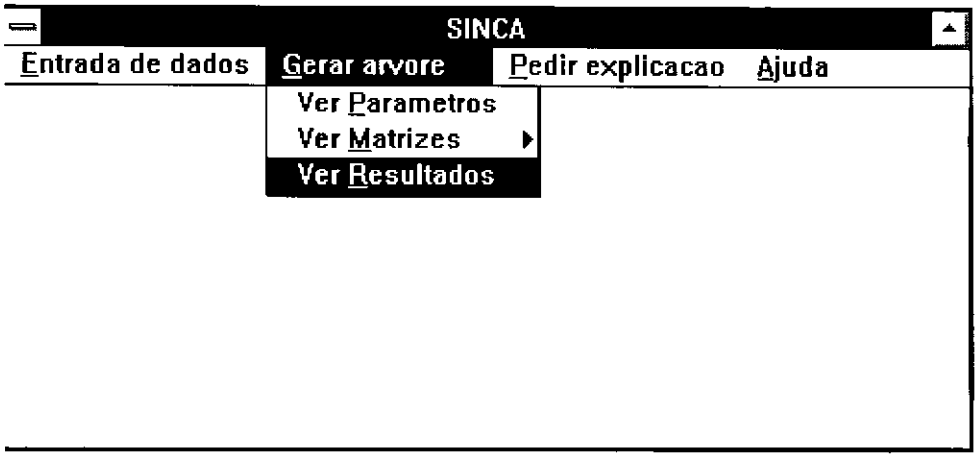


Fig. 4.11 - Menu pull-down da opção "Gerar árvore"

De posse da matriz distância da Tabela 4.2, inicialmente o módulo simbólico verificará se o número de espécies desta matriz é maior ou igual a 5 e se existe conflito entre os dados armazenados na matriz fornecida, então o módulo simbólico gravará no arquivo saida.sim a matriz distância fornecida na forma tabular e enviará um sinal de controle para o seu módulo conexionista. Quando o módulo conexionista receber o sinal de controle, ele inicia o processo de leitura da matriz distância gravada no arquivo saida.sim. Em seguida, o módulo conexionista analisará as espécies da matriz gravada nesse arquivo e construirá o conjunto de super-espécies {(A,B), (C,D)}. Após a construção do conjunto de super-espécies o módulo conexionista gravará este conjunto no arquivo saida.rnh na forma de fatos e enviará um sinal de controle para informar que já terminou a sua tarefa.

Ao receber o sinal de controle, o módulo simbólico lerá o conjunto de todas as super-espécies construídos do arquivo saida.rnh, para depois considerar esta informação na construção da árvore filogenética. Se a cardinalidade do conjunto de super-espécies for igual a 1, então o módulo simbólico criará a única super-espécie desse conjunto, senão o módulo simbólico pesquisará em sua memória de trabalho um fato que justifique a escolha de uma dessas super-espécies. Se existir na memória de trabalho algum fato que justifique a escolha da super-espécie (A,B) (ou (C,D)), então essa será a super-espécie construída pelo SINCA. Porém, se nenhum fato for encontrado na memória de trabalho, então o módulo simbólico mostrará uma janela informando ao usuário as super-espécies possíveis (Fig. 4.12) com um botão de confirmação. Quando o usuário pressionar este botão o módulo simbólico mostrará uma janela para que o usuário faça a sua escolha e forneça os motivos que o levaram a fazer sua escolha (Fig. 4.13).

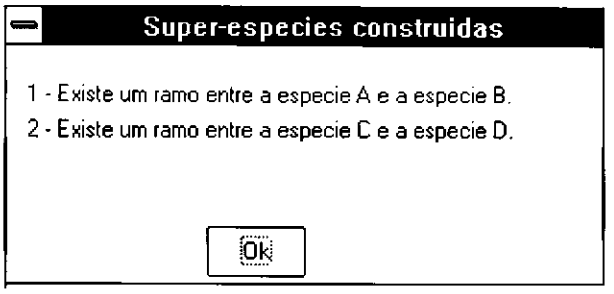


Fig. 4.12 - Tela que informa ao usuário as super-espécies construídas

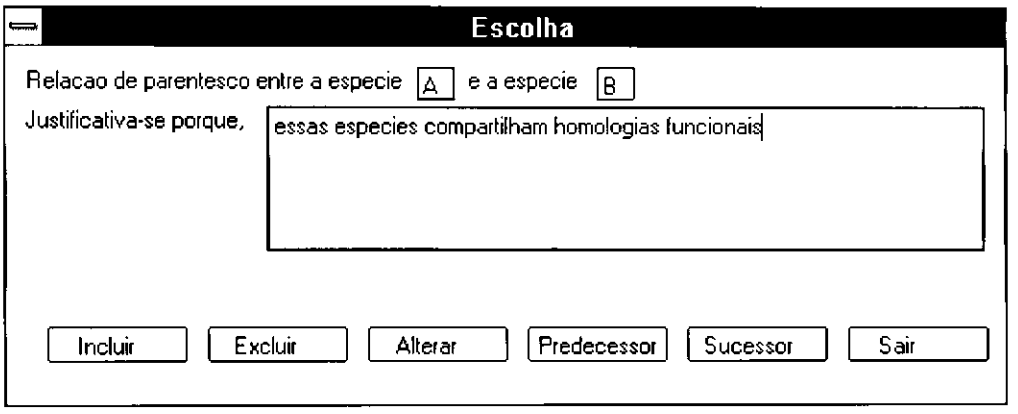


Fig. 4.13 - Tela para que o usuário forneça informações adicionais sobre as espécies investigadas

Em seguida, o módulo simbólico irá gerar a Tabela 4.3. O módulo simbólico então verificará se o número de espécies é maior ou igual a 5 e se existe conflito entre as informações armazenadas na matriz gerada. Como nenhuma dessas condições são verdadeiras, o módulo simbólico usará a regra **R4.2** de sua base de conhecimento sobre o conjunto de espécies $\{S_1,C,D\}$.

	S_1	C	D
S_1	0	4	6
C	4	0	2
D	6	2	0

Tabela 4.3 - Matriz gerada a partir da Tabela 4.2 com S_1 = (A,B)

A regra **R4.2** construirá o super-espécie S_2 = (C,D), gerará a matriz distância da Tabela 4.4 e acionará a regra **R4.1**.

	S_1	S_2
S_1	0	5
S_2	5	0

Tabela 4.4 - Matriz gerada a partir da Tabela 4.3 com S_2 = (C,D)

A regra **R4.1** construirá o super-espécie $S_3 = (S_1, S_2)$ e terminará o processo de construção dos ramos da árvore filogenética.

Depois, o módulo simbólico iniciará o processo de construção da árvore filogenética. A árvore filogenética construída pelo SINCA será então exibida na tela para que o usuário possa questionar e validar as super-espécies construídas (Fig. 4.14).

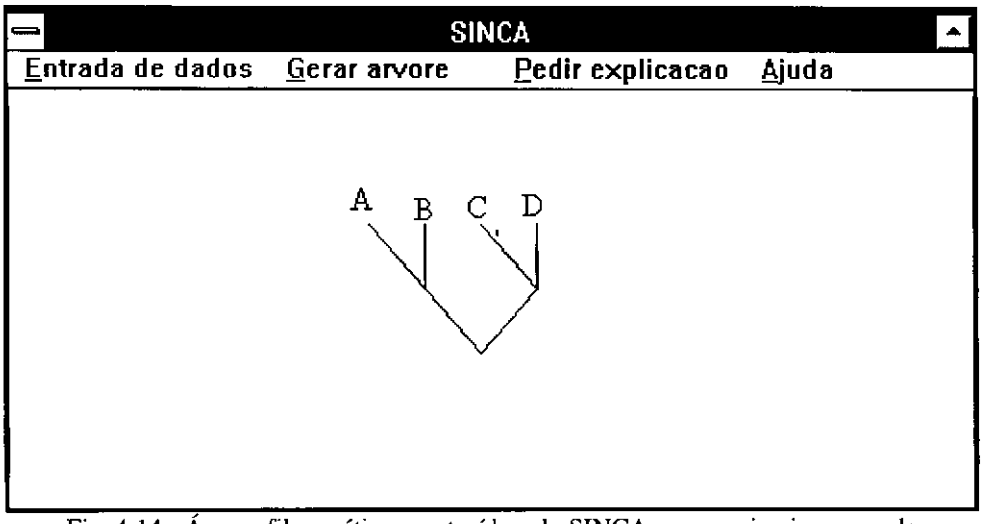


Fig. 4.14 - Árvore filogenética construída pelo SINCA para o primeiro exemplo

Supondo que o usuário acesse a opção "Pedir explicação" da Tela Principal do SINCA para saber porque a super-espécie (A,B) foi construída. O SINCA apresentará uma tela para que o usuário forneça o nome da super-espécie questionada (Fig. 4.15).

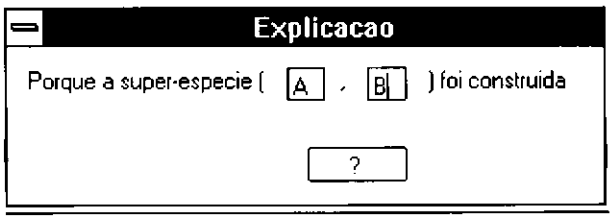


Fig. 4.15 - Tela de Explicação do SINCA

Após o usuário fornecer o nome da super-espécie questionada, o SINCA pesquisará em sua memória de trabalho um fato que justifique a construção dessa super-espécie. Ao termino dessa pesquisa o SINCA apresentará uma Tela de Justificativa (Fig. 4.16).

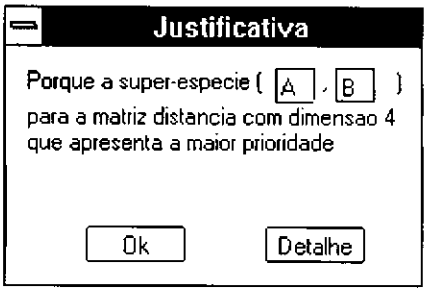


Fig. 4.16 - Tela de Justificativa do SINCA

A Tela de Justificativa contém uma explicação para a construção da super-espécie questionada, um botão de confirmação e um botão de detalhes. Se o usuário já estiver satisfeito com a justificativa apresentada pelo SINCA ele deve pressionar o botão de confirmação. Mas, se o usuário não estiver satisfeito com a resposta do SINCA ele deverá pressionar o botão de detalhes, então o SINCA apresentará a Tela de Detalhes. A Tela de Detalhes contem um texto explicativo e a matriz distância analisada com o(s) fato(s) considerados na construção da super-espécie questionada (Fig. 4.17) ou a regra usada na construção da super-espécie questionada.

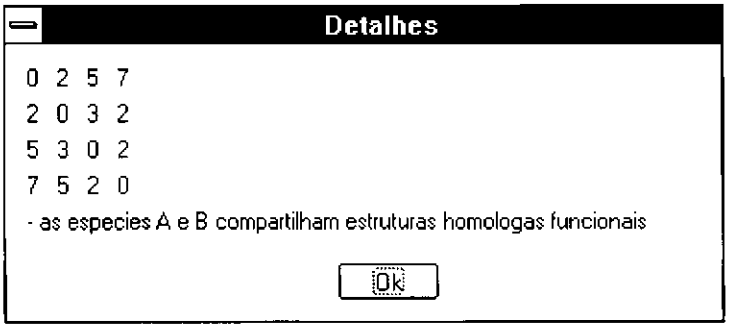


Fig. 4.17 - Tela de Detalhes do SINCA

A seguir é mostrado como o SINCA pode construir a árvore filogenética para um conjunto de espécies a partir da matriz distância. Usando a opção "Entrada de dados", o usurário fornecerá a matriz distância da Tabela 4.5 acessando a opção "Criar" do menu pull-down da opção "Matriz distância".

	A	B	C	D	E
A	0	4	1	4	1
B	1	0	5	2	3
C	1	5	0	5	2
D	4	2	5	0	3
E	1	3	2	3	0

Tabela 4.5 - Matriz distância do segundo exemplo do capítulo 4

A criação de um novo arquivo para conter a matriz distância fornecida pelo usuário exige que o usuário antes de fornecer a matriz informe ao SINCA o nome do arquivo e do diretório que conterá essa matriz (Fig. 4.18).

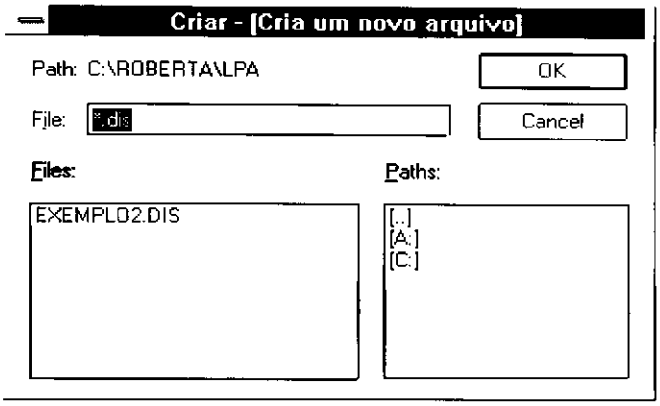


Fig. 4.18 - Tela de Entrada do Diretório da matriz distância

Se o nome do arquivo já existir, o SINCA mostrará um janela de alerta, senão o módulo simbólico do SINCA criará o arquivo neste diretório e chamará o WordPad para que o usuário entre com a matriz distância na forma de um conjunto de fatos como mostrado na Fig 4.19.

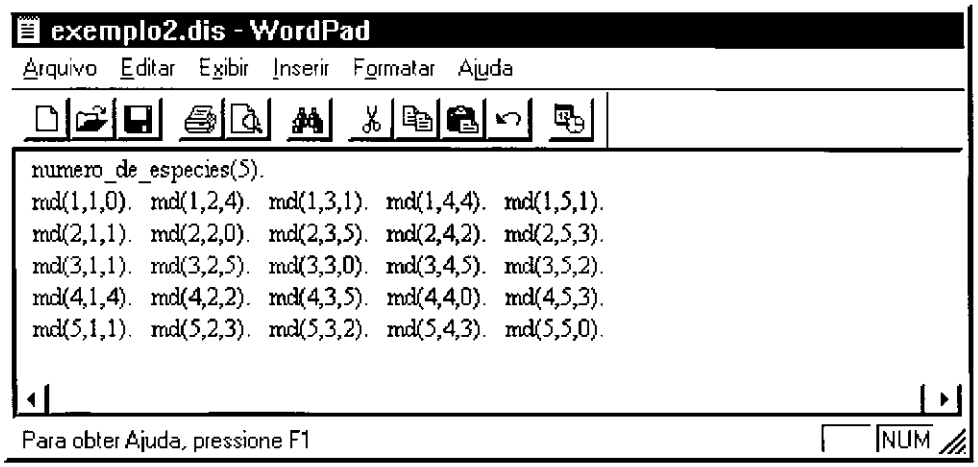


Fig. 4.19 - Tela de Entrada da Matriz Distância

Após, a entrada da matriz distância da Tabela 4.5, o usuário deverá acessar a opção "Gerar árvore" da Tela Principal e escolher a opção "Ver resultados" do menu pull-down da opção "Gerar árvore" (Fig. 4.11).

De posse da matriz distância da Tabela 4.5, inicialmente, o módulo simbólico verificará se o número de espécies desta matriz é maior ou igual a 5 e se existe conflito entre os dados armazenados na matriz fornecida, então o módulo simbólico gravará no arquivo saida.sim a

matriz distância fornecida na forma tabular e enviará um sinal de controle para o seu módulo conexionista. Quando o módulo conexionista receber o sinal de controle, ele inicia o processo de leitura da matriz distância gravada no arquivo saida.sim. Em seguida, o módulo conexionista analisará todas as linhas da matriz distância da Tabela 4.5.

A primeira linha a ser analisada será a linha 1 da matriz distância, o vetor de entrada correspondente a esta linha é (0.00, 0.25, 1.00, 0.22, 1.00). Para este vetor de entrada é criada uma rede neural de Hopfield formada por 5 neurônios. A seguir, inicia-se a competição dos neurônios da rede criada durante 300 iterações. Os resultados da competição, usando o vetor de entrada externa (0.00, 0.25, 1.00, 0.22, 1.00), são apresentados na Fig. 4.20 na forma de círculos e na forma de curvas.

Na Fig. 4.20 os círculos são numerados da esquerda para a direita e representam os valores das saídas dos neurônios da rede neural de Hopfield após o processo de competição. Os círculos preenchidos com a cor branca correspondem aos neurônios com valor de saída menor do que 0. Os círculos preenchidos com a cor preta correspondem aos neurônios com valor de saída maior do que 0. Os diâmetros dos círculos são diretamente proporcionais aos valores das saídas dos neurônios. Observe que os neurônios 3 e 5 foram os neurônios vencedores. As curvas da Fig. 4.20 são numeradas da esquerda para a direita de cima para baixo e representam a trajetória descrita pelos neurônios da rede neural de Hopfield criada durante o processo de competição. Observe que inicialmente todos os neurônios tem valor menor do que 0 devido a excitação (-0.5). O valor das saídas dos neurônios da rede serão alterados durante o processo de competição até a rede neural de Hopfield atingir um estado estável. Quando a rede neural de Hopfield atinge um estado estável a saída dos neurônios, V_t = (-0.91, -0.54, 0.99, -0.65, 0.99), não se altera mais.

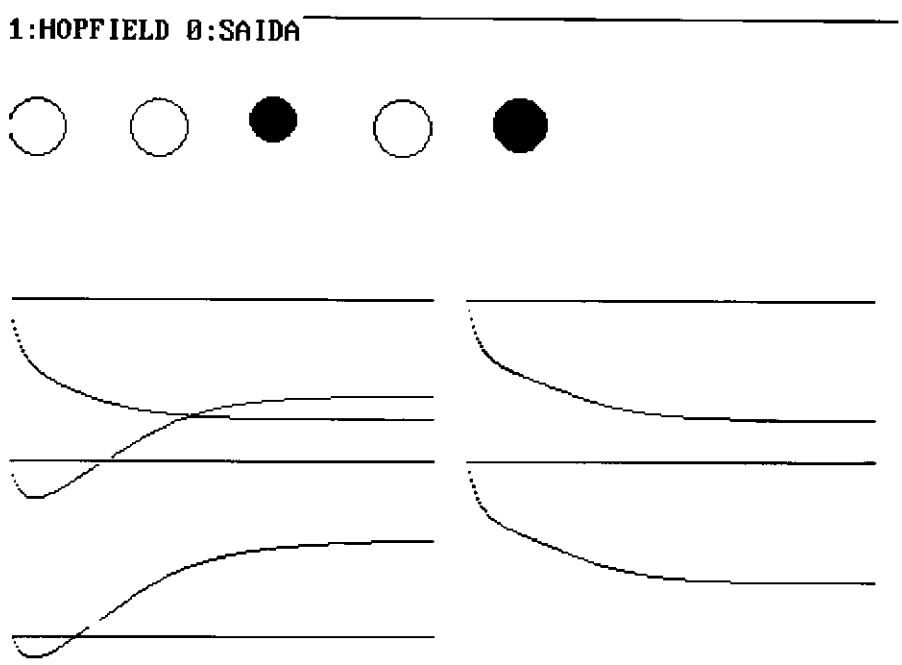


Fig. 4.20 - Competição dos neurônios da rede neural de Hopfield para primeira linha da Tabela 4.5

O vetor de saída V_1 indica que a espécie 1 pode formar uma super-espécie com a espécie 3 e com a espécie 5. Mas para se construir estas super-espécies, o módulo conexionista do SINCA precisa verificar se a espécie 3 confirma a construção da super-espécie (A,C) e se a espécie 5 confirma a construção da super-espécie (A,E).

A Tabela 4.6 descreve os próximos passos executados pelo módulo conexionista do SINCA. A primeira coluna dessa tabela apresenta o número da linha analisada; a segunda coluna apresenta o vetor de entrada externa fornecido a rede neural de Hopfield formada por 5 neurônios; a terceira coluna apresenta o vetor de saída da rede neural de Hopfield formada por 5 neurônios após a competição entre esses neurônios durante 300 iterações; a quarta e última coluna apresenta as hipóteses construídas.

Linha analisada	Vetor de entrada externa	Vetor de saída	Hipóteses construídas
3 (C)	(1.00, 0.20, 0.00, 0.20, 0.50)	(0.99, -0.74, -0.91, -0.46, -0.99)	(C, E)
5 (E)	(1.00, 0.33, 0.50, 0.33, 0.00)	(0.99, -0.44, 0.98, -0.65, -0.94)	(A, E) (C, E)
2 (D)	(1.00,0.00,0.20, 0.50, 0.33)	(0.08, -0.97, -0.93, 0.97, 0.96)	(A, B) (B, D) (B, E)
4 (E)	(0.25, 0.50, 0.20, 0.00, 0.33)	(0.25, 0.96, -0.96, -0.98, 0.94)	(A, D) (B, D) (E, D)

Tabela 4.6 - Resumo dos passos do ANCA para a análise da Tabela 4.5

Após a criação das super-espécies possíveis para as linhas da matriz distância da Tabela 4.5 o módulo conexionista do SINCA verificará a menor distância mútua apresentada por estas super-espécies. As super-espécies que apresentarem a menor distância serão guardados pelo módulo conexionista do SINCA como o conjunto de todas as super-espécies possíveis para a matriz distância da Tabela 4.5. A super-espécie (A,C) tem distância mútua 1, a super-espécie (A,E) tem distância mútua 1 e a super-espécie (C,E) tem distância mútua 2. O módulo conexionista do SINCA escolherá as super-espécies (A,C) e (A,E).

Em seguida, o módulo conexionista analisará as espécies da matriz gravada nesse arquivo e construirá o conjunto de super-espécies {(A,C), (A,E)}. Após a construção do conjunto de super-espécies o módulo conexionista gravará este conjunto no arquivo saida.rnh na forma de fatos e enviará um sinal de controle para informar ao módulo simbólico que já terminou a sua tarefa.

Ao receber o sinal de controle o módulo simbólico lerá o conjunto de todas as super-espécies construídas do arquivo saida.rnh para depois considerar esta informação na construção da árvore filogenética. Se a cardinalidade do conjunto de super-espécies for igual a 1, então o módulo simbólico criará a única super-espécie desse conjunto, senão o módulo simbólico pesquisará em sua memória de trabalho um fato que justifique a escolha de uma dessas super-espécies. Se existir na memória de trabalho algum fato que justifique a escolha da super-espécie (A,C) (ou (C,E)), então essa será a super-espécie construída pelo SINCA. Porém, se nenhum fato for encontrado na memória de trabalho, então o módulo simbólico mostrará uma janela informando ao usuário as super-espécies possíveis (Fig. 4.21) com um botão de confirmação. Quando o usuário pressiona este botão, o módulo simbólico mostrará uma janela para que o usuário faça a sua escolha e forneça os motivos que o levaram a fazer sua escolha (Fig. 4.22).

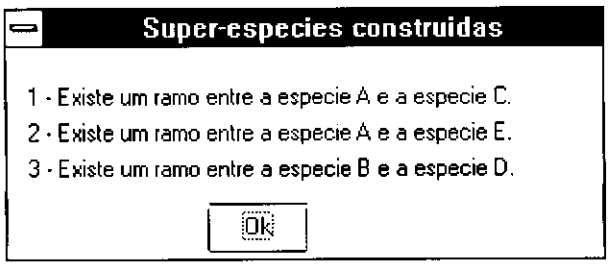


Fig. 4.21 - Tela que informa ao usuário as super-espécies construída para a matriz da Tabela 4.5

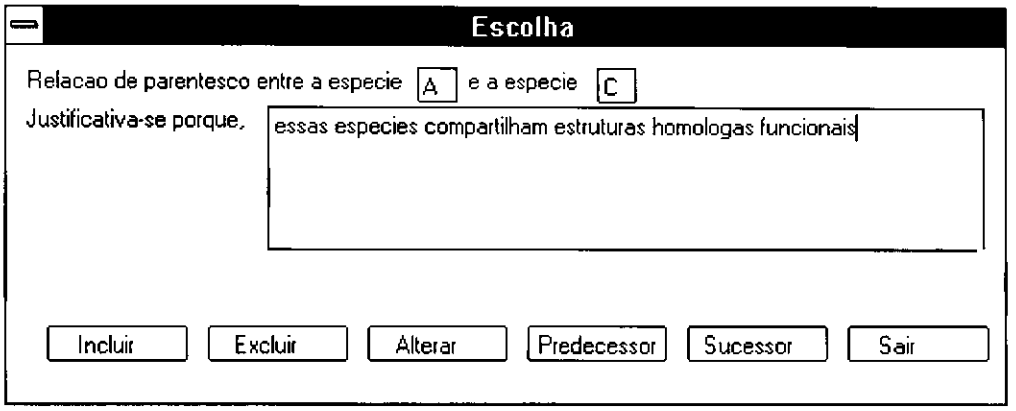


Fig. 4.22 - Tela para que o usuário forneça informações adicionais sobre as espécies da Tabela 4.5

Supondo que o usuário escolheu a super-espécie (A,C). O módulo simbólico irá gerar a Tabela 4.7.

	S_1	B	D	E
S_1	0	4.5	4.5	1.5
B	4.5	0	2	3
D	4.5	2	0	3
E	1.5	3	3	0

Tabela 4.7 - Matriz distância construída a partir da Tabela 4.5

O módulo simbólico então verificará se o número de espécies é maior ou igual a 5 e se existe conflito entre as informações armazenadas na matriz gerada. Como nenhuma dessas condições são verdadeiras o módulo simbólico irá usar a regra **R4.3** de sua base de conhecimento sobre o conjunto de espécies $\{S_1,B,D,E\}$. A regra **R4.3** construirá a super-espécie $S_2 = (S_1,E)$, acionará a regra **R4.2** sobre o conjunto de espécies $\{S_2,B,D\}$ e gerará a matriz distância da Tabela 4.8.

	S_2	B	D
S_2	0	3.75	3.75
B	3.75	0	2
D	3.75	2	0

Tabela 4.8 - Matriz distância construída a partir da Tabela 4.7

A regra **R4.2** construirá a super-espécie $S_3 = (B,D)$, acionará a regra **R4.1** sobre o conjunto de espécies $\{S_2, S_3\}$ e gerará a Tabela 4.9.

	S_2	S_3
S_2	0	3.75
S_3	3.75	0

Tabela 4.9 - Matriz distância construída a partir da Tabela 4.8

características apresentadas pelas espécies investigadas baseada nos fatos fornecidos pelo usuário que estão de acordo com a linha filosófica da taxinomia cladística, ou da sistemática filogenética, ou da classificação evolucionário, etc. [Christoffersen, 1995][Amorim, 1994].

O SINCA permite que o usuário (filogeneticista) tome conhecimento de todas as situações excepcionais ocorridas durante a construção de uma árvore filogenética que não foram previstas por ele ou que ainda não haviam ocorrido. A ocorrência de uma situação considerada excepcional pelo SINCA e absurda pelo usuário, servirá para o usuário identificar um erro cometido durante a fase de entrada de dados. Nesse caso o usuário poderá interromper a construção da árvore filogenética e reavaliar a matriz fornecida ao sistema. A ocorrência de uma situação considerada excepcional pelo SINCA e pelo usuário, servirá para o usuário reconhecer que o SINCA precisa de novos fatos para tratar a situação excepcional em questão.

A árvore filogenética construída pelo SINCA não será dependente da ordem na qual as espécies estão dispostas na matriz característica. Isso implica no fato do SINCA construir sempre a mesma árvore filogenética para um dado conjunto de espécies e um dado conjunto de características independente da ordem na qual as espécies estão dispostas na matriz característica analisada. Esse fato torna o SINCA mais confiável do que os sistemas que usam abordagens baseadas no algoritmo das médias e no algoritmo de Wagner. Isto porque a árvore filogenética construída por esses algoritmos é dependente da ordem na qual as espécies encontram-se dispostas na matriz característica analisada.

O SINCA é o primeiro sistema híbrido desenvolvido para construir árvores filogenéticas. A utilização da rede neural de Hopfield para controlar a explosão combinatorial e de um sistema especialista para construir árvores filogenéticas segundo uma determinada corrente filosófica também é uma iniciativa inédita.

5.3 Sugestões de Trabalhos Futuros

Como trabalhos futuros sugerimos:

- aplicar o SINCA a outros tipos de problemas NP-completos, que tenham uma função de custo a otimizar e que trabalhem com objetos facilmente representáveis em uma matriz que relacione estes objetos aos seus atributos;
- estender o SINCA para construir árvores filogenéticas a partir de dados de Genética, de Citologia, etc., das seguintes maneiras:
  i. construir uma árvore filogenética a partir de dados de Genética, uma árvore filogenética a partir de dados de Citologia, e assim por diante. Após o sistema ter construído uma árvore filogenética para todas as diferentes natureza de dados consideradas, o sistema construirá uma nova árvore filogenética para o conjunto de espécies investigadas que apresente somente as semelhanças morfológicas entre as árvores filogenéticas de todas as diferentes natureza consideradas;
  ii. combinar todas as matrizes distâncias construídas para cada uma das natureza consideradas em uma única matriz A e depois executar a construção da árvore filogenética a partir da análise dos dados armazenados em A.
- construir um módulo que forneça ao usuário o valor da excitação que deverá ser atribuída como entrada externa aos neurônios da rede neural de Hopfield para que o resultado de sua análise seja correto, este módulo deverá variar o valor da excitação da rede neural de Hopfield formada por n neurônios, com n igual ao número de espécies. Considerando que o vetor de entrada externa dos neurônios da rede assume os valores (0.1, 0.2, ..., 0.n) e (0.n, 0.n-1, ...,0.1). Se após uma competição de 300 iterações o neurônio vencedor da rede com um valor de excitação igual a δ for o neurônio 1 para o vetor de entrada externa (0.n, 0.n-1, ...,0.1) e for o neurônio n para o vetor de entrada externa (0.1, 0.2, ..., 0.n).
- construir um módulo que permita o usuário entrar com a matriz característica na forma tabular. Este módulo deve também ser capaz de gerar os fatos que descrevem a matriz características fornecida pelo usuário para que o SINCA possa manipular as informações armazenadas nessa matriz.

Apêndice A - Exemplo da combinação de dados de diferentes naturezas em uma matriz distância

Considere que a distância entre os dados de natureza morfológica é calculada pela fórmula (A.1) e que a distância entre os dados de natureza genética é calculada pela fórmula (A.2). A fórmula (A.2) só pode ser aplicada sobre duas seqüências genéticas alinhadas. Diz-se que duas ou mais seqüências genéticas estão alinhadas se são inseridos buracos (espaços em branco) nestas seqüências em pontos chaves, de modo que todas fiquem com o mesmo comprimento e apresentem o maior número de bases nucleotídicas (A - adenina, T - timina, C - citosina e G - guanina) emparelhadas [Meidanis, 1994][Meidanis, 1995][Waterman, 1991]. Isso porque, as alterações ocorridas nas características morfológicas de uma espécie são expressas no seu código genético através da inserção, exclusão ou troca de bases nucleotídicas.

$$\mathrm{Dist}(i,j,k)=\begin{cases}\left|\mathrm{Caract}_j[i]-\mathrm{Caract}_k[i]\right|+\mathrm{Dist}(i-1) & ,\mathrm{Caract}_j[i]\neq\mathrm{Caract}_k[i]\\ 0+\mathrm{Dist}(i-1) & ,\mathrm{Caract}_j[i]=\mathrm{Caract}_k[i]\end{cases} \tag{A.1}$$

onde Caract_i é o vetor característica ($c_{i,1}$, ..., $c_{i,n}$) da espécie da *i* linha da matriz polarizada *C*, $c_{i,j}$ é o elemento da *i* linha e *j* coluna de *C* e *k* é um valor inteiro.

$$\mathrm{Dist}(i,j,k)=\begin{cases}1+\mathrm{Dist}(i-1) & ,\mathrm{base}(\mathrm{DNA}_j[i])\wedge\mathrm{base}(\mathrm{DNA}_k[i])\wedge(\mathrm{DNA}_j[i]=\mathrm{DNA}_k[i])\\ -1+\mathrm{Dist}(i-1) & ,\mathrm{buraco}(\mathrm{DNA}_j[i])\wedge\mathrm{base}(\mathrm{DNA}_k[i])\\ -1+\mathrm{Dist}(i-1) & ,\mathrm{buraco}(\mathrm{DNA}_k[i])\wedge\mathrm{base}(\mathrm{DNA}_j[i])\\ 0+\mathrm{Dist}(i-1) & ,\mathrm{base}(\mathrm{DNA}_j[i])\wedge\mathrm{base}(\mathrm{DNA}_k[i])\wedge(\mathrm{DNA}_j[i]\neq\mathrm{DNA}_k[i])\end{cases} \tag{A.2}$$

onde a saída da função *buraco* é VERDADE se o caracter de entrada não for uma base nucleotídica e FALSO caso contrário (A.3), enquanto que a saída da função *base* é VERDADE se

o caracter de entrada for uma base nucleotídica e FALSO caso contrário (A.4), e $DNA_j[i]$ é o caracter i da seqüência genética da espécie j.

$$\text{buraco}(x) = \begin{cases} \mathbf{V} & ,(x \neq A) \wedge (x \neq T) \wedge (x \neq C) \wedge (x \neq G) \\ \mathbf{F} & ,(x = A) \vee (x = T) \vee (x = C) \vee (x = G) \end{cases} \quad \text{(A.3)}$$

$$\text{base}(x) = \begin{cases} V & ,(x = A) \vee (x = T) \vee (x = C) \vee (x = G) \\ F & ,(x \neq A) \wedge (x \neq T) \wedge (x \neq C) \wedge (x \neq G) \end{cases} \quad \text{(A.4)}$$

A combinação dos dados de natureza morfológica e genética armazenados em uma matriz distância será obtida pela aplicação da fórmula (A.5) sobre essas matrizes.

$$\text{Matriz\_combinada} = \sum_{j=1}^{n} \sum_{i=1}^{n} \alpha_1 m_{ij} + \alpha_1 g_{ij}$$

(A.5)

onde m_{ij} representa o elemento da linha i e coluna j da matriz distância de dados morfológicos, g_{ij} representa o elemento da linha i e coluna j da matriz distância de dados genéticos, n é o número de espécies investigadas, e α_i é a prioridade atribuída aos dados de natureza morfológica (i = 1) e genética (i = 2).

Por exemplo, considere a existência de três espécies esp_1, esp_2 e esp_3 que apresentam as séries de transformação expressas na Tabela A.1 e as seguinte seqüências de DNA AAACCCTG, AAGTC e TTACACT para a esp_1, esp_2 e esp_3, respectivamente. Utilizando a fórmula (A.1) sobre os dados de natureza morfológica obtém-se a matriz distância da Tabela A.2 e utilizando a fórmula (A.2) sobre as seqüências de DNA alinhadas entre duas espécie:

- esp_1 = AAACCCTG e

  esp_2 =AABGBBTC,

- esp_1 = AAACCCTG e

  esp_3 =TTACACTB,

- esp_2 = ABAGTCB e

  esp_3 =TTACACT

onde a letra B simboliza um buraco, obtém-se a matriz distância da Tabela A.3.

	1	2	3	4
A	1	1	0	0
B	1	0	1	1
C	0	1	2	1

Tabela A.1 - Matriz característica utilizada para exemplificar o cálculo da distância morfológica

	A	B	C
A	0	3	4
B	3	0	3
C	4	3	0

Tabela A.2 - Matriz distância da análise morfológica das espécies da Tabela A.1

	A	B	C
A	0	0	3
B	0	0	0
C	3	0	0

Tabela A.3 - Matriz distância da análise genética das espécies estudadas

Aplicando a fórmula (A.5) sobre a matriz distância dos dados morfológicos (Tabela A.2) e genéticos (Tabela A.3), obtém-se a matriz distância da Tabela A.4 que apresenta uma análise mais completa das relações de parentescos entre as espécies estudadas, considerando que a prioridade dos dados de natureza morfológica e genética é 1.

	A	B	C
A	0	3	7
B	3	0	3
C	7	3	0

Tabela A.4 - Matriz distância que combina os resultados da Tabela A.2 e A.3.

Apêndice B - Exemplo da combinação de dados de diferentes naturezas em uma matriz similaridade

Considere que a similaridade entre os dados de natureza morfológica é calculada pela fórmula (B.1) e que a similaridade entre os dados de natureza genética é calculada pela fórmula (B.2). A fórmula (B.2) só pode ser aplicada sobre duas seqüências genéticas alinhadas.

$$\mathrm{Sim}(i,j,k) = \begin{cases} 1 + \mathrm{Sim}(i-1) & ,\mathbf{Caract}_j[i] = \mathbf{Caract}_k[i] \\ 0 + \mathrm{Sim}(i-1) & ,\mathbf{Caract}_j[i] \neq \mathbf{Caract}_k[i] \end{cases} \quad \text{(B.1)}$$

onde $\mathbf{Caract}_i$ é o vetor característica ($c_{i,1}$, ..., $c_{i,n}$) da espécie da *i* linha da matriz polarizada *C*, $c_{i,j}$ é o elemento da *i* linha e *j* coluna de *C* e *k* é um valor inteiro.

$$\mathrm{Sim}(i,j,k) = \begin{cases} 1 + \mathrm{Sim}(i-1) & ,\mathrm{base}(\mathrm{DNA}_j[i]) \wedge \mathrm{base}(\mathrm{DNA}_k[i]) \wedge (\mathrm{DNA}_j[i] = \mathrm{DNA}_k[i]) \\ -1 + \mathrm{Sim}(i-1) & ,\mathrm{caso\_contrario} \end{cases} \quad \text{(B.2)}$$

onde a saída da função *base* é VERDADE se o caracter de entrada for uma base nucleotídica e FALSO caso contrário, e $DNA_j[i]$ é o caracter *i* da seqüência genética da espécie *j*.

A combinação dos dados de natureza morfológica e genética armazenados em uma matriz similaridade será obtida pela aplicação da fórmula (B.3) sobre essas matrizes.

$$\mathrm{Matriz\_combinada} = \sum_{j=1}^{n}\sum_{i=1}^{n} \alpha_1 m_{ij} + \alpha_2 g_{ij} \quad \text{(B.3)}$$

onde m_{ij} representa o elemento da linha i e coluna j da matriz similaridade de dados morfológicos, g_{ij} representa o elemento da linha i e coluna j da matriz similaridade de dados genéticos, n é o número de espécies investigadas, e α_i prioridade atribuída aos dados de natureza morfológica (i = 1) e genética (i = 2).

Por exemplo, considere a existência de três espécies esp_1, esp_2 e esp_3 que apresentam as séries de transformação expressas na Tabela B.1 e as seguinte seqüências de DNA AAACCCTG, AAGTC e TTACACT para o esp_1, esp_2 e esp_3, respectivamente. Utilizando a fórmula (B.1) sobre os dados de natureza morfológica obtém-se a matriz similaridade da Tabela B.2 e utilizando a fórmula (B.2) sobre as seqüências de DNA alinhadas entre duas espécies:

- esp_1 = AAACCCTG e

  esp_2 =AABGBBTC,

- esp_1 = AAACCCTG e

  esp_3 =TTACACTB,

- esp_2 = ABAGTCB e

  esp_3 =TTACACT

onde a letra B simboliza um buraco, obtém-se a matriz distância da Tabela B.3.

	1	2	3	4
A	1	1	0	0
B	1	0	1	1
C	0	1	2	1

Tabela B.1 - Matriz característica utilizada para exemplificar o cálculo da similaridade morfológica

	A	B	C
A	0	1	1
B	1	0	3
C	1	1	0

Tabela B.2 - Matriz similaridade da análise morfológica das espécies da Tabela B.1

	A	B	C
A	0	3	4
B	3	0	2
C	4	2	0

Tabela B.3 - Matriz similaridade da análise genética das espécies estudadas

Aplicando-se a fórmula (B.3) sobre a matriz similaridade dos dados morfológicos (Tabela B.2) e genéticos (Tabela B.3), obtém-se a matriz similaridade da Tabela B.4 que apresenta uma

análise mais completa das relações de parentescos entre as espécies estudadas, considerando que a prioridade dos dados de natureza morfológica e genética é 1.

	A	B	C
A	0	4	5
B	4	0	3
C	5	3	0

Tabela B.4 - Matriz similaridade que combina os resultados da Tabela B.2 e B.3.

Abstract

This work presents a neural symbolic system for the construction of phylogenetic trees, called SINCA. In this system the symbolic and connectionist technical work in a cooperative way. SINCA's connectionist module use a Hopfield neural network in order to run to the smaller distance between the branches of the phylogenetic tree, controlling the combinatorial explosion created by the number of possible phylogenetic trees for the set of investigated organisms. SINCA's symbolic module uses an expert system for the construction of phylogenetic trees based on the users knowledge, about the rules of its knowledge base and of the knowledge created by the connectionist module. It presents a phylogenetic trees study, the main algorithms for their construction, a Hopfield neural network's study, its stability and weights, the neural algorithm implementation for phylogenetic tree construction (ANCA) and an example for phylogenetic tree construction with the ANCA. Finally, it presents SINCA's implementation, some examples of phylogenetic tree construction with SINCA and it give suggestions on future works.

Referências Bibliograficas

[Abe,1991] Abe J. M., Papavero N., "Teoria Intuitiva dos Conjuntos" . São Paulo: Editora Makron Books, 1991.

[Aiyer, 1990] Aiyer A. V. B., Niranjan M., and Fallside F., "A Theorical Investigation into the Performance od the Hopfield Model". IEEE Transactions on Neural Networks, vol. 1, nº 2, pp. 204-215, June 1990.

[Ali, 1993] Ali M. K. M., and Kamoun F., "Neural Networks for Shotrest Path Computation and Routing in Computer Networks". IEEE Transactions on Neural Networks, vol. 4, nº 6, pp. 943-955, November 1993.

[Andrade, 1997] Andrade P. S., "Sistemas Híbridos Neurosimbólicos, Estudo e Implementação", Dissertação de Mestrado, Universidade Federal da Paraíba, Campina Grande, 1997.

[Ansari, 1995] Ansari N., Hou E. S. H and Yu Y., "A New Method to Otimize the Satellite Broadcasting Schedules Using the Mean Field Annealing of a Hopfield Neural Network". IEEE Transactions on Neural Networks, vol. 6, nº 2, pp. 470-482, march 1995.

[Amorim, 1994] Amorim D.S., "Elementos Básicos de Sistemática Filogenética". São Paulo. Sociedade Brasileira de Entomologia, 1994.

[Bachmann, 1995] Bachmann K., "Progress and pitfalls in systematics: cladistics, DNA and morphology". Acta Bot. Neerl. 44(4), December 1995, pp.403-419.

[Bernardi, 1981] Bernardi N., "Resumo das aulas sobre: Sistemática Filogenética". I Curso Especial de Sistemática Zoológica do Departamento de Ciências Biológicas, São Carlos - SP.

[Campello, 1994] Campelo R. E, e Maculan N., "Algoritmos e Heurísticas: desenvolvimento e avaliação de performance". Rio de Janeiro: Universidade Fluminense, 1994.

[Christoffersen, 1995] Christoffersen M. L., "Cladistic Taxonomy, Phylogenetic Systematics, and Evolutionary Ranking". Syst. Biol. 44(3):440-454, 1995.

[Clement, 1986] Clement C. R., "Descriptores minimos para el pejibaye (Bactris gasipaes H.B.K.) y sus implicaciones filogeneticas". Tese de Mestrado. Universidade de Costa Rica, 1986.

[Giarratano, 1989] Giarratano J. and Riley G., "Expert Systems". Estados Unidos. PWS-KENT, 1989.

[Haykin, 1994] Haykin S., "Neural Networks: A Comprehensive Foundation". New York: Macmillan College Publishing Company, 1994.

[Hopfield, 1982] Hopfield J. J., "Neural networks and physical systems with emergent collective computational abilities". **In**: IEEE Press, pp. 25-29 ,1994.

[Hopfield, 1984] Hopfield J. J., "Neurons with graded response have collective computational properties like those of two-state neurons". Proc. Nat. Acad. Sci. U.S., vol.81, pp.3088-3092, May 1984.

[Hopfield, 1986] Hopfield J. J., and Tank D. W., "Simple "Neural" Optimization Networks: An A/D Converter, Signal Decision Circuit, and a Linear Programming Circuit". **In**: IEEE Press, pp. 444-452, 1992.

[Kopchenova, 1975] Kopchenova N. V., and Maron I. A., "Computational Mathematics: worked examples and problems with elements of theory" , Mir Publishers Moscow, 1975.

[Kovács, 1996] Kovács Z. L., "Redes Neurais Artificiais: Fundamentos e Aplicações". Livro 1ª edição, Edição Acadêmica, São Paulo, 1996.

[Lawrence, 1951] Lawrence G. H. M., "Taxonomy of Vascular Plants". Livro, vol. 1, Copyright the Macmillan company, 1951.

[McClelland, 1989] McClelland, J.L. & Rumelhart, D.E., "Explorations in Parallel Distributed Processing", The MIT Press, USA, Fourth Printing, 1989.

[Meidanis, 1994] Meidanis J., Setubal J., "Uma introdução à biologia computacional". IX Escola de Computação, Recife, 1994.

[Meidanis, 1995] Meidanis J. e Setubal J., "Multipe Alignment of Biological Sequences with Gap Flexibility". Second South American Workshop on String Processing, Chile, pp. 138-153, April 1995.

[Rizzini, 1979] Rizzini C. T., " Trabalho de Fitogeografia do Brasil". Livro 2º volume, São Paulo: Hueitec-Edu, 1979.

[Tagliarini, 1991] Taglirini G. A., Christ J. F., and Page E. W., "Optimization Using Neural Networks". IEEE Transactions on Computers, vol. 40, nº 12, pp. 1347-1358, December 1991.

[Waterman, 1991] Waterman M. S., "Mathematical Methods for DNA Sequences". Informatique et Genome, pp. 1-242, Avril 1991.

[Wiley, 1990] Wiley E.O., Siegel-Causey D., Brooks D.R., "Funk V.A.,The compleast cladist: A primer of philogenetic procedures". Publicação especial nº 19 da University of Kansas Museum of Natural Históry, 1991.

Bibliografia

[Belew, 1988] Belew R. K. and Forrest S., "Learning and Programming in Classifier Systems". Machine Learning, vol 3, nº 2/3, pp. 193-223, October 1988.

[Booker, 1988] Booker L. B., "Classifier Systems the Learn Internal World Models". Machine Learning, vol 3, nº 2/3, pp. 162-192, October 1988.

[Chakrabarti, 1995] Chakrabarti C., Bindal N. and Theaghajan K., "Robust Radar Target Classifier Using Artificial Neural Networks". IEEE Transations on Neural Networks, vol. 6, nº 3, pp. 760-767, May 1995.

[Christoffersen, 1994a] Christoffersen M. L., "An Overview of Cladistic Applications". Rev. Nordestina Biol., 9(2):133-141.

[Christoffersen, 1994b] Christoffersen M. L., "A Phylogenetic Framework of the Enterocoela (Metameria: Coelomata)". Rev. Nordestina Biol., 9(2):173-208.

[Farris, 1995] Farris J. S., "Conjectures and Refutations". Cladistics (1995) 11:105-118, 1995.

[Kumar, 1995] Kumar S., Tamura K., Nei M., "MEGA: Molecular Evolutionary Genetics Analysis, Version 1.02". Syst. Biol. 44(4):576-577, 1995.

[Hennig86, 1996] Hennig86 Help. (http://www.vims.edu/~mês/mês/henhelp.html)

[Jagota, 1995] Jagota A., "Approximating Maximum Clique with a Hopfield Network". IEEE Transactions on Neural Networks, vol. 6, nº 3, pp. 724-735, May 1995.

[Lipscomb, 1994] Lipscomb D., "Cladistic Analysis usando Hennig86". Manual.

[Maeda, 1995] Maeda Y., Yotsumoto Y. and Kanata Y., "Unsupervised Learning of Neural Networks for Separation of Unknown Data". IEEE, 9/95.

[Petridis, 1995] Petridis V. and Paraschdis K., "On the Properties of the Feedforward Method a Simple Training Law for Onclip Learning". IEEE Transactions on Neural Networks, vol 6, nº 6, pp. 1536-1540, November 1995.

[Phylip, 1995] Complete documentation. (http://utmmg.med.uth.tme.edu/mmg/genetics/phylip_info/main.html)

[Phylogenetic, 1996] Phylogenetic Analysis Computer programs. (http://phylogeny.arizona.edu/tree/programs/programs.html)

[Rannala, 1995] Rannala B., "Polymorphic Characters and Phylogenetic Analysis: A Statistical Perspective". Syst. Biol. 44(3):421-429, 1995.

[Rich, 1993] Rich E., Knight Kevin., "Inteligência Artificial". São Paulo: Makron Books, 1993.

[Robertson, 1988] Robertson G. G. and Riolo R. L., "A Tale of Two Classifier Systems". Machine Learning, vol 3, nº 2/3, pp. 139-159, October 1988.

[Ronquist, 1996] Ronquist F., "Matrix Representation of Trees, Redundancy, and