

PAULO HENRIQUE SIQUEIRA

**UMA NOVA ABORDAGEM NA RESOLUÇÃO DO
PROBLEMA DO CAIXEIRO VIAJANTE**

Tese apresentada como requisito parcial à obtenção do grau de Doutor em Ciências. Programa de Pós-Graduação em Métodos Numéricos em Engenharia – Programação Matemática, Setores de Tecnologia e de Ciências Exatas, Universidade Federal do Paraná.

Orientador: Prof. Dr. Sergio Scheer

Co-orientadora: Prof.^a Dr.^a Maria Teresinha
Arns Steiner

**CURITIBA
2005**

Siqueira, Paulo Henrique

Uma nova abordagem na resolução do problema do Caixeiro Viajante / Paulo Henrique Siqueira. — Curitiba, 2005.
xiii, 102 f.: il.; grafs.; tab.

Orientador: Sergio Scheer

Co-orientador: Maria Teresinha Arns Steiner

Tese (doutorado) – Universidade Federal do Paraná. Setor de Ciências Exatas.

1. Redes neurais (Computação). 2. Problema do caixeiro viajante.
3. Otimização matemática. I. Scheer, Sergio. II. Steiner, Maria Teresinha Arns. III. Título.

CDD 20 519.3

TERMO DE APROVAÇÃO

PAULO HENRIQUE SIQUEIRA

UMA NOVA ABORDAGEM NA RESOLUÇÃO DO PROBLEMA DO CAIXEIRO VIAJANTE

Tese aprovada como requisito parcial para obtenção do grau de Doutor em Ciências, no Programa de Pós-Graduação em Métodos Numéricos em Engenharia – Programação Matemática da Universidade Federal do Paraná, pela seguinte banca examinadora:

Orientador:

Prof. Dr. Sergio Scheer
Departamento de Construção Civil, UFPR

Prof.^a Dr.^a Maria Teresinha Arns Steiner
Departamento de Matemática, UFPR

Prof. Dr. Júlio Cesar Nievola
Programa de Pós-Graduação em Informática Aplicada, PUC-PR

Prof. Dr. Nei Yoshihiro Soma
Divisão da Ciência da Computação, ITA-SP

Prof. Dr. Flavio Bortolozzi
Programa de Pós-Graduação em Informática Aplicada, PUC-PR

Curitiba, 27 de setembro de 2005.

Dedico este trabalho aos meus familiares
e aos meus amigos

AGRADECIMENTOS

À minha família, em especial à minha mãe Claudete Maria Rossa Siqueira, pelo apoio, incentivo e paciência durante a realização deste curso.

Ao professor Sérgio Scheer, pela orientação para a realização deste trabalho, pela confiança depositada em mim na escolha do tema, e principalmente pelo companheirismo e encorajamento passados para mim em todas as fases deste curso.

À professora Maria Teresinha Arns Steiner, pela co-orientação deste trabalho, pelas valiosas sugestões, e pelo companheirismo e acompanhamento constante na obtenção dos resultados deste trabalho.

Às minhas colegas e grandes amigas Luzia Vidal de Souza, Sonia Isoldi Marty Gama Muller, Maria Eugênia de Carvalho e Silva Sampaio e Ângela Olandoski Barboza, pelo apoio e companheirismo em nossos grupos de seminários e de estudos para o primeiro exame de qualificação.

Aos professores: Anselmo Chaves Neto, Sérgio Scheer, Maria Teresinha Arns Steiner, Celso Carnieri e Mildred Ballin Hecke, pelos ensinamentos transmitidos.

Ao coordenador do curso, professor Dr. Anselmo Chaves Neto, pela motivação para a realização deste trabalho.

Aos funcionários do CESEC, pela disposição em sempre ajudar, quando necessário.

À secretária do CESEC, Maristela Bandil, pela motivação e grande eficiência nos serviços prestados durante o curso.

Ao criador de uma das Redes Neurais Recorrentes utilizadas neste trabalho para resolver o problema da Designação, professor Jun Wang, que confirmou que as técnicas de

ajuste apresentadas neste trabalho são eficientes para corrigir problemas de convergência de tal Rede Neural, e manteve-se sempre interessado no andamento deste trabalho.

Aos meus colegas do Departamento de Desenho, da UFPR, pela licença concedida e pelo incentivo para a realização deste trabalho.

À professora Simone da Silva Soria Medina, chefe do departamento de Desenho, pelo apoio e pela concessão de um espaço para os estudos deste trabalho.

A todos, que de alguma forma contribuíram para a realização deste trabalho.

SUMÁRIO

1 INTRODUÇÃO.....	15
1.1 DESCRIÇÕES DOS PROBLEMAS	15
1.2 OBJETIVOS DO TRABALHO.....	16
1.3 IMPORTÂNCIA DO TRABALHO	17
1.4 LIMITAÇÕES DO TRABALHO	19
1.5 INOVAÇÕES PROPOSTAS NESTE TRABALHO.....	20
1.6 ESTRUTURA DO TRABALHO	21
2 REVISÃO DA LITERATURA.....	23
2.1 INTRODUÇÃO	23
2.2 MÉTODOS PARA A RESOLUÇÃO DO PROBLEMA DA DESIGNAÇÃO	23
2.3 APLICAÇÕES DAS REDES NEURAS ARTIFICIAIS.....	25
2.3.1 APLICAÇÕES DOS MAPAS AUTO-ORGANIZÁVEIS	25
2.3.2 O USO DE REDES NEURAS PARA RESOLVER O PROBLEMA DA DESIGNAÇÃO.....	30
2.4 MÉTODOS PARA A RESOLUÇÃO DO PROBLEMA DO CAIXEIRO VIAJANTE	31
2.5 AS TÉCNICAS UTILIZADAS NESTE TRABALHO	38
3 RESOLUÇÃO DO PROBLEMA DA DESIGNAÇÃO LINEAR COM A UTILIZAÇÃO DE REDES NEURAS RECORRENTES.....	39
3.1 DESCRIÇÃO DO PROBLEMA	39
3.1.1 FORMULAÇÃO DO PROBLEMA DA DESIGNAÇÃO	41
3.2 PROPOSTA DE CONSTRUÇÃO DA MATRIZ DE CUSTOS PARA O PROBLEMA DA DESIGNAÇÃO.....	42
3.2.1 O MAPA AUTO-ORGANIZÁVEL DE KOHONEN.....	42
3.2.2 CÁLCULO DE ERROS EM MAPAS AUTO-ORGANIZÁVEIS DE KOHONEN.....	46
3.2.3 UTILIZAÇÃO DA REDE DE KOHONEN PARA CONSTRUÇÃO DA MATRIZ DE CUSTOS DO PROBLEMA DA DESIGNAÇÃO	47
3.2.4 REPRESENTAÇÃO GRÁFICA DE MAPAS AUTO-ORGANIZÁVEIS DE KOHONEN.....	49
3.3 PROPOSTA DE RESOLUÇÃO DO PROBLEMA DA DESIGNAÇÃO UTILIZANDO REDES NEURAS RECORRENTES.....	54
3.3.1 A REDE NEURAL RECORRENTE DE WANG PARA O PROBLEMA DA DESIGNAÇÃO.....	54
3.3.2 MÚLTIPLAS SOLUÇÕES ÓTIMAS GLOBAIS E SOLUÇÕES ÓTIMAS LOCAIS PRÓXIMAS DA SOLUÇÃO ÓTIMA GLOBAL	56
3.3.3 USO DO PRINCÍPIO <i>WINNER TAKES ALL</i> APLICADO À REDE NEURAL RECORRENTE DE WANG.....	59
3.3.4 ALGUMAS METODOLOGIAS PARA AJUSTAR OS PARÂMETROS DA REDE NEURAL DE WANG	63

4 ESTUDO DE CASO	67
4.1 DESCRIÇÃO DO PROBLEMA	67
4.2 APLICAÇÃO DA REDE DE KOHONEN PARA CRIAR A MATRIZ DE CUSTOS DO PROBLEMA	69
4.2.1 DADOS DE ENTRADA	69
5 ANÁLISES DE RESULTADOS DA METODOLOGIA PROPOSTA PARA O PROBLEMA DA DESIGNAÇÃO LINEAR	73
5.1 INTRODUÇÃO	73
5.2 RESULTADOS DA APLICAÇÃO DA TÉCNICA <i>WINNER TAKES ALL</i> EM MATRIZES COM CUSTOS ALEATÓRIOS PARA O PROBLEMA DA DESIGNAÇÃO.....	73
5.3 RESULTADOS OBTIDOS COM A APLICAÇÃO DA REDE DE KOHONEN AO PROBLEMA DE ALOCAÇÃO DE SALAS DE AULA DA UFPR	76
5.4 APLICAÇÃO DA REDE NEURAL DE WANG COM O PRINCÍPIO <i>WINNER TAKES ALL</i> PARA O PROBLEMA DE ALOCAÇÃO DE SALAS DA UFPR	80
6 RESOLUÇÃO DO PROBLEMA DO CAIXEIRO VIAJANTE COM A APLICAÇÃO DA REDE NEURAL DE WANG COM O PRINCÍPIO <i>WINNER TAKES ALL</i>.....	82
6.1 DESCRIÇÃO DO PROBLEMA DO CAIXEIRO VIAJANTE	82
6.1.1 FORMULAÇÃO DO PROBLEMA DO CAIXEIRO VIAJANTE.....	83
6.2 SOLUÇÃO DO PROBLEMA DO CAIXEIRO VIAJANTE UTILIZANDO A REDE NEURAL DE WANG COM O PRINCÍPIO <i>WINNER TAKES ALL</i>	84
6.2.1 EXEMPLO ILUSTRATIVO	86
6.3 RESULTADOS DA APLICAÇÃO DA TÉCNICA PROPOSTA EM PROBLEMAS DO <i>TSPLIB</i>	89
7 CONCLUSÕES E TRABALHOS FUTUROS.....	101
7.1 RESULTADOS.....	101
7.2 SUGESTÕES PARA FUTUROS TRABALHOS	102
REFERÊNCIAS.....	107
LISTA DE FIGURAS.....	viii
LISTA DE TABELAS.....	xi
RESUMO.....	xii
ABSTRACT.....	xiii

LISTA DE FIGURAS

FIGURA 1.1 – O PROBLEMA DA DESIGNAÇÃO LINEAR	15
FIGURA 1.2 – FLUXOGRAMA DA METODOLOGIA PROPOSTA PARA RESOLVER O PROBLEMA DA DESIGNAÇÃO	18
FIGURA 1.3 – FLUXOGRAMA DA METODOLOGIA PROPOSTA PARA RESOLVER O PROBLEMA DO CAIXEIRO VIAJANTE.....	19
FIGURA 3.1 – EXEMPLOS DE SOLUÇÕES DE UM PROBLEMA DE <i>MATCHING</i> PONDERADO (A) E DE UM PROBLEMA DA DESIGNAÇÃO (B)	40
FIGURA 3.2 – EXEMPLO DE UMA REDE NEURAL DE KOHONEN COM SAÍDA BIDIMENSIONAL.....	43
FIGURA 3.3 – VIZINHANÇAS DO TIPO QUADRADA (A) E HEXAGONAL (B).	43
FIGURA 3.4 – EXEMPLO DE CRIAÇÃO DE AGRUPAMENTOS DOS CONJUNTOS DE VÉRTICES V' E V'' EM MAPAS M' E M''	49
FIGURA 3.5 – EXEMPLO DA REPRESENTAÇÃO DE AGRUPAMENTOS DE MAPAS AUTO-ORGANIZÁVEIS ATRAVÉS DA U-MATRIZ.....	51
FIGURA 3.6 – EXEMPLO DA REPRESENTAÇÃO DE AGRUPAMENTOS DE MAPAS AUTO-ORGANIZÁVEIS ATRAVÉS DA U-MATRIZ EM ESCALAS DE CINZA.....	51
FIGURA 3.7 – EXEMPLO DE REPRESENTAÇÕES DE AGRUPAMENTOS PARA O PROBLEMA DE ALOCAÇÃO DE SALAS DE AULA ATRAVÉS DA U-MATRIZ	52
FIGURA 3.8 – EXEMPLOS DA REPRESENTAÇÃO DA U-MATRIZ (A) E DO MAPA DE DENSIDADE (B) PARA O PROBLEMA DE ALOCAÇÃO DE SALAS, COM FORMAÇÃO DE 8 AGRUPAMENTOS.....	53
FIGURA 3.9 – REPRESENTAÇÃO DA ESTRUTURA DA REDE NEURAL RECORRENTE DE WANG.....	55
FIGURA 4.1 – REPRESENTAÇÃO DA VISTA SUPERIOR DOS PRÉDIOS QUE POSSUEM AS SALAS DE AULA UTILIZADAS.	71
FIGURA 4.2 – REPRESENTAÇÃO DA LOCALIZAÇÃO DE CADA BLOCO DIDÁTICO, UTILIZADA COMO TERCEIRA COORDENADA DOS PADRÕES DE ENTRADA DA REDE	72

FIGURA 5.1 – MAPA DE DENSIDADE E U-MATRIZ DE UMA REDE DE KOHONEN 10 × 10, COM 433 PADRÕES DE ENTRADA.....	77
FIGURA 5.2 – ERRO TOPOLÓGICO PARA OS TRÊS GRUPOS DE DADOS	78
FIGURA 5.3 – ERRO DE QUANTIZAÇÃO PARA OS TRÊS GRUPOS DE DADOS	79
FIGURA 5.4 – ERRO MÉDIO QUADRÁTICO PARA OS TRÊS GRUPOS DE DADOS ...	79
FIGURA 5.5 – ERRO MÉDIO ENCONTRADO PARA A ALOCAÇÃO DAS DISCIPLINAS PARA AS SALAS DE AULA	81
FIGURA 6.1 – EXEMPLO DO PROBLEMA DO CAIXEIRO VIAJANTE COM 10 CIDADES	86
FIGURA 6.2 – (A) SOLUÇÃO FACTÍVEL, ENCONTRADA COM O MÉTODO PROPOSTO, COM ERRO MÉDIO DE 1,7%. (B) SOLUÇÃO ÓTIMA, ENCONTRADA ATRAVÉS DO MÉTODO PROPOSTO	89
FIGURA 6.3 – SOLUÇÕES FACTÍVEIS ENCONTRADAS PARA O PCB442 ATRAVÉS: (A) DO MÉTODO PROPOSTO PURO, COM ERRO MÉDIO DE 9,16%; (B) DO MÉTODO PROPOSTO COM A TÉCNICA 2- <i>OPT</i> , COM ERRO MÉDIO DE 2,87%.	92
FIGURA 6.4 – RESULTADOS DO PROBLEMA BIER127 COM A TÉCNICA “PURA” PROPOSTA NESTE TRABALHO E COM A TÉCNICA DE MELHORIA 2- <i>OPT</i>	94
FIGURA 6.5 – RESULTADOS DO PROBLEMA ATT532 COM A TÉCNICA “PURA” PROPOSTA NESTE TRABALHO E COM A TÉCNICA DE MELHORIA 2- <i>OPT</i>	95
FIGURA 6.6 – ERROS MÉDIOS ENCONTRADOS NOS PROBLEMAS SIMÉTRICOS DO CAIXEIRO VIAJANTE	97
FIGURA 6.7 – ERROS MÉDIOS ENCONTRADOS NOS PROBLEMAS ASSIMÉTRICOS DO CAIXEIRO VIAJANTE.....	98
FIGURA 6.8 – QUALIDADE DAS SOLUÇÕES DE ACORDO COM O NÚMERO DE ITERAÇÕES PARA OS PROBLEMAS GR96, KROC100 E KROD100, COM E SEM A TÉCNICA DE MELHORIA 2- <i>OPT</i>	99
FIGURA 6.9 – QUALIDADE DAS SOLUÇÕES DE ACORDO COM O NÚMERO DE ITERAÇÕES PARA OS PROBLEMAS EIL101, LIN105, RD100 E PR107, COM E SEM A TÉCNICA DE MELHORIA 2- <i>OPT</i>	99
FIGURA A1 – SOLUÇÃO PARA O PROBLEMA BAYS29 COM A TÉCNICA “PURA” (ERRO DE 3,41%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2- <i>OPT</i> (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA).....	104

FIGURA A2 – SOLUÇÃO PARA O PROBLEMA BERLIN52 COM A TÉCNICA ‘PURA’ (ERRO DE 6,96%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2- <i>OPT</i> (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA).....	104
FIGURA A3 – SOLUÇÃO PARA O PROBLEMA ATT48 COM A TÉCNICA ‘PURA’ (ERRO DE 3,99%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2- <i>OPT</i> (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA).....	105
FIGURA A4 – SOLUÇÃO PARA O PROBLEMA PR107 COM A TÉCNICA ‘PURA’ (ERRO DE 3,1%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2- <i>OPT</i> (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA).....	105
FIGURA A5 – SOLUÇÃO PARA O PROBLEMA GR202 COM A TÉCNICA ‘PURA’ (ERRO DE 4,7%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2- <i>OPT</i> (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA).....	105
FIGURA A6 – SOLUÇÃO PARA O PROBLEMA RAT195 COM A TÉCNICA ‘PURA’ (ERRO DE 6,67%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2- <i>OPT</i> (ERRO DE 3,31%)	106
FIGURA A7 – SOLUÇÃO PARA O PROBLEMA EIL51 COM A TÉCNICA ‘PURA’ (ERRO DE 1,38%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2- <i>OPT</i> (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA).....	106
FIGURA A8 – SOLUÇÃO PARA O PROBLEMA EIL76 COM A TÉCNICA ‘PURA’ (ERRO DE 0,48%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2- <i>OPT</i> (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA).....	106

LISTA DE TABELAS

TABELA 4.1 – DISTÂNCIAS ENTRE OS BLOCOS DIDÁTICOS DO CENTRO POLITÉCNICO – UFPR	72
TABELA 5.1 – RESULTADOS PARA AS 32 MATRIZES COM ÚNICO ÓTIMO GLOBAL	74
TABELA 5.2 – RESULTADOS PARA AS 41 MATRIZES COM MÚLTIPLAS SOLUÇÕES ÓTIMAS	75
TABELA 5.3 – RESULTADOS PARA AS 73 MATRIZES TESTADAS	75
TABELA 5.4 – SIGNIFICADO DE CADA AGRUPAMENTO CRIADO NO EXEMPLO DA FIGURA 5.1	77
TABELA 6.1 - PROBLEMAS SIMÉTRICOS DO CAIXEIRO VIAJANTE UTILIZADOS NESTE TRABALHO	90
TABELA 6.2 - PROBLEMAS ASSIMÉTRICOS DO CAIXEIRO VIAJANTE UTILIZADOS NESTE TRABALHO.....	91
TABELA 6.3 – RESULTADOS DOS EXPERIMENTOS PARA O PCV SIMÉTRICO COM AS TÉCNICAS: <i>KNIESG</i> , <i>KNIESL</i> , <i>TS</i> , <i>SOM</i> DE BUDINICH, <i>ESOM</i> , <i>EISOM</i> , <i>SETSP</i> , <i>CAN</i> , RNRW COM <i>WTA</i> . AS SOLUÇÕES EM NEGRITO MOSTRAM OS MELHORES RESULTADOS PARA CADA PROBLEMA, DESCONSIDERANDO-SE OS RESULTADOS DA RNRW COM A TÉCNICA <i>2-OPT</i>	91
TABELA 6.4 – RESULTADOS DOS EXPERIMENTOS PARA O PCV ASSIMÉTRICO COM AS TÉCNICAS: <i>GR</i> , <i>RI</i> , <i>KSP</i> , <i>GKS</i> , <i>RPC</i> , <i>COP</i> E RNRW COM <i>WTA</i> . AS SOLUÇÕES EM NEGRITO MOSTRAM OS MELHORES RESULTADOS PARA CADA PROBLEMA, DESCONSIDERANDO-SE OS RESULTADOS DA RNRW COM A TÉCNICA <i>2-OPT</i>	93
TABELA 6.5 – RESULTADOS PARA TODOS OS PROBLEMAS SIMÉTRICOS DO <i>TSPLIB</i> QUE FORAM UTILIZADOS NESTE TRABALHO.....	96
TABELA 6.6 – RESULTADOS PARA TODOS OS PROBLEMAS ASSIMÉTRICOS DO <i>TSPLIB</i> QUE FORAM UTILIZADOS NESTE TRABALHO.....	97

RESUMO

Neste trabalho são apresentadas duas Redes Neurais Recorrentes para resolver o problema da Designação Linear. Na fase inicial do problema, onde os elementos da matriz de custos do problema da Designação devem ser determinados, utiliza-se Mapas de Kohonen, conhecidos também como Mapas Auto-Organizáveis, e na resolução do problema da Designação propriamente dito, a técnica utilizada é a Rede Neural Recorrente de Wang, com a aplicação de um princípio aqui proposto, denominado *Winner Takes All*. A fase de definição dos custos na resolução de um problema da Designação é de grande importância, pois se os custos não forem determinados de forma adequada, a solução final não será a ideal. O cálculo de custos para problemas da Designação com a utilização de Redes Neurais Artificiais é um assunto pouco explorado, que depende do tipo de aplicação pretendida. Quando a matriz de custos do problema da Designação é tal que admite múltiplas soluções ótimas, ou soluções ótimas locais muito próximas, a Rede Neural de Wang não converge, e a proposta apresentada neste trabalho mostra a utilização do princípio *Winner Takes All* para esta rede, obtendo-se soluções ótimas globais na maioria das matrizes testadas, utilizando-se aproximadamente 1% do número necessário de iterações da Rede de Wang original. Neste trabalho são apresentados os resultados da aplicação desta técnica (a Rede Neural Recorrente de Wang com o princípio *Winner Takes All*) para 73 matrizes com custos definidos aleatoriamente para o problema da Designação, além de alguns critérios para ajustes de parâmetros da Rede Neural de Wang, entre eles alguns tradicionais, e outros que utilizam medidas de dispersão entre os elementos da matriz de custos do problema. A metodologia proposta neste trabalho é aplicada em um estudo de caso: o Problema de Alocação de Salas de Aula para disciplinas de graduação e pós-graduação da UFPR, onde são testados mapas com diversas dimensões para a determinação dos custos deste problema. Os resultados encontrados com a aplicação desta metodologia no estudo de caso são considerados satisfatórios, com erro médio na solução final da Designação inferior a 3% para os melhores mapas encontrados. Uma outra aplicação da Rede Neural de Wang com o princípio *Winner Takes All* é a resolução do problema clássico do Caixeiro Viajante, com soluções ótimas globais em vários problemas do banco de dados *TSPLIB*, e com soluções ótimas locais com erros inferiores a 16%. Para aplicar a metodologia proposta neste trabalho para o problema do Caixeiro Viajante uma adaptação do princípio *Winner Takes All* é feita, obtendo-se sempre rotas factíveis para este problema. A mesma técnica é utilizada para problemas do Caixeiro Viajante simétricos e assimétricos, e a técnica *2-opt* é utilizada para melhorar as soluções encontradas.

ABSTRACT

In this work two Recurrent Neural Networks are presented to solve the Linear Assignment problem. In the initial phase of the problem, where the Assignment problem's costs' must be determined, uses Kohonen Maps, also known as Self-Organizing Maps, and in the Assignment's problem resolution properly said, the used technique is the Wang's Recurrent Neural Network, with the application of a called "Winner Takes All" principle, proposal in this work. The phase of definition of the costs in the resolution of the Assignment problem is of great importance, therefore if the costs will not be determined of adjusted form, the final solution will not be the ideal. The calculation of costs for Assignment problems with the use of Artificial Neural Networks is a subject little explored, that depends on the type of intended application. When the Assignment problem's cost's matrix is such that admits multiple global optimal solutions, or very next local optimal solutions, the Wang's Neural Network does not converge, and the proposal presented in this work shows the use of the principle "Winner Takes All" for this network, getting global optimal solutions in the majority of the tested matrices, using approximately 1% of the necessary number of iterations of the original Wang's Network. In this work the results of the application of this technique (Wang's Recurrent Neural Network with the principle "Winner Takes All") are presented for 73 matrices with random costs to the Assignment problem, beyond some criteria for adjustments of parameters of the Wang's Neural Network, between them some traditional, and others that use measured of dispersion enter the problem's costs. The methodology proposal in this work is applied in a case study: the Problem of Allocation of Classrooms for disciplines of graduation and post-graduation of the UFPR, where maps with many dimensions for the determination of the problem's costs are tested. The results found with the application of this methodology in the case study are considered satisfactory, with average error in the solution of the Assignment problem less than 3% for the best-founded maps. Another application of the Wang's Neural Network with the principle "Winner Takes All" is the resolution of the classic Traveling Salesman problem, with global optimal solutions in some problems of the data base TSPLIB, and with local optimal solutions with errors less than 16%. To apply the methodology proposal in this work for the Traveling Salesman problem an adaptation of "Winner Takes All" principle is made, getting always feasible routes for this problem. The same technique is used for symmetrical and asymmetrical Traveling Salesman problems, and the technique 2-opt is used to improve the founded solutions.

CAPÍTULO I

1 INTRODUÇÃO

1.1 DESCRIÇÕES DOS PROBLEMAS

Uma nova metodologia é apresentada neste trabalho para resolver dois problemas clássicos da Pesquisa Operacional: o problema da Designação Linear e o problema do Caixeiro Viajante (PCV).

Considere o conjunto de vértices V de um grafo bi-partido G , e os subconjuntos V' e V'' do conjunto V , onde cada elemento do conjunto V' possui um determinado custo para sua designação a um elemento do conjunto V'' , e este custo é denotado por c_{ij} . O problema da Designação Linear consiste em designar cada elemento do conjunto V' a um único elemento do conjunto V'' , minimizando o custo total das designações (CHRISTOFIDES, 1975 e AHUJA; MAGNANTI; ORLIN, 1993). Na figura 1.1 encontra-se a representação de um grafo bi-partido com os arcos representando custos para as designações entre os vértices dos conjuntos mencionados.

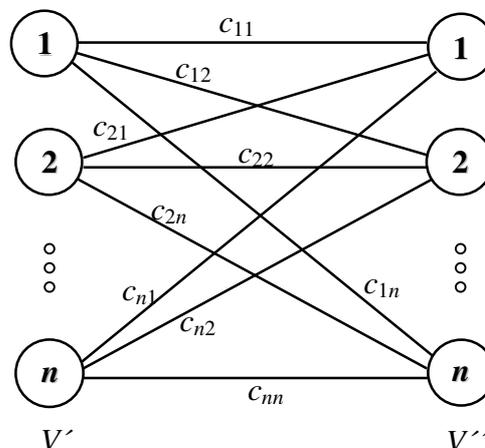


FIGURA 1.1 – O PROBLEMA DA DESIGNAÇÃO LINEAR

Seja um conjunto de pontos representando n cidades. O problema do Caixeiro Viajante consiste na determinação de uma rota que inicia em uma cidade, passa por cada cidade do conjunto apenas uma vez, e retorna à cidade inicial da rota perfazendo uma distância total mínima. Esta rota é denominada ciclo Hamiltoniano de custo mínimo (MURTY, 1985). A representação deste problema pode ser feita através de um grafo bi-partido, onde cada subconjunto de vértices deste grafo contém as n cidades consideradas, e os custos para a designação de cada par de cidades representam a distância Euclidiana ou o custo da viagem entre estas cidades.

A mesma formulação do problema da Designação pode ser utilizada para resolver o problema do Caixeiro Viajante, com a restrição adicional da determinação de um ciclo Hamiltoniano (MURTY, 1985). Desta forma, a mesma metodologia proposta neste trabalho é utilizada para os dois problemas considerados, com algumas alterações para satisfazer a restrição adicional do problema do Caixeiro Viajante.

1.2 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é mostrar a aplicação de Redes Neurais Recorrentes para resolver os problemas da Designação Linear e do Caixeiro Viajante (PCV). Na primeira fase do problema da Designação, a determinação dos elementos da matriz de custos é feita utilizando-se a Rede de Kohonen, mais conhecida na literatura como mapa Auto-Organizável (*Self-Organizing map* – *SOM*). Na resolução do problema da Designação propriamente dito, a Rede Neural Recorrente de Wang (RNRW) é utilizada com a introdução de um princípio proposto neste trabalho, denominado “o vencedor leva tudo” (*Winner Takes All* – *WTA*). A mesma metodologia para resolver o problema da Designação é também utilizada para resolver o problema do Caixeiro Viajante, com pequenas alterações no princípio *Winner Takes All* de forma a obter rotas factíveis para este problema.

Na fase inicial do problema da Designação, são criados mapas Auto-Organizáveis com diferentes dimensões, com o objetivo de comparar as soluções obtidas com as matrizes de custos construídas através destes mapas com as soluções ótimas conhecidas para estas matrizes.

Além disso, são apresentados neste trabalho alguns critérios para ajustes de parâmetros da Rede Neural de Wang, entre eles alguns tradicionais, e outros que utilizam medidas de

dispersão entre os elementos da matriz de custos do problema. Foram criadas 73 matrizes com custos aleatórios para o problema da Designação, e os resultados da Rede de Wang com a utilização destes parâmetros e o princípio *Winner Takes All* para estas matrizes são mostrados neste trabalho, comprovando a eficiência desta nova técnica em comparação com a Rede Neural de Wang pura.

A metodologia proposta neste trabalho para o problema da Designação é aplicada em um problema real: o problema de Alocação de Salas de aula para as disciplinas de graduação e pós-graduação da Universidade Federal do Paraná (UFPR). São criadas 23 matrizes de custos através de mapas Auto-Organizáveis com dimensões diferentes, e os resultados encontrados com a aplicação da Rede de Wang com o princípio *Winner Takes All* para estas matrizes são avaliados, mostrando que problemas similares podem ser resolvidos da mesma forma.

A metodologia foi adaptada para o problema do Caixeiro Viajante e aplicada em problemas simétricos e assimétricos do banco de dados do problema do Caixeiro Viajante *TSPLIB (Traveling Salesman Problem Library)* (REINELT, 1991; *TSPLIB95*, 2005). Os parâmetros utilizados para este problema são os que resultam em melhores resultados para as 73 matrizes testadas para o problema da Designação. São feitas comparações com as seguintes técnicas: Redes de Kohonen adaptadas para resolver o problema do Caixeiro Viajante; Têmpera Simulada (*Simulated Annealing*); e técnicas heurísticas de inserção de remoção de arcos.

1.3 IMPORTÂNCIA DO TRABALHO

A fase inicial de um problema da Designação é extremamente importante, pois a solução final para problemas deste tipo depende da escolha dos custos para a obtenção de resultados satisfatórios. Se a construção da matriz de custos não é adequada, a solução final não é ideal. O cálculo de custos para problemas de Designação com a utilização de Redes Neurais Artificiais (RNA) é um assunto pouco explorado, que depende do tipo de aplicação pretendida, sendo que na maioria das publicações que tratam da resolução deste tipo de problema é omitido.

Neste trabalho, a metodologia proposta para determinar os custos para um problema da Designação tem o objetivo de criar agrupamentos de vértices com características similares

com a utilização de um mapa Auto-Organizável. Os custos para os pares de vértices de um problema da Designação são calculados levando-se em conta os pesos dos neurônios do mapa, eliminando-se pequenas diferenças que podem surgir no cálculo dos custos de vértices com características similares. Desta forma, um determinado conjunto de vértices com características similares é associado a um mesmo neurônio, e os custos entre os vértices são determinados através das características deste neurônio. Portanto, custos entre vértices associados a um mesmo neurônio são nulos, embora existam pequenas diferenças entre as características destes vértices. Os custos entre vértices associados a neurônios vizinhos do mapa possuem custos reduzidos, e os vértices associados a neurônios muito afastados do mapa possuem os maiores custos.

Quando a matriz de custos do problema da Designação admite múltiplas soluções ótimas, ou soluções ótimas locais muito próximas da solução ótima global, a Rede Neural de Wang não converge para uma solução, e a proposta apresentada neste trabalho para “corrigir” este problema mostra a utilização do princípio *Winner Takes All* para esta rede, obtendo-se soluções ótimas globais em mais de 61% das matrizes testadas. Com este princípio aplicado à Rede Neural de Wang, além da mesma convergir, o processo de convergência torna-se bem mais rápido, necessitando cerca de 1% do número de iterações necessárias para a convergência da Rede Neural original. Desta forma, a metodologia proposta neste trabalho mostra-se eficiente para resolver problemas da Designação de grande escala, pois o número de iterações da Rede Neural é bem reduzido, e as soluções são satisfatórias. Um esquema das fases da proposta de metodologia para a resolução do problema da Designação é apresentado na Figura 1.2 a seguir:

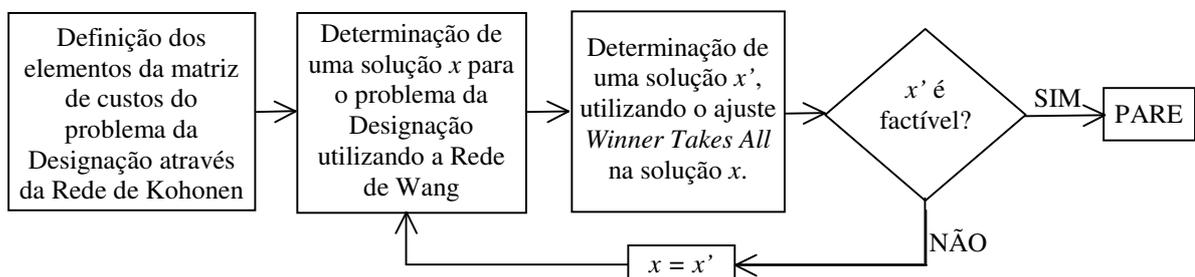


FIGURA 1.2 – FLUXOGRAMA DA METODOLOGIA PROPOSTA PARA RESOLVER O PROBLEMA DA DESIGNAÇÃO

A aplicação desta metodologia adaptada ao problema do Caixeiro Viajante determina resultados satisfatórios, com erro médio abaixo de 16% e soluções com valores próximos aos valores encontrados com a utilização de mapas Auto-Organizáveis (ARAS; OOMEN;

ALTINEL, 1999, BUDINICH, 1996, COCHRANE; BEASLEY, 2003, VIEIRA; DORIA NETO; COSTA, 2002, JIN et al., 2003, LEUNG; JIN; XU, 2004), Têmpera Simulada (BUDINICH, 1996) e de heurísticas de inserção e remoção de arcos (GLOVER et al., 2001). A utilização da técnica de melhoria *2-opt* (LIN; KERNIGHAN, 1973, BIANCHI; KNOWLES; BOWLER, 2005) nas soluções encontradas pela técnica proposta determina melhores resultados na maioria dos problemas testados do banco de dados *TSPLIB*. A técnica proposta neste trabalho possui a vantagem da versatilidade para resolver tanto os problemas do Caixeiro Viajante simétricos quanto os assimétricos, além de convergir para rotas factíveis em poucas iterações ao se utilizar a Rede Neural de Wang. Um esquema das fases da proposta de metodologia para a resolução do problema do Caixeiro Viajante é apresentado na Figura 1.3 a seguir:

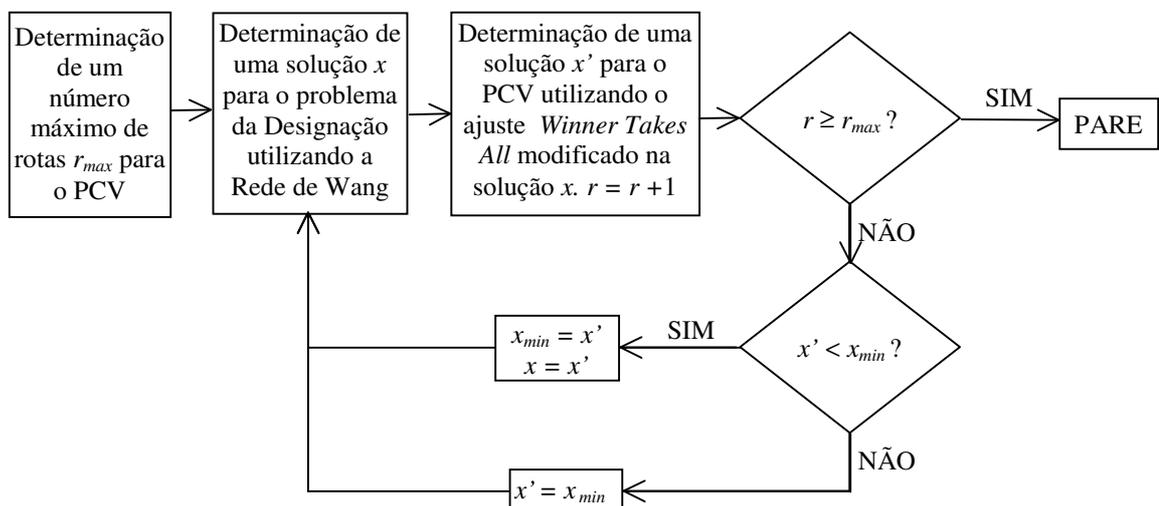


FIGURA 1.3 – FLUXOGRAMA DA METODOLOGIA PROPOSTA PARA RESOLVER O PROBLEMA DO CAIXEIRO VIAJANTE

A aplicação da técnica de melhoria *2-opt* é feita na solução encontrada com o ajuste *Winner Takes All* (x'), e esta solução com melhoria é apresentada para a Rede de Wang somente quando seu valor for o menor encontrado até aquele momento (x_{min}). O mesmo critério vale para a escolha da solução que deve ser apresentada à Rede de Wang sem a utilização da melhoria *2-opt*.

1.4 LIMITAÇÕES DO TRABALHO

Uma das dificuldades encontradas neste trabalho foi a definição dos parâmetros da Rede Neural de Wang. As referências para a aplicação destes parâmetros mostram a utilização

de elementos relacionados à matriz de custos do problema da Designação, como o custo máximo (WANG, 1997, XIA; WANG; HUNG, 1999; LIU; HU; WANG, 2004). São considerados satisfatórios os resultados encontrados com a aplicação de parâmetros que utilizam medidas de dispersão entre os custos do problema, ou seja, média e desvio padrão, pois estes parâmetros determinam soluções ótimas globais em mais de 63% das matrizes testadas neste trabalho. Em mais de 61% das matrizes testadas para o problema da Designação, a solução ótima global foi encontrada com os parâmetros propostos neste trabalho.

As dimensões das matrizes testadas para o problema da Designação variam entre 3×3 e 20×20 , que podem ser consideradas matrizes com dimensões irrelevantes, pois podem ser resolvidas através do método Húngaro (CHRISTOFIDES, 1975) ou da própria formulação do problema através de algum *software* capaz de resolver problemas de Programação Linear Inteira (LINGO, 1999). Os testes feitos para o problema do Caixeiro Viajante, os quais utilizam a mesma técnica usada para resolver o problema da Designação, utilizam matrizes bem maiores, retiradas do *TSPLIB* com dimensões variando entre 10×10 e 532×532 , comprovando a eficiência da técnica proposta neste trabalho para problemas de grande escala.

1.5 INOVAÇÕES PROPOSTAS NESTE TRABALHO

A implementação de mapas Auto-Organizáveis para determinar os custos para o problema da Designação é uma das inovações deste trabalho, pois a simples comparação das características entre os vértices do grafo bi-partido do problema da Designação é a forma mais utilizada para determinar os custos do problema da Designação (SIQUEIRA, 1999, SIQUEIRA et al., 2004).

As propostas apresentadas neste trabalho para os parâmetros da Rede Neural de Wang que utilizam média e desvio padrão entre os elementos da matriz de custos do problema da Designação, são consideradas inovadoras, pois os parâmetros propostos na literatura para esta rede utilizam o custo máximo da matriz ou o número máximo de iterações da rede (WANG, 1997, XIA; WANG; HUNG, 1999; LIU; HU; WANG, 2004).

Outra inovação apresentada neste trabalho é a definição dos parâmetros na forma vetorial, possibilitando que as linhas da matriz de decisão do problema da Designação tenham

penalidades similares quando os custos máximos de cada linha são apresentados à Rede Neural.

A aplicação do princípio *Winner Takes All* na Rede Neural de Wang também representa uma inovação para aplicações em Redes Neurais Recorrentes. A aplicação deste princípio garante a determinação de soluções factíveis em todos os casos testados, com soluções ótimas globais entre 61% e 95% das matrizes de custos utilizadas neste trabalho para o problema da Designação, além de erros médios inferiores a 8% no problema da Designação e 16% para os problemas do Caixeiro Viajante do banco de dados *TSPLIB*.

1.6 ESTRUTURA DO TRABALHO

Este trabalho está dividido em 7 capítulos, incluindo-se esta introdução.

No capítulo II são apresentadas algumas referências de trabalhos sobre os problemas da Designação e do Caixeiro Viajante, além de aplicações da Rede de Kohonen e de aplicações das Redes Neurais na resolução dos problemas da Designação e do Caixeiro Viajante.

No capítulo III são definidos o problema da Designação e as Redes Neurais utilizadas neste trabalho: a Rede de Kohonen e a Rede Neural de Wang. São definidos três tipos de erros para medir a qualidade dos mapas Auto-Organizáveis, e duas técnicas de visualização para estes mapas são mostradas. A aplicação do princípio *Winner Takes All* à Rede de Wang é mostrada através de um exemplo, e um algoritmo para a utilização desta nova técnica é apresentado. Além disso, são apresentadas algumas técnicas para ajustar os parâmetros da Rede Neural de Wang com a utilização de medidas de dispersão entre os elementos da matriz de custos do problema.

O capítulo IV mostra um estudo de caso para este trabalho: o problema de Alocação de disciplinas para as salas de aula da UFPR. São definidas as restrições para a construção das matrizes de custos deste problema e os dados de entrada para o treinamento da rede e para a obtenção da designação.

O capítulo V apresenta os resultados encontrados com as Redes Neurais definidas no capítulo III. São mostrados os resultados da utilização da Rede Neural de Wang com o princípio *Winner Takes All* para 73 matrizes com custos aleatórios para o problema da Designação. Além disso, são apresentadas as performances dos 23 mapas Auto-Organizáveis

e seus respectivos erros para o problema de alocação de disciplinas, assim como as matrizes construídas com a utilização destes 23 mapas que são utilizadas para a resolução do problema de Alocação de disciplinas para as salas de aula.

No capítulo VI são definidos o problema do Caixeiro Viajante e o princípio *Winner Takes All* modificado para resolver este problema. São mostrados os resultados da utilização da técnica proposta neste trabalho para resolver o problema do Caixeiro Viajante para matrizes simétricas e assimétricas do *TSPLIB*, além de mostrar os resultados de outras técnicas para estas matrizes.

No capítulo VII são feitas as considerações finais do trabalho, bem como algumas sugestões para futuros trabalhos.

CAPÍTULO II

2 REVISÃO DA LITERATURA

2.1 INTRODUÇÃO

Neste capítulo são citadas referências de algumas técnicas que vêm sendo utilizadas para a resolução dos problemas da Designação e do Caixeiro Viajante, além de publicações que mostram diversas aplicações das Redes Neurais Artificiais, em particular a Rede de Kohonen e a Rede Recorrente de Wang, que são as redes utilizadas neste trabalho.

Para resolver problemas de otimização, alguns trabalhos mostram adaptações em técnicas conhecidas, tais como as Redes Neurais e os Algoritmos Genéticos, e outras técnicas que consistem em adaptações dos próprios métodos exatos, como pode ser visto na próxima seção.

2.2 MÉTODOS PARA A RESOLUÇÃO DO PROBLEMA DA DESIGNAÇÃO

O Problema da Designação é um problema clássico de otimização combinatória da Pesquisa Operacional. O objetivo deste problema é designar um número de elementos a um mesmo número de posições, além de minimizar a função linear de custos. De acordo com Ahuja, Magnanti e Orlin (1993) este problema é conhecido como Problema da Designação Linear ou problema de *Matching* com custos, e encontra-se descrito no próximo capítulo.

A resolução clássica do problema da Designação é uma fase preliminar da resolução do problema do Caixeiro Viajante, onde considera-se a solução ótima para a matriz de custos do problema da Designação como uma solução inicial para o problema do Caixeiro Viajante. Se esta solução inicial determina uma rota viável, a solução ótima do problema do Caixeiro Viajante é a própria solução inicial, e, em caso contrário, um método variante do *Branch and*

Bound pode ser utilizado, criando-se vários outros problemas da Designação para determinar a melhor rota para o Caixeiro Viajante.

Além destas, outras técnicas tradicionais, como o método Húngaro e o método Simplex são bastante utilizadas. Outras diversas maneiras de se resolver o problema da Designação têm sido apresentadas, destacando-se os Algoritmos Genéticos (GOLDBERG, 1986) e as Redes Neurais Artificiais (FAUSETT, 1994).

Em problemas de grande escala, isto é, quando a matriz de custos do problema é muito grande (por exemplo, matrizes com dimensão maior do que 500×500), as técnicas tradicionais anteriormente mencionadas não se mostram eficientes, pois o número de restrições e o tempo computacional são elevados. A proposta de solução de Siqueira et al. (2004) para o problema de construção de jornadas de trabalho para motoristas e cobradores de ônibus mostra que o modelo da Designação pode ser implementado em cidades que possuem as tabelas dos motoristas e cobradores separadas em grupos, de acordo com o tipo de linhas de ônibus, dividindo o problema da Designação de jornadas de trabalho para funcionários em vários subproblemas de menor escala. Os maiores grupos apresentados possuem aproximadamente 300 jornadas de trabalho.

Um algoritmo alternativo para o problema da Designação é apresentado por Ji, Lee e Li (1997) onde um problema P' é criado utilizando-se relaxações nas restrições do problema da Designação, e trabalhando-se com o problema dual de P' . O algoritmo trabalha de forma similar ao método dual *Simplex*, obtendo-se uma solução factível para o problema P' e, através da teoria da Dualidade, encontra-se uma solução ótima para o problema da Designação.

Em Harper et al. (2005) encontra-se uma proposta para a solução do problema da alocação de projetos para estudantes de graduação e pós-graduação da Universidade de Southampton, Inglaterra, utilizando-se Algoritmos Genéticos. Os estudantes escolhem os projetos por ordem de preferência, criando-se uma matriz de custos. Um cromossomo é a representação de uma solução factível, com a alocação de um projeto por estudante, permitindo-se em alguns casos, mais de um estudante por projeto no início do processo. A função de *fitness* mede a satisfação de cada estudante com o projeto alocado, e através de técnicas similares às apresentadas por Chu e Beasley (1997) e Wilson (1997) várias soluções factíveis são encontradas, com média de erros entre 4% e 7% em matrizes com dimensões variando entre 100×100 e 400×400 .

O trabalho de Haken, Schanz e Starke (1999) mostra o uso de equações do movimento para resolver o Problema da Designação. As variáveis booleanas do problema da Designação são transformadas em contínuas, formando-se um sistema de equações dinâmicas (aproximação *synergetic*) a partir das restrições do problema da Designação. Os parâmetros iniciais do sistema são definidos com o objetivo de determinar um conjunto de pontos assintoticamente estáveis do sistema, que consistem em soluções factíveis do problema da Designação. Estas soluções são ótimas globais na maioria dos casos testados, com média de erros entre 0,0003% e 0,04% para matrizes de dimensão entre 10×10 e 300×300 , respectivamente.

2.3 APLICAÇÕES DAS REDES NEURAI ARTIFICIAIS

Uma Rede Neural Artificial é definida como um sistema de processamento de informações com características de aprendizado semelhantes às das Redes Neurais biológicas. Geralmente, os modelos de RNA's possuem inspiração biológica, mas são algoritmos computacionais que representam o aprendizado humano de maneira bastante simples, através dos elementos de processamento chamados de neurônios (FAUSETT, 1994).

Existem diversos tipos de Redes Neurais, sendo que a Rede Recorrente de Wang e a Rede de Kohonen são as redes utilizadas neste trabalho. A implementação computacional destas redes não é complexa, e os resultados de aplicações destas redes, que podem ser encontrados em diversos artigos na literatura, mostram que se tratam de técnicas eficientes. As definições destas redes são apresentadas no próximo capítulo. A seguir são feitos alguns comentários de alguns trabalhos que mostram aplicações destas Redes Neurais.

2.3.1 Aplicações dos mapas Auto-Organizáveis

O mapa Auto-Organizável ou Rede Neural de Kohonen é uma rede de duas camadas, que vem sendo utilizada em diversas áreas do conhecimento. A maioria das publicações sobre esta rede trata da resolução de problemas de Reconhecimento de Padrões, de Classificação e de Otimização, de acordo com os trabalhos citados a seguir.

A publicação de Ong e Adibi (1999) mostra que a Rede Neural de Kohonen é uma boa ferramenta para visualização de dados para a classificação de países com características

geográficas semelhantes em agrupamentos. Utilizando a mesma metodologia, a classificação de elementos finitos, através da Rede de Kohonen, pode ser encontrada em Betltzer e Sato (2003), com os seguintes dados de entrada: dimensão do elemento, número de nós, número máximo de graus de interpolação polinomial e número de graus de liberdade por nó. Em ambos os trabalhos, as classificações efetuadas pela Rede de Kohonen determinam agrupamentos com os melhores critérios considerados pelos autores, o que possibilita uma boa visualização para os agrupamentos para os conjuntos de dados apresentados em ambos os trabalhos.

Na publicação de Zampigui, Kavanau e Zampigui (2004), a Rede de Kohonen é utilizada para fazer a classificação de partículas, usando como dados de entrada as imagens das partículas em perspectiva cônica. São feitos testes de variação de tamanho e de forma das imagens, com o objetivo de testar a estabilidade da classificação da Rede Neural. Os resultados encontrados foram considerados satisfatórios pelos autores, com fácil implementação da rede. A mesma metodologia foi utilizada por Coppini, Diciotti e Valli (2004) para fazer a classificação de imagens de raio-X, onde a qualidade dos resultados encontrados foi considerada satisfatória.

Em Bonabeau e Hénaux (1998) um mapa Auto-Organizável é utilizado para visualizar grafos com grande número de vértices. Os agrupamentos de vértices são criados a partir das informações dos arcos existentes do grafo, minimizando o número de vértices conectados pertencentes a agrupamentos distintos e melhorando a visualização do grafo. Os testes mostrados pelos autores mostram a visualização de grafos com número de vértices entre 25 e 200.

Serrano-Cinca (1996) apresenta uma aplicação de Redes de Kohonen para executar diagnósticos de instituições financeiras, classificando as empresas em grupos com risco de falência em diversos níveis, além de grupos de empresas que não apresentam risco de falência. Os dados de entrada são as taxas financeiras de empresas, criando-se grupos com taxas similares. Para determinar a existência de risco de falência e em qual nível este risco se apresenta em um determinado grupo criado, são utilizadas as Redes Neurais de múltiplas camadas (*Multilayer Perceptron*) e a Análise Linear Discriminante da Análise Multivariada.

Algumas comparações entre métodos clássicos para criação de agrupamentos e a Rede de Kohonen são apresentadas em Zaït e Messatfa (1997). A Rede de Kohonen mostrou-se mais estável, com melhor qualidade de criação dos agrupamentos, além de conseguir tempo

computacional bem inferior em comparação aos demais métodos utilizados: Criação de agrupamentos rápida (*FastClust* – FC), Auto-classificação (*AutoClass* – AC) e Análise das relações entre os dados (*Relational data analysis* – RDA).

Com a modelagem apropriada para resolver determinados problemas de otimização, algumas modificações na estrutura das Redes de Kohonen permitem a resolução de alguns problemas clássicos, tais como o problema do Caixeiro Viajante (ARAS; OOMMEN; ALTINEL, 1999, VIEIRA; DORIA NETO; COSTA, 2002a), o problema de Alocação de Recursos (LOZANO et al., 1998, HSIEH; TIEN, 2004), e o problema de Roteamento de Veículos (GOMES; VON ZUBEN, 2001).

Os trabalhos de Hsieh e Tien (2004) e Lozano et. al. (1998) mostram a implementação da Rede de Kohonen na resolução do problema de Alocação de recursos. Hsieh e Tien (2004) mostram a criação de uma metodologia que utiliza um refinamento nas soluções encontradas com a inserção e a retirada de pontos durante o processo. Os resultados encontrados mostram o alcance das soluções ótimas em quase todos os problemas testados. O trabalho de Lozano et. al. (1998) comprova a eficiência da utilização da Rede de Kohonen para resolução do problema de Alocação de recursos.

Com a topologia do tipo unidimensional, a entrada dos padrões utilizando o número de veículos, as coordenadas das localizações dos consumidores e as suas demandas, o problema de Roteamento de veículos é resolvido através da Rede de Kohonen em Gomes e Von Zuben (2001). A proposta dos autores para a criação dos pesos iniciais utiliza informações sobre a localização dos consumidores, com o objetivo de criar grupos similares para minimizar o caminho da rota final de cada veículo.

Outras aplicações interessantes das Redes de Kohonen estão presentes na área médica, como no artigo de Wyns et al. (2004), que mostra a aplicação da Rede de Kohonen para previsão de diagnósticos em pacientes com artrite. Os autores comparam a Rede de Kohonen com a Rede de Múltiplas Camadas, e os erros de classificação encontrados foram de 25% com a Rede de Kohonen e de 46% com a Rede de Múltiplas Camadas.

A aplicação de Redes de Kohonen para prever o nível do mar pode ser encontrada em Ultsch e Röske (2002). Utilizando características coletadas em diversos locais do mar, tais como temperatura da água, direção do vento e direção das ondas, a previsão do nível do mar pode ser feita em locais que possuem características semelhantes. Outras aplicações similares

da Rede de Kohonen são a classificação de peixes em lagos (BROSSE; GIRAUDEL; LEK, 2001), e a classificação de insetos aquáticos (OBACH et al., 2001).

A previsão de consumo diário de energia elétrica através das Redes de Kohonen e de múltiplas camadas *Perceptron*, pode ser encontrada em Beccali et al. (2004). A implementação feita pelos autores utiliza dados sobre o clima da cidade de Palermo, Itália, tais como: umidade do ar, temperatura e radiação solar. Em uma fase preliminar, a Rede de Kohonen foi utilizada para classificar os dados de consumo de energia, com o objetivo de distinguir diferentes perfis de consumo, e a rede de múltiplas camadas *Perceptron* foi utilizada para fazer a previsão de consumo diário de energia. Em Kowalski e Kowalska (2003) a mesma metodologia é utilizada para classificar motores, e prever falhas dos mesmos.

Auffermann, Ngan e Hu (2002) mostram o uso de técnicas estatísticas para medir a qualidade da criação de agrupamentos feita através de um mapa Auto-Organizável. São mostradas técnicas baseadas na variância entre os elementos que estão nos agrupamentos (*intracluster*), *Bootstrap* e no discriminante de Fisher, e os resultados encontrados demonstram a eficiência desta técnica para formação de agrupamentos.

No artigo de Lee e Verleysen (2002) são propostas algumas mudanças nas regras de aprendizado dos mapas Auto-Organizáveis, além de mostrar o conceito de vizinhanças recursivas utilizando princípios da Física. São feitas comparações entre as funções de vizinhanças recursivas, Gaussianas e radiais.

O trabalho de Xu e Li (2002) mostra a utilização da Rede de Kohonen para fazer a classificação de imagens impressas. Os dados de entrada são as imagens impressas, com saída em um mapa 7×7 . Cada protótipo do mapa calcula o número de *pixels* da imagem que têm cores similares, criando agrupamentos que são representados em um mapa de densidade. As zonas de transição entre os agrupamentos possuem cores intermediárias, e uma Rede Neural tipo *Fuzzy* é utilizada para determinar estas cores.

A publicação de Martins, Guimarães e Fonseca (2002) mostra a utilização das Redes de Kohonen para a criação de um classificador de texturas para a busca e recuperação de informações visuais em bancos de imagens de sensoriamento remoto. Com a apresentação de uma pequena parte de uma imagem como padrão de entrada, a Rede Neural reconhece qual é a classe mais próxima deste padrão. Através de um mapa bidimensional 10×10 e algumas simulações dos parâmetros da rede, resultados satisfatórios foram encontrados.

Em Wu e Chow (2004) uma nova técnica de agrupamento de dados em mapas Auto-Organizáveis é apresentada. Esta técnica faz uma análise preliminar do mapa, utilizando medidas de dispersão entre os pesos da rede, como média e desvio padrão. A análise entre os agrupamentos é feita durante todo o processo através do índice de densidade entre os agrupamentos (*Composing Density Between and with clusters – CDbw*), que mede a similaridade entre os vetores de um mesmo agrupamento. Os resultados encontrados com esta técnica são melhores do que os encontrados com técnicas tradicionais de agrupamento de dados, como os métodos das *k*-médias (baseado no critério dos mínimos quadrados), Ligação-Única (*Single-Linkage*), Ligação-Completa (*Complete-Linkage*), Ligação-Centróide (*Centroid-Linkage*) e Ligação-Média (*Average-Linkage*).

A Rede de Kohonen é utilizada para determinar agrupamentos de dados sísmicos nos trabalhos de Fleck et al. (2003) e Radeva e Radev (2005). As características utilizadas para a criação de agrupamentos dos dados sísmicos são: a posição do dado, indicando a camada onde este dado está localizado, e o atributo sísmico detectado para este dado. Os resultados apresentados nestes artigos mostram que a Rede de Kohonen é uma técnica eficiente para a classificação destes dados, e que a visualização dos mapas de classificação permitem que novos dados sejam analisados para previsões do comportamento da superfície terrestre.

A classificação de dados Geoespaciais é feita com a utilização da Rede de Kohonen no trabalho de Silva (2004). Foram utilizadas 10 características para a criação dos agrupamentos, envolvendo as coordenadas geográficas dos locais pesquisados, além das informações sócio-econômicas dos moradores destes locais. São feitas análises dos erros de Quantização e Topológico dos diversos mapas criados com os dados existentes, e a dimensão considerada mais adequada para o mapa efetuar a classificação dos dados utilizados é 20×15 , que representa o mapa com o menor erro de Quantização. O erro Topológico não foi adequado para a decisão sobre a escolha da dimensão do mapa, pois seu comportamento mostrou-se instável. As metodologias utilizadas pelo autor para determinação dos agrupamentos foram os critérios estabelecidos pelos índices de Davies-Bouldin e CDbw, apresentados em Wu e Chow (2004).

Outra metodologia semelhante para agrupamento de dados em mapas Auto-Organizáveis pode ser encontrada em Kiang (2001). Os critérios utilizados para determinação dos agrupamentos foram de mínima variância (melhores resultados), e de distância mínima. O algoritmo proposto foi testado com 3 técnicas conhecidas na literatura: *k*-médias, método de

criação de agrupamentos de Ward (baseado no critério de mínima variância) e MODECLUS (método não paramétrico), e os resultados do algoritmo de Kiang foram melhores em todos os testes.

Vesanto e Alhoniemi (2000) apresentam métodos de criação de agrupamentos aglomerativos e de partição para mapas Auto-Organizáveis. O método aglomerativo faz a determinação de cada protótipo para um grupo, e faz a fusão entre os grupos com vetores de pesos semelhantes. O método de partição inicia com um único grupo determinado pelo mapa, e efetua a divisão do mapa em vários grupos, de acordo com a diferença entre os protótipos de um mesmo grupo. São apresentados vários resultados da aplicação destas técnicas, mostrando que tratam-se de métodos eficientes para a determinação de agrupamentos em Redes de Kohonen.

2.3.2 O uso de Redes Neurais para resolver o problema da Designação

Desde a publicação de Hopfield e Tank (1985) diversos trabalhos sobre a utilização de Redes Neurais para resolver problemas de otimização foram desenvolvidos. As adaptações que devem ser feitas para a resolução de problemas de Programação Linear através das Redes Neurais podem ser encontradas em diversas publicações.

No trabalho de Xia, Wang e Hung (1999) um modelo de Rede Neural Recorrente é apresentado para resolver sistemas de equações e inequações. São apresentados exemplos da aplicação desta Rede Neural, e os autores mostram uma aplicação interessante deste método: a resolução de problemas de Programação Linear.

Um modelo de Rede Neural Recorrente apresentado por Wang (1996) é utilizado para resolver o problema do Caminho Mínimo. A análise teórica deste tipo de Rede Neural pode ser encontrada em Wang (1993a).

Em Matsuda (1998) são apresentados três tipos de Redes Neurais de Hopfield para resolver problemas de Programação Linear. Algumas simulações feitas com estas redes mostram que o melhor modelo apresenta solução ótima em 58% dos casos testados, enquanto o pior tem solução ótima em apenas 1% dos casos testados. Algumas simulações feitas pelo autor utilizam o problema da Designação.

O modelo de Rede Neural de Hopfield proposto por Serpen e Livingston (2000) possui uma metodologia para determinar os limites dos parâmetros da rede, com o objetivo de

garantir a convergência da rede para soluções factíveis de um conjunto de modelos clássicos de problemas de Programação Linear, tais como o problema de Transportes, o problema da Designação e o problema do *Matching* com custos e o problema do Caixeiro Viajante simétrico. A metodologia proposta pelos autores garante a convergência para soluções factíveis em todas as matrizes dos problemas testados.

Em Wang (1992), um tipo de Rede Neural Recorrente é proposto para resolver o problema da Designação, sendo que o número necessário de iterações para chegar-se a uma solução factível, no entanto, é elevado para problemas de grande escala: aproximadamente 15.000. Além disso, em problemas com múltiplas soluções ótimas ou soluções ótimas muito próximas, tal rede não converge, como é mostrado no capítulo III deste trabalho.

A resolução do problema de construção de horários (*timetabling*) para professores, através de Redes Neurais, pode ser encontrado em Carrasco e Pato (2004). Um modelo de Rede de Hopfield com a utilização de uma Rede *Winner Takes All* discreta é apresentada. As variáveis do problema são binárias, e a Rede de Hopfield proposta possui convergência lenta, e obtém soluções infactíveis. Definidas as funções energia que minimizam os custos e satisfazem as restrições do problema, os neurônios da Rede de Hopfield têm suas variáveis processadas pela Rede *Winner Takes All*, mostrando convergência mais rápida, em comparação com a Rede de Hopfield contínua. O número de soluções factíveis obtidas com as técnicas apresentadas pelos autores não é expressivo (entre 0 e 30%) em alguns grupos de problemas testados.

2.4 MÉTODOS PARA A RESOLUÇÃO DO PROBLEMA DO CAIXEIRO VIAJANTE

Outro problema clássico de otimização combinatória da Pesquisa Operacional é o problema do Caixeiro Viajante. Considerando-se n cidades, o objetivo deste problema é iniciar a rota de uma cidade, visitando as demais cidades apenas uma vez, retornando à primeira cidade da rota (MURTY, 1985). A formulação deste problema é a mesma do problema da Designação, com a restrição adicional da criação do ciclo Hamiltoniano.

Devido ao grande número de rotas possíveis para este problema, $(n - 1)!$, sua solução torna-se difícil, e este problema tem sido utilizado nos últimos anos como base de comparação para melhorias em diversas técnicas de otimização, tais como Algoritmos Genéticos, Têmpera

Simulada, Busca Tabu, Busca Local, Colônia de formigas, Redes Neurais, dentre outras. A definição deste problema é apresentada no capítulo VI deste trabalho.

A resolução do problema do Caixeiro Viajante utilizando-se a técnica de Colônia de Formigas pode ser encontrada em Dorigo e Garambardella (1997a e 1997b). A idéia desta técnica baseia-se na capacidade das formigas encontrarem um caminho mínimo até sua colônia. Neste artigo, são mostradas comparações entre a técnica proposta e as técnicas Têmpera Simulada, Algoritmos Genéticos e Redes de Kohonen, e os resultados da técnica de Colônia de Formigas são melhores do que estas três técnicas para as matrizes de custos consideradas. Além disso, a técnica proposta é aplicada com a técnica de melhoria *3-opt*, reduzindo os erros médios que se encontravam entre 0% e 4% para 0% e 0,02% nos grafos considerados.

Outros trabalhos semelhantes, que utilizam a técnica híbrida de Colônia de Formigas com a melhoria *3-opt* para problemas de grande escala do banco de dados *TSPLIB*, podem ser encontrados em: Stutzle et al. (2000); Chu, Roddick e Pan (2004).

O trabalho de Pilat e White (2002) mostra a aplicação de um método híbrido de Algoritmos Genéticos aplicado ao método de Colônias de Formigas para o problema do Caixeiro Viajante simétrico. Os Algoritmos Genéticos são utilizados para determinar quais formigas artificiais servem para melhorar as soluções para o problema.

O trabalho de Glover et al. (2001) mostra alguns métodos heurísticos para resolver o problema do Caixeiro Viajante assimétrico (onde a matriz de custos do problema é assimétrica, ou seja, o grafo é direcionado). As técnicas do caminho de Karp-Steele (Karp-Steele *Path* – *KSP*) e o Karp-Steele geral (*General* Karp-Steele – *GKS*) iniciam com um ciclo, e através da remoção de arcos e da colocação de novos arcos, transformam-no em um ciclo Hamiltoniano. A diferença entre estas duas técnicas é que o *GKS* utiliza todos os vértices do ciclo para as mudanças dos arcos do ciclo, e a outra utiliza somente alguns vértices. Outra técnica mostrada neste artigo é a Contração Recursiva de Caminhos (*Recursive Path Contraction* – *RPC*), que consiste na formação de um ciclo inicial, e através da remoção de arcos de cada sub-rota, transforma o ciclo inicial em um ciclo Hamiltoniano. A técnica de Contração ou Determinação de Caminhos (*Contract Or Path heuristic* – *COP*) é uma combinação das técnicas *GKS* e *RPC*. A técnica heurística “gulosa” (*Greedy heuristic* – *GR*) escolhe o menor arco do grafo e contrai este arco, criando um novo grafo. Fazendo este procedimento até o último arco, uma rota factível é encontrada. A técnica de Inserção

Aleatória (*Random Insertion heuristic – RI*) escolhe inicialmente 2 vértices, insere um vértice aleatoriamente, ainda não escolhido, criando um ciclo, e prossegue repetindo este procedimento até criar uma rota com todos os vértices. Os melhores resultados mostrados para os grafos do banco de dados *TSPLIB* são encontrados na aplicação da técnica *COP*.

Em Burke, Cowling e Keuthen (2001) uma técnica que utiliza Busca Local para as variáveis de decisão do problema do Caixeiro Viajante é apresentada. Esta técnica modifica a vizinhança quando reinicia a busca de uma solução. Os resultados apresentados com esta técnica são comparados com a técnica híbrida com melhorias *3-opt* e *4-opt*, apresentando resultados satisfatórios para os problemas assimétricos apresentados, com erros médios variando entre 0% e 15%.

Outro artigo que utiliza uma técnica de Busca Local é devido a Bianchi, Knowles, e Bowler (2005). São apresentadas algumas expressões de correção para melhorar os métodos de busca local *2-opt* e *1-shift* para o problema do Caixeiro Viajante. Os erros médios apresentados no artigo para matrizes com custos aleatórios são inferiores a 2%.

Em Wood e Downs (1998) quatro algoritmos baseados no algoritmo *Demon*, conhecido também como algoritmo semi-canônico de Monte Carlo, e na técnica Têmpera Simulada são apresentados para resolver problemas de otimização, e estes algoritmos são utilizados na resolução do problema do Caixeiro Viajante devido a Grotschel, de 442 cidades (REINELT, 1991; *TSPLIB95*, 2005). Os algoritmos mostrados possuem erros que variam entre 3,75% e 6,28% para o problema citado.

O artigo de Hopfield e Tank (1985) mostra a Rede Neural de Hopfield para resolver problemas de otimização, além de definir os parâmetros desta rede. Além disso, a adaptação desta rede para resolver o problema do Caixeiro Viajante é mostrada pelos autores, onde os erros médios dos dois exemplos apresentados no artigo são de 60,3% e 18,8% para problemas com 10 e 30 cidades, respectivamente.

No trabalho de Feng e Douligeris (2000) são mostradas as aplicações da Rede Neural de Hopfield para o problema do Caixeiro Viajante simétrico e a equação dinâmica proposta por Aiyer para esta rede. A análise do estado de equilíbrio da Rede de Hopfield é feita através do estudo dos autovalores da matriz de pesos da rede. O erro para o problema eil51 (REINELT, 1991; *TSPLIB95*, 2005) varia entre 3% e 15%.

O trabalho de Wang, Tang e Cao (2002) mostra uma maneira de balancear o termo do conjunto de restrições do problema do Caixeiro Viajante e o termo do custo do problema para

a Rede Neural de Hopfield. Os resultados apresentados são consideravelmente melhores do que os encontrados com a Rede de Hopfield sem a utilização deste termo de equilíbrio na equação desta Rede Neural.

O artigo de Talaván e Yáñez (2002) mostra procedimentos de ajustes dos parâmetros da Rede Neural de Hopfield aplicada ao problema do Caixeiro Viajante, baseando-se nas condições de estabilidade da função energia da Rede Neural e das condições para determinar rotas válidas para o problema do Caixeiro Viajante. Os resultados mostrados no artigo para os problemas do *TSPLIB* possuem erros médios que variam entre 29% e 409%.

O artigo de Martín-Valdivia, Ruiz-Sepúlveda e Triguero-Ruiz (2000) apresenta a Rede Neural de Hopfield para o problema do Caixeiro Viajante com a aplicação de funções de penalidades, e um método de fuga de mínimos locais baseado no método de Lagrange aumentado. Os erros médios apresentados para dois problemas do *TSPLIB* são de 12,4% e 31,8%.

Nas publicações de Kobayashi (1998) e Nagashima, Kobayashi e Obayashi (2000) são apresentadas técnicas para a criação de agrupamentos para os nós do grafo do problema do Caixeiro Viajante, com o objetivo de reduzir o tempo computacional para Redes Neurais Recorrentes com a mesma formulação da Rede de Hopfield (*Clustering Hopfield Neural Network – CHNN*). Os erros médios encontrados em problemas do banco de dados *TSPLIB* encontram-se entre 5,7% e 9,1%.

Na publicação de Tsubakitani e Evans (1998) é mostrada uma maneira de encontrar a melhor dimensão da Lista Tabu para aplicação do método de Busca Tabu para o problema do Caixeiro Viajante simétrico. São aplicadas as técnicas de busca local *2-opt* e *3-opt* para melhorar as soluções encontradas. Os resultados da aplicação da Busca Tabu com o *2-opt* apresentaram erros médios entre 0,45% e 5,31%, e com a técnica *3-opt*, os erros médios ficaram entre 0% e 6,84%. Os problemas apresentados no artigo possuem dimensão variando entre 20 e 100 cidades.

O artigo de Gendreau, Laporte e Semet (1998) mostra a aplicação da técnica de Busca Tabu para resolver o problema do Caixeiro Viajante Seletivo, que consiste na busca de uma rota que visite um subconjunto de cidades, não excedendo um limite de custos previamente escolhido. O algoritmo remove e inclui cidades na rota, automaticamente, criando condições para a criação de soluções que satisfazem as restrições impostas.

A publicação de Liu et al. (2003) mostra a aplicação da técnica de Busca Tabu para resolver o problema do Caixeiro Viajante, onde a escolha dos possíveis candidatos para uma nova solução é feita através de uma nova estratégia, criando dois conjuntos de candidatos: elementos de intensificação e de diversificação. Esta técnica produz soluções com índice de convergência variando entre 23% e 99% para dados to *TSPLIB*.

O artigo de Mulder e Wunsch II (2003) mostra a combinação da Rede Neural do tipo dividir e conquistar, com a técnica de melhoria proposta por Lin e Kernighan (1973) para resolver o problema do Caixeiro Viajante simétrico. Neste artigo são feitas comparações desta técnica com técnicas semelhantes encontradas na literatura para os problemas to *TSPLIB*.

A publicação de Aras, Oommen e Altinel (1999) mostra a técnica da Rede de Kohonen com a implementação de técnicas estatísticas entre os pesos dos neurônios (*Kohonen Network Incorporating Explicit Statistics – KNIES*) para resolver o problema do Caixeiro Viajante simétrico. A aprendizagem da rede consiste em duas fases: a fase de atração, onde os neurônios mais próximos de uma determinada cidade são atraídos para esta cidade (como na aprendizagem da Rede de Kohonen), e a fase de dispersão, onde os pesos dos demais neurônios são modificados com o objetivo que os mesmos fiquem mais distantes da cidade em questão. Os critérios utilizados para definir os elementos que participam da atração são baseados nas relações estatísticas entre os dados de entrada e os pesos da rede. Depois do aprendizado da rede, o método faz a dispersão dos neurônios, com o objetivo de igualar as medidas estatísticas destes neurônios. Se todos os neurônios são envolvidos nesta seleção, o método é denominado global (*KNIESG*), e se alguns neurônios são escolhidos para participar desta seleção, o método é denominado local (*KNIESL*). Os resultados apresentados para alguns problemas do *TSPLIB* para esta Rede Neural mostram erros médios variando entre 0,07% e 12,24%.

O denominado mapa de Kohonen melhorado para resolver o problema do Caixeiro Viajante é apresentado em Zhu e Yang (2003). A estrutura dos neurônios do mapa de Kohonen possui a forma de anel, e os padrões de entrada da rede são as coordenadas das cidades. Com mudanças na atualização da taxa de aprendizagem e nos pesos da rede, esta rede apresenta resultados melhores do que os encontrados com as atualizações conhecidas para este tipo de Rede Neural.

O trabalho de Cochrane e Beasley (2003) mostra a resolução do problema do Caixeiro Viajante simétrico através de mapas de Kohonen, com a idéia de cooperação entre os vizinhos

próximos dos neurônios através da técnica denominada Rede Cooperativa Adaptativa de Kohonen (*Cooperative Adaptive Network – CAN*). Nesta técnica utiliza-se um número maior de neurônios do que o número de cidades do problema. São feitas comparações entre várias Redes Neurais desenvolvidas para resolver este problema.

O trabalho de Ghaziri e Osman (2003) mostra a aplicação dos mapas de Kohonen para o problema do Caixeiro Viajante simétrico, com a estrutura em anel modificada para uma estrutura com duas cadeias. Os resultados apresentados para problemas com dados aleatórios são considerados satisfatórios, com melhorias em relação à aplicação da mesma rede com estrutura em anel.

O artigo de Hu et al. (2002) apresenta os mapas de Kohonen com a utilização de um coeficiente de comparação em escala de cinza (*Grey Relational Coefficient – GRC*) entre os padrões de entrada e os pesos da rede. O *GRC* tenta encontrar as relações entre uma seqüência de referência e uma seqüência comparativa, visualizando a seqüência de referência como um objetivo que cada seqüência de comparação deve atingir. Os resultados apresentados são comparados com outras estruturas de aprendizado das Redes de Kohonen desenvolvidas para classificação e para o problema do Caixeiro Viajante. São apresentados resultados para alguns problemas do *TSPLIB* através desta técnica.

Com a mudança da topologia unidimensional em barra para anel, a inicialização dos pesos de acordo com as maiores distâncias entre as cidades, e a introdução de novas heurísticas para ajustar os parâmetros da rede, Vieira, Doria Neto e Costa (2002a), mostram a resolução do problema do Caixeiro Viajante através da denominada Rede de Kohonen eficiente (*SOM Eficiente para o TSP – SETSP*), com resultados satisfatórios para os grafos do *TSPLIB* e tempo computacional reduzido.

O trabalho de Leung, Jin, e Xu (2004) mostra o chamado mapa Auto-Organizável expandido (*Expanding Self-Organizing Map – ESOM*) para resolver o problema do Caixeiro Viajante simétrico. São feitas mudanças nos critérios de aprendizagem da Rede Neural, com o objetivo de satisfazer as condições de otimalidade do problema. São apresentados resultados da aplicação desta técnica, os quais são comparados com os resultados de mapas Auto-Organizáveis com outros critérios de aprendizagem, além da técnica Têmpera Simulada e do mapa de Kohonen simples devido a Budinich (1996). A publicação de Delgado e Martín (2003) também mostra a Rede Neural de Kohonen simples para resolver o problema do Caixeiro Viajante.

Jin et al. (2003) mostra a aplicação dos Algoritmos Genéticos no sistema evolucionário de mapas Auto-Organizáveis para resolver o problema do Caixeiro Viajante na técnica denominada mapa Auto-Organizável eficiente e melhorado (*Efficient Improved Self Organizing Map – eISOM*), onde cada cromossomo representa um esquema de aprendizagem da Rede Neural. Desta forma, os Algoritmos Genéticos determinam quais são os melhores resultados para o problema encontrado através da Rede de Kohonen. Em cada iteração, a Rede de Kohonen coloca os neurônios próximos dos respectivos dados de entrada (cidades), e ao mesmo tempo os coloca na envoltória convexa determinada pelas cidades. Os resultados desta técnica são comparados com a Rede de Kohonen com diferentes formas de aprendizagem, e esta nova técnica apresenta melhorias em quase todas as matrizes do *TSPLIB*.

Na publicação de Merz e Freisleben (1997) é apresentada a aplicação de Algoritmos Genéticos para o problema do Caixeiro Viajante simétrico, com a aplicação da técnica de Busca Local *2-opt* para cada rota encontrada, com o objetivo de encontrar melhores rotas para as próximas gerações. Os resultados encontrados para o banco de dados *TSPLIB* mostram que esta técnica é capaz de obter soluções ótimas na maioria dos casos citados. A publicação de Qu e Sun (1999) mostra aplicação semelhante de Algoritmos Genéticos, com a introdução da aproximação *synergetic* no cálculo da função de *fitness*.

O trabalho de Affenzeller e Wanger (2003) mostra uma técnica para determinar boas soluções para o problema do Caixeiro Viajante com o uso de Algoritmos Genéticos através das comparações entre o *fitness* de novas soluções com *fitness* da população e dos *fitness* dos novos indivíduos e de seus pais. Os resultados para alguns problemas do *TSPLIB* são melhores do que os resultados encontrados com a aplicação de Algoritmos Genéticos da forma tradicional.

Em Choi, Kim e Kim (2003) encontra-se a aplicação de Algoritmos Genéticos para o problema do Caixeiro Viajante assimétrico. Esta técnica gera tanto soluções factíveis como infactíveis, com o objetivo de melhorá-las através das operações de *crossover*. Desta forma, o espaço de busca é mesclado em regiões factíveis e infactíveis. As soluções para o *TSPLIB* são ótimas na maioria dos casos mostrados no artigo.

O trabalho de Sari, Sherali e Bhootra (2005) mostra uma nova formulação para o problema do Caixeiro Viajante assimétrico, criando restrições para eliminar a criação de sub-

rotas. Os resultados para alguns problemas do *TSPLIB* mostram que esta nova formulação permite a obtenção de soluções ótimas em todos os casos testados.

2.5 AS TÉCNICAS UTILIZADAS NESTE TRABALHO

Com a característica de criar agrupamentos, os mapas Auto-Organizáveis de Kohonen são utilizados para criar grupos de vértices com características similares, e os pesos definidos nestes mapas são utilizados para calcular custos para o problema da Designação. Esta rede não possui implementação muito complexa, e os resultados encontrados na literatura comprovam que esta técnica é eficiente para criação de agrupamentos.

A Rede Neural Recorrente de Wang é específica para resolver o problema da Designação, possui complexidade algorítmica da ordem $o(n^2)$, onde n é o número de vértices de um dos conjuntos do grafo bi-partido, além de ser facilmente implementada computacionalmente. Os resultados da implementação desta Rede Neural encontrados na literatura mostram que esta rede é eficiente para resolver o problema da Designação. Além disso, algumas adaptações nas soluções apresentadas nesta rede para o problema da Designação geram soluções viáveis para o problema do Caixeiro Viajante.

No próximo capítulo são descritas, detalhadamente, as técnicas utilizadas neste trabalho.

CAPÍTULO III

3 RESOLUÇÃO DO PROBLEMA DA DESIGNAÇÃO LINEAR COM A UTILIZAÇÃO DE REDES NEURAS RECORRENTES

3.1 DESCRIÇÃO DO PROBLEMA

Considere o conjunto de vértices V de um grafo bi-partido G , e os subconjuntos V' e V'' do conjunto V , tais que $V' \cup V'' = V$ e $V' \cap V'' = \emptyset$. Cada elemento v'_i do conjunto V' possui um determinado custo para sua designação a um elemento v''_j do conjunto V'' , e este custo é denotado por c_{ij} . Nos casos em que a designação entre dois elementos não é possível, o custo de tal designação é dado por um número tão grande quanto se queira, simbolizado por ∞ .

O problema da Designação consiste em designar cada elemento do conjunto V' a um único elemento do conjunto V'' , minimizando o custo total das designações. Esta definição é a mesma de um conjunto de emparelhamentos ponderados, mais conhecido na literatura como *Matching* ponderado ou ainda *Matching* com custos, com a diferença de que o conjunto de vértices para o problema do *Matching* não necessita formar um grafo bi-partido (CHRISTOFIDES, 1975 e AHUJA; MAGNANTI; ORLIN, 1993). A figura 3.1 mostra exemplos de soluções de um problema de *matching* ponderado e de um problema da designação, respectivamente, com grafos de 7 vértices. Os arcos que não estão representados na figura 3.1 representam uma designação com custo ∞ .

Quando os conjuntos V' e V'' possuem quantidades distintas de vértices, como no exemplo mostrado na figura 3.1B, são criados vértices artificiais pertencentes ao conjunto com menor número de elementos, igualando-se os números de elementos destes conjuntos. Estes vértices possuem custo ∞ com relação aos demais vértices do grafo considerado.

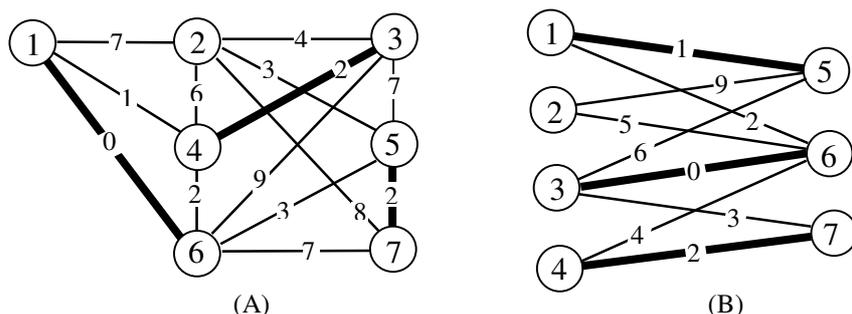


FIGURA 3.1 – EXEMPLOS DE SOLUÇÕES DE UM PROBLEMA DE *MATCHING* PONDERADO (A) E DE UM PROBLEMA DA DESIGNAÇÃO (B)

São diversas as técnicas que podem ser utilizadas para a resolução do problema da Designação, tais como o Método Simplex aplicado na modelagem do problema, o Método Húngaro, *Matching* com pesos, Algoritmos Genéticos, Têmpera Simulada, Redes Neurais, dentre outros.

O problema da Designação pode ser resolvido através do algoritmo do *Matching* com custos (custo mínimo ou máximo). A complexidade algorítmica deste algoritmo é da ordem $O(n^3)$, onde n é o número de vértices do grafo considerado (CHRISTOFIDES, 1975; SIQUEIRA et al., 2004).

Outro algoritmo exato utilizado para resolver o problema da Designação é conhecido na literatura como método Húngaro, que consiste em uma simplificação do algoritmo do *Matching* com custos, desconsiderando-se os cálculos para a formação de ciclos ímpares de vértices (SIQUEIRA, 1999). Estes algoritmos encontram-se na literatura (MURTY, 1985, CHRISTOFIDES, 1975; MINIEKA, 1978; AHUJA; MAGNANTI; ORLIN, 1993) e não são apresentados neste trabalho.

Em problemas de grande escala, ou seja, quando as matrizes de custos tornam-se muito grandes, tanto o método Húngaro quanto o algoritmo do *Matching* com custos não se mostram eficientes, pois seu tempo computacional aumenta consideravelmente (SIQUEIRA et al., 2004).

Neste trabalho são propostas duas Redes Neurais para a resolução do problema da Designação: a Rede de Kohonen para determinar os elementos da matriz de custos do problema, e a Rede Neural Recorrente de Wang, com a utilização do princípio *Winner Takes All*, para resolver o problema da Designação. Uma aplicação da metodologia proposta neste trabalho é a resolução do problema de Alocação de salas de aula para disciplinas de graduação e pós-graduação, implementada na Universidade Federal do Paraná. Este problema está

definido no próximo Capítulo deste trabalho, onde são apresentados alguns resultados encontrados com esta aplicação.

Uma aplicação da Rede Neural de Wang com o princípio *Winner Takes All* é a resolução do problema do Caixeiro Viajante, o qual possui a mesma formulação do problema da Designação, com a restrição adicional da necessidade da formação de ciclos Hamiltonianos de custos mínimos. Este problema é definido no Capítulo VI deste trabalho, onde são apresentados resultados encontrados para alguns problemas do banco de dados *TSPLIB*, os quais são utilizados como comparação entre as diversas técnicas apresentadas na literatura para resolver o problema do Caixeiro Viajante.

Na próxima seção, é apresentada a formulação matemática do problema da Designação e são citados os métodos tradicionais utilizados para resolver este problema.

3.1.1 Formulação do problema da Designação

A formulação matemática para o problema da Designação Linear é dada pelo seguinte Modelo de Programação Linear Inteira Binária:

$$\text{Minimizar } c = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \quad (3.2.1)$$

$$\text{Sujeito a } \sum_{i=1}^n x_{ij} = 1, \quad j = 1, 2, \dots, n \quad (3.2.2)$$

$$\sum_{j=1}^n x_{ij} = 1, \quad i = 1, 2, \dots, n \quad (3.2.3)$$

$$x_{ij} \in \{0, 1\}, \quad i, j = 1, 2, \dots, n \quad (3.2.4)$$

onde c_{ij} e x_{ij} são, respectivamente, os custos e as variáveis de decisão associados à designação do elemento i à posição j . A forma usual de representação de c no método Húngaro é a matricial. Quando $x_{ij} = 1$, o elemento i é designado ao elemento j .

A função objetivo (3.2.1) representa o custo total a ser minimizado. Os conjuntos de restrições (3.2.2) e (3.2.3) garantem que cada elemento i será designado para exatamente uma posição j . O conjunto (3.2.4) representa as restrições de integralidade das variáveis zero-um x_{ij} . O conjunto de restrições (3.2.4) pode ser substituído por restrições do seguinte tipo (BAZARAA; JARVIS; SHERALI, 1990):

$$x_{ij} \geq 0, \quad i, j = 1, 2, \dots, n \quad (3.2.5)$$

Uma proposta de construção da matriz de custos para o problema da designação é mostrada neste trabalho, e a técnica escolhida para esta proposta é definida na próxima seção.

3.2 PROPOSTA DE CONSTRUÇÃO DA MATRIZ DE CUSTOS PARA O PROBLEMA DA DESIGNAÇÃO

A proposta apresentada neste trabalho utiliza a Rede Neural de Kohonen, mais conhecida na literatura como mapa Auto-Organizável, para calcular os custos do problema da Designação.

Os vértices dos conjuntos V' e V'' são classificados em mapas M' e M'' , respectivamente, e o custo da designação de um vértice $v'_i \in V'$ para um vértice $v''_j \in V''$ é determinado pela distância Euclidiana entre os neurônios ou agrupamentos associados aos vértices v'_i em M' e v''_j em M'' .

3.2.1 O mapa Auto-Organizável de Kohonen

O mapa Auto-Organizável é uma Rede Neural de treinamento não supervisionado, que utiliza uma estrutura topológica entre as unidades de seus agrupamentos. Existem r unidades de neurônios, que podem ser denominadas protótipos, arranjos em um mapa de 1, 2 ou 3 dimensões (FAUSETT, 1994), e os sinais de entrada são v vetores com dimensão s .

O vetor peso para cada neurônio serve como modelo de padrão de entrada associado a cada agrupamento. Durante o treinamento da rede, o neurônio cujos pesos mais se aproximam do padrão de entrada é denominado "vencedor". A distância de cada neurônio para cada padrão de entrada é, geralmente, dada pelo quadrado da distância Euclidiana entre estes vetores. O neurônio vencedor i^* e sua vizinhança V_{i^*} (com relação à topologia do mapa) têm os pesos atualizados, embora os vizinhos de um neurônio vencedor não possuam, necessariamente, os pesos próximos ao padrão de entrada. O princípio utilizado em uma Rede de Kohonen é de reforçar as ligações que produzem respostas mais eficientes (KOHONEN, 1998).

O funcionamento desta rede executa uma simulação da atividade cerebral, baseada em estudos sobre as células nervosas corticais, que possuem um arranjo anatômico em relação

aos estímulos recebidos dos sensores ligados às mesmas. Estudos mostram que as células nervosas possuem centros de atividade mais intensos, de acordo com as atividades executadas no momento (TAFNER; XEREZ; RODRIGUES FILHO, 1996), que é o princípio fundamental da Rede de Kohonen.

Na figura 3.2 encontra-se um exemplo de uma Rede de Kohonen com saída bidimensional, que contém 25 neurônios e 1 entrada sendo apresentada à rede. Nesta figura, o protótipo a_{12} representa um vencedor em uma determinada iteração do treinamento da rede, com vizinhança do tipo quadrada representada com os raios 0, 1 e 2. Os tipos de vizinhança podem ser representados de diversas formas, e as formas mais comuns são as vizinhanças quadrada e hexagonal (figura 3.3).

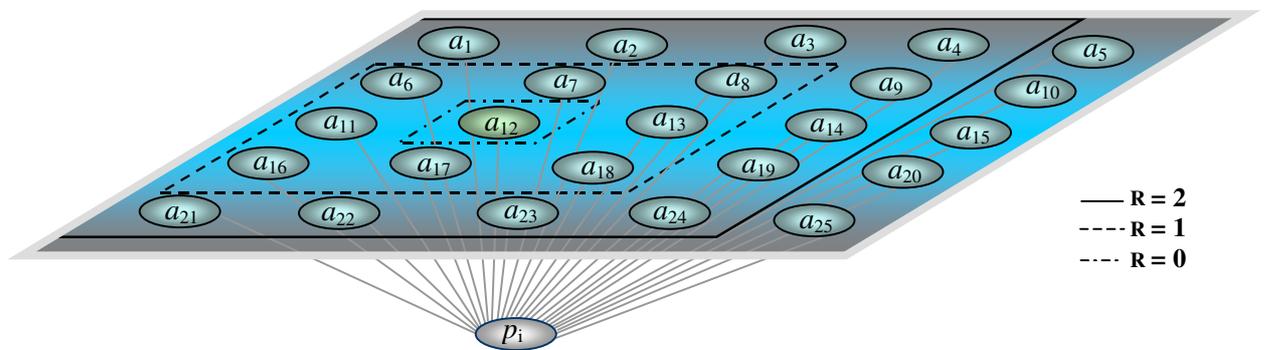


FIGURA 3.2 – EXEMPLO DE UMA REDE NEURAL DE KOHONEN COM SAÍDA BIDIMENSIONAL

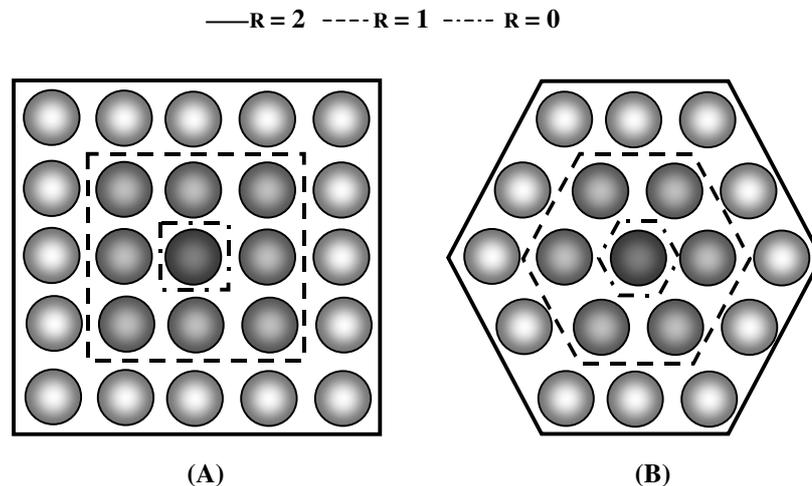


FIGURA 3.3 – VIZINHANÇAS DO TIPO QUADRADA (A) E HEXAGONAL (B).

O conjunto de neurônios vizinhos de um neurônio vencedor i^* com raio unitário é denominado conjunto de vizinhos imediatos, \bar{V}_{i^*} . Em mapas com vizinhança do tipo

quadrada, a cardinalidade destes conjuntos pode ser igual a 8, 5 ou 3, ou seja; 8, quando o neurônio vencedor não está no contorno do mapa; 5, quando o neurônio vencedor pertence ao contorno, mas fora de uma das extremidades do mapa; ou 3, quando o neurônio vencedor está localizado em uma das extremidades do mapa. Em mapas com vizinhança do tipo hexagonal estes números são, respectivamente, iguais a 6, 4 e 3 (WITKOWSKI et al., 1997).

No treinamento da Rede de Kohonen, a ordem de entrada dos padrões pode ser aleatória, e os padrões de entrada devem ser vetores normalizados, com o objetivo padronizar tais vetores e evitar a classificação com base em coordenadas dos vetores de entrada que possuem valores muito altos em comparação com as outras coordenadas (JAIN; MURTY; FLYNN, 1999).

O algoritmo da Rede de Kohonen possui 3 fases. A primeira é a competitiva, onde os neurônios da camada de saída competem entre si, de acordo com um critério, geralmente utiliza-se a distância Euclidiana, definindo um neurônio vencedor. A segunda fase é denominada cooperativa, onde a vizinhança do neurônio vencedor é definida. A última fase é denominada adaptativa, pois os pesos do neurônio vencedor e de sua vizinhança são ajustados (SILVA, 2004).

A função de vizinhança controla a atuação dos neurônios vizinhos em torno do vencedor, e como no modelo biológico, esta atuação diminui quando a distância do vizinho aumenta. As propriedades de convergência do algoritmo da Rede de Kohonen podem ser encontradas nas seguintes publicações: Flanagan (1996) e Cottrell, Fort e Pagès (1998).

O algoritmo para o treinamento de uma Rede de Kohonen unidimensional ou bidimensional com r neurônios é apresentado a seguir, onde a entrada dos padrões é considerada aleatória.

Algoritmo para a Rede de Kohonen

Passo 1: Inicie os pesos (w_{ij}) dos r neurônios da rede, com valores aleatórios pertencentes ao intervalo $[0, 1]$. Faça $t = 1$, determine os valores iniciais do raio da vizinhança R e da taxa de aprendizagem α . Escolha um critério de parada, e vá ao passo 2.

Passo 2: Encontre um padrão de entrada k ainda não apresentado à rede na iteração t , onde $k \in (0, v]$. Se todos os padrões já foram apresentados na iteração t , faça $t = t + 1$, e vá ao Passo 5; caso contrário, apresente à rede a entrada p_k , e vá ao Passo 3.

Passo 3: Determine o neurônio i que possui a menor distância Euclidiana do peso w com o vetor de entrada p_k .

$$d_i = \sum_{j=1}^s (p_{kj} - w_{ij}(t))^2, \text{ para } i = 1, \dots, r. \quad (3.2.6)$$

Este neurônio é denominado “vencedor”, e denotado por i^* . Vá ao passo 4.

Passo 4: Ajuste os pesos do neurônio vencedor e de todos os neurônios que pertencem a uma vizinhança centrada nele, $V_{i^*}(t)$:

$$w_{ij}(t+1) = w_{ij}(t) + \alpha(t) \Lambda_{ii^*}[p_{kj} - w_{ij}(t)], \text{ para } j = 1, \dots, s \text{ e } i \in V_{i^*}(t). \quad (3.2.7)$$

Vá ao passo 2.

Passo 5: Calcule o erro do mapa, ajuste a taxa de aprendizagem α e o raio de vizinhança R . Se o critério de parada for satisfeito, pare; caso contrário, volte ao passo 2.

A função Λ , apresentada em (3.2.7), constitui uma forma de ajustar a vizinhança dos neurônios vencedores. Esta função é usualmente definida de duas formas. Determinado o conjunto de neurônios vizinhos de i^* , V_{i^*} , a função de vizinhança discreta é dada por:

$$\Lambda_{ii^*}(t) = \begin{cases} 1, & \text{se } i \in V(i^*) \\ 0, & \text{caso contrário} \end{cases}. \quad (3.2.8)$$

A função de vizinhança Gaussiana é dada por:

$$\Lambda_{ii^*}(t) = \exp\left(\frac{-d_{ii^*}^2}{2R^2(t)}\right), \quad (3.2.9)$$

onde $R(t)$ representa o raio topológico no mapa, que decresce com o tempo (iterações), e d_{ii^*} representa a distância topológica entre os neurônios i e i^* (KANGAS; KOHONEN, 1996). Esta função de vizinhança cobre todo o mapa inicialmente, mas reduz-se a uma vizinhança de raio zero progressivamente, ou seja, após um certo número de iterações somente o neurônio vencedor é atualizado. Esta redução do raio da vizinhança é feita utilizando-se a seguinte atualização:

$$R(t) = R(0) \exp\left(\frac{-t}{z_1}\right), \quad (3.2.10)$$

onde z_1 é uma constante, dada por $z_1 = 1000/\log(R(0))$ (HAYKIN, 1999), e o raio inicial inicia o processo de aprendizagem cobrindo o mapa todo. A atualização da taxa de aprendizagem do mapa de Kohonen também é exponencial, dada por:

$$\alpha(t) = \alpha(0) \exp\left(\frac{-t}{z_2}\right), \quad (3.2.11)$$

onde z_2 é uma constante, dada por $z_2 = 1000$ (HAYKIN, 1999), e a taxa de aprendizagem inicial, $\alpha(0)$, possui valor menor do que 1, decaindo para valores próximos de zero no final do aprendizado do mapa.

O critério de parada que pode ser utilizado para o “treinamento” da Rede de Kohonen é o alcance de um dos seguintes valores pré-determinados: erro mínimo desejável, número máximo de iterações, ou valor mínimo para a taxa de aprendizagem.

Na seção seguinte são apresentadas algumas formas de medir a qualidade dos mapas Auto-Organizáveis.

3.2.2 Cálculo de erros em mapas Auto-Organizáveis de Kohonen

Existem várias medidas de erro que podem ser utilizadas para determinar a qualidade de um mapa Auto-Organizável, e as três medidas escolhidas neste trabalho encontram-se descritas a seguir.

O Erro de Quantização (KIVILUOTO, 1996; KOHONEN, 2001) é dado por:

$$E_Q = \frac{1}{v} \sum_{k=1}^v \|p_k - w^*\|, \quad (3.2.12)$$

onde v é a quantidade de padrões de entrada apresentados ao mapa e w^* representa o vetor peso do neurônio vencedor para o padrão p_k . Este erro representa o erro médio correspondente à diferença entre os vetores dos padrões de entrada e os respectivos pesos de seus neurônios vencedores.

Já o Erro Médio Quadrático (LEE; VERLEYSSEN, 2002) é calculado da seguinte forma:

$$E_{MQ} = \frac{1}{v} \sum_{k=1}^v \|p_k - w^*\|^2. \quad (3.2.13)$$

Outra medida de erro utilizada para verificar a qualidade do mapa Auto-Organizável é denominada Erro Topológico (KIVILUOTO, 1996; KOHONEN, 2001), definida por:

$$E_T = \frac{1}{v} \sum_{k=1}^v u_k, \quad (3.2.14)$$

onde

$$u_k = \begin{cases} 1, & \text{se } i^{**} \notin \bar{V}_{i^*} \\ 0, & \text{caso contrário} \end{cases}, \quad (3.2.15)$$

onde i^{**} é o segundo neurônio vencedor para o padrão k , e \bar{V}_{i^*} é o conjunto de neurônios vizinhos imediatos do vencedor i^* . O valor do Erro Topológico representa o percentual de neurônios vencedores que não possuem o segundo vencedor em uma vizinhança de raio unitário centrada no neurônio vencedor. Este erro mede quanto o mapa aproxima os padrões de entrada com vetores peso próximos.

A seguir, é apresentada uma proposta de utilização da Rede de Kohonen para calcular os custos entre os vértices de um grafo para o problema da Designação.

3.2.3 Utilização da Rede de Kohonen para construção da matriz de custos do problema da Designação

Com a utilização da Rede Neural de Kohonen, podem ser criados dois mapas, M' e M'' , onde são criados agrupamentos para os vértices dos conjuntos V' e V'' , respectivamente, em cada mapa.

O procedimento inicial para criar os agrupamentos de vértices é o treinamento da rede, que pode ser executado com um determinado número de elementos do conjunto V . Outro conjunto que pode ser utilizado para fazer o treinamento da rede é um conjunto de vértices que contém as características consideradas essenciais para representar agrupamentos distintos de vértices de V . Seja V_e o conjunto de v vértices escolhidos para o treinamento da rede. Os padrões de entrada são vetores de dimensão s , que contém as características escolhidas para fazer os agrupamentos de elementos do conjunto V .

Após o treinamento da rede, são criados agrupamentos para os elementos dos conjuntos V' e V'' nos mapas M' e M'' , respectivamente, onde os mesmos pesos obtidos após o treinamento da rede são considerados para os dois mapas.

Uma das formas propostas neste trabalho para calcular os custos entre os vértices destes conjuntos é dada por:

$$c_{ij} = \|w_f^* - w_g^*\|, \quad (3.2.16)$$

onde w_f^* e w_g^* são, respectivamente, os vetores peso dos neurônios vencedores dos padrões de entrada $i \in V'$ e $j \in V''$.

Cada agrupamento possui um neurônio mais representativo, ou seja, existe um conjunto com número considerável de padrões de entrada relacionados a este neurônio. Este neurônio mais representativo é denominado vencedor ou centróide do agrupamento, e a determinação deste neurônio é feita após o treinamento da rede, com a apresentação dos v padrões de entrada por t iterações.

Outra proposta para calcular os custos pode ser feita após o agrupamento dos vértices em seus respectivos mapas. Após classificar todos os vértices de V' e V'' em seus respectivos agrupamentos em seus mapas, a distância entre o grupo $gr(i)$ de um vértice $i \in V'$, e o grupo $gr(j)$ de $j \in V''$ é a distância entre os centróides dos grupos $gr(i)$ e $gr(j)$, e o custo entre os elementos i e j é dado por:

$$c_{ij} = \|w_{gr(i)}^* - w_{gr(j)}^*\|, \quad (3.2.17)$$

onde $w_{gr(i)}^*$ e $w_{gr(j)}^*$ representam os vetores peso dos centróides dos grupos $gr(i)$ e $gr(j)$, respectivamente.

Na figura 3.4 encontra-se um exemplo de representação para um agrupamento de dois conjuntos de vértices, com os centróides em destaque, e os vértices representados por triângulos. Vale lembrar que o conjunto de pesos criado para os dois mapas deve ser o mesmo, pois em caso contrário, as comparações entre as características dos dois conjuntos perdem o sentido.

O objetivo principal desta nova técnica apresentada para o cálculo dos custos para o problema da Designação é de criar condições para que os pares de vértices tenham custos adequados para a designação. Com a aplicação desta técnica, os vértices que possuem características similares têm custos menores, e quando estes vértices possuem o mesmo

neurônio vencedor em mapas distintos, o custo da designação entre estes elementos é nulo. Os neurônios dos mapas criados servem para representar as características necessárias para a resolução do problema da Designação.

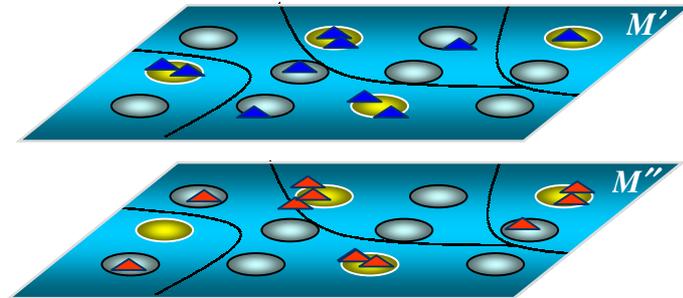


FIGURA 3.4 – EXEMPLO DE CRIAÇÃO DE AGRUPAMENTOS DOS CONJUNTOS DE VÉRTICES V' E V'' EM MAPAS M' E M''

A simples comparação entre os vetores de entrada relativos aos vértices dos conjuntos V' e V'' tende a criar diferenças mínimas entre os vértices com características muito próximas, o que afeta a designação final entre estes conjuntos. Com a Rede de Kohonen, se os vetores possuem valores muito próximos, a tendência é de que os neurônios vencedores dos vértices relacionados a estes vetores sejam coincidentes, ou pertencentes a vizinhanças muito próximas, resultando em custos reduzidos para a designação de tais vértices.

Na próxima seção, são apresentadas algumas técnicas para fazer agrupamentos de dados como os utilizados neste trabalho.

3.2.4 Representação Gráfica de mapas Auto-Organizáveis de Kohonen

Existem diversas técnicas para criação de agrupamentos em mapas Auto-Organizáveis, (JAIN; MURTY; FLYNN, 1999; VESANTO; ALHONIEMI, 2000; KIANG, 2001; WU; CHOW, 2004), sendo que as técnicas utilizadas neste trabalho são descritas a seguir.

3.2.4.1 A matriz de distância unificada: U-matriz

A técnica mais utilizada para visualização de mapas Auto-Organizáveis é denominada U-matriz (Matriz de distância unificada), criada por Ultsch nos anos 80. A U-matriz consiste em uma representação topológica das relações entre os neurônios do mapa, utilizando o

mesmo tipo de distância da fase de treinamento do mapa. Neste trabalho, a distância utilizada para a fase de treinamento da rede é a Euclidiana.

Considerando-se um mapa $b \times q$, a U-matriz é uma imagem deste mapa em um mapa de dimensão $(2b - 1) \times (2q - 1)$, onde são representados os neurônios do mapa, além das distâncias intermediárias entre os neurônios do mapa original através de cores, de acordo com os coeficientes do vetor peso de cada neurônio (ULTSCH, 2003a). Esta representação envolve a escolha de um conjunto de cores associadas às características dos vetores, e cada neurônio possui uma determinada cor representada.

Por exemplo, considerando-se o mapa 3×3 representado na figura 3.5 com vetores peso de 2 dimensões com valores aleatórios, a representação da U-matriz é um mapa 5×5 construído da seguinte forma: o conjunto de cores escolhido é vermelho, verde, azul e branco; as cores são atribuídas aos neurônios, de acordo com os valores das coordenadas dos vetores peso $w(x, y)$: quando $x \in [0; 0,5]$ e $y \in [0; 0,5]$, a cor atribuída ao neurônio é vermelha; quando $x \in [0; 0,5]$ e $y \in (0,5; 1]$, então a cor atribuída ao neurônio é verde; quando $x \in (0,5; 1]$ e $y \in [0; 0,5]$, a cor designada ao neurônio é azul; e se $x \in (0,5; 1]$ e $y \in (0,5; 1]$, a cor atribuída ao neurônio é branca.

A representação dos elementos entre os neurônios vizinhos imediatos do mapa original pode ser feita utilizando-se medidas como a média, a mediana ou o valor máximo entre os vetores pesos dos neurônios vizinhos. Na figura 3.5 foram calculadas as médias entre os pares de neurônios vizinhos.

Outra forma de representação da U-matriz utiliza o cálculo de uma medida para cada neurônio, denominada U-altura:

$$\text{U-altura}(i) = \sum_{j \in V_i} \|w_i - w_j\|, \quad (3.2.18)$$

que representa valores altos para neurônios que se distanciam muito dos vizinhos imediatos, e valores baixos para neurônios com vizinhos próximos que contém vetores-peso similares (ULTSCH, 2003b). A imagem de um mapa $b \times q$ é um mapa de mesma dimensão, com a atribuição de cores em escalas de cinza, com as tonalidades calculadas através da U-altura de cada neurônio. A representação da U-matriz da Figura 3.5 em escalas de cinza é mostrada na Figura 3.6.

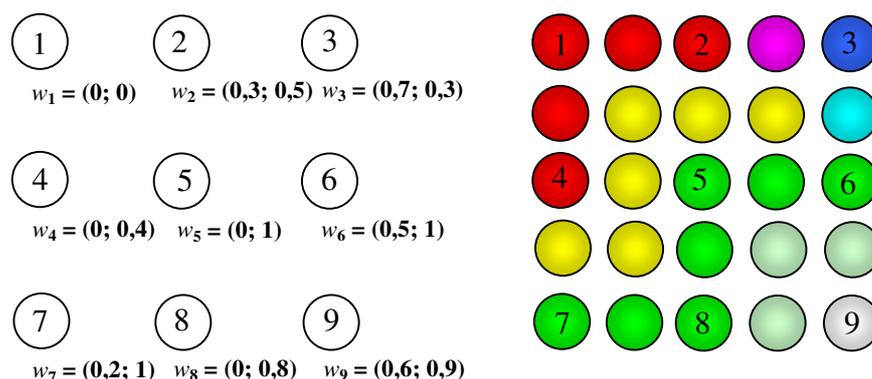


FIGURA 3.5 – EXEMPLO DA REPRESENTAÇÃO DE AGRUPAMENTOS DE MAPAS AUTO-ORGANIZÁVEIS ATRAVÉS DA U-MATRIZ

Os agrupamentos são separados pelas tonalidades mais escuras de cinza. A visualização dos agrupamentos em mapas com número reduzido de neurônios, como no exemplo da Figura 3.6, torna-se difícil, e um exemplo de aplicação desta técnica de visualização para o problema de Alocação de salas de aula, que será abordado no Capítulo IV, encontra-se na Figura 3.7.

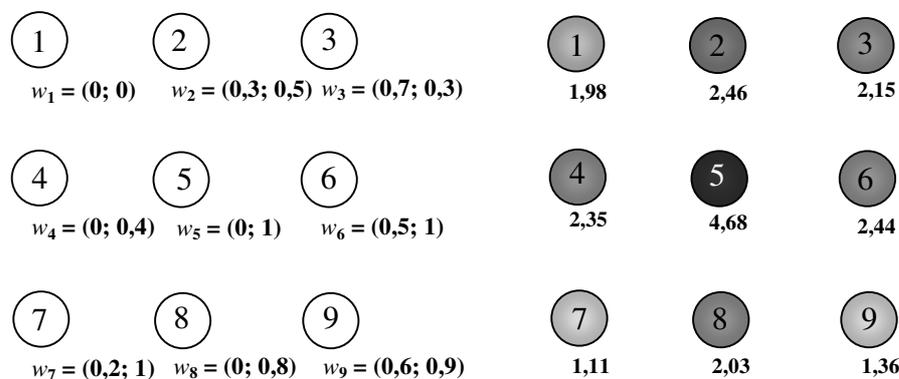


FIGURA 3.6 – EXEMPLO DA REPRESENTAÇÃO DE AGRUPAMENTOS DE MAPAS AUTO-ORGANIZÁVEIS ATRAVÉS DA U-MATRIZ EM ESCALAS DE CINZA

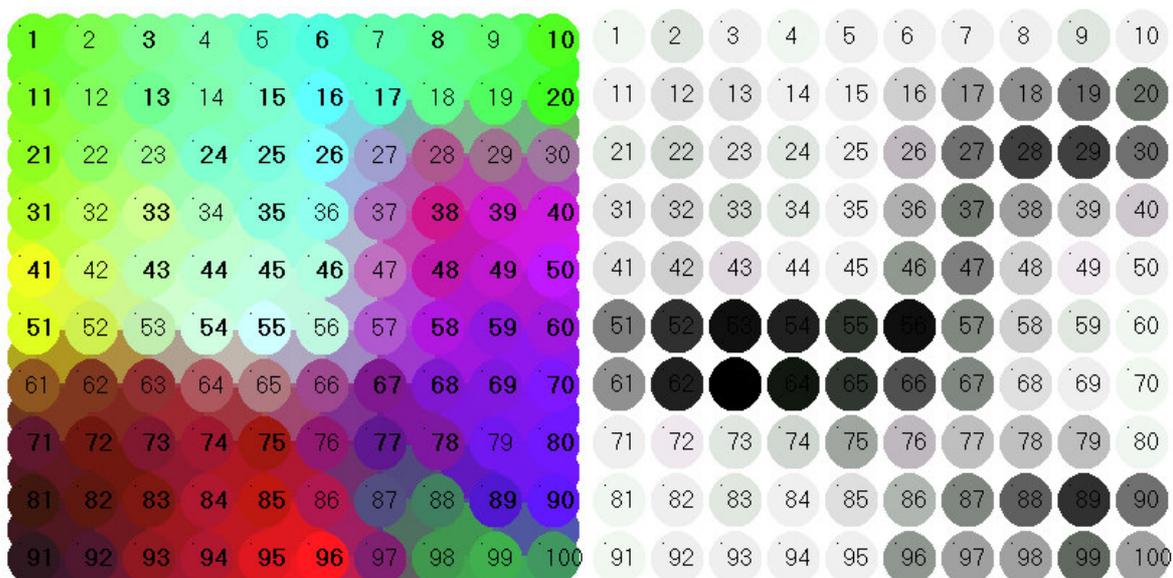


FIGURA 3.7 – EXEMPLO DE REPRESENTAÇÕES DE AGRUPAMENTOS PARA O PROBLEMA DE ALOCAÇÃO DE SALAS DE AULA ATRAVÉS DA U-MATRIZ

3.2.4.2 O mapa de densidade

Uma técnica baseada no princípio de densidade do mapa é apresentada no trabalho de Zhang e Li (1993). Para cada neurônio do mapa, calcula-se o número de padrões associados ao mesmo, criando-se um mapa de densidade. A determinação dos agrupamentos é feita utilizando-se os neurônios com número reduzido de padrões associados, encontrando-se as “fronteiras” topológicas entre os agrupamentos no mapa, conhecidos também como “vales” do mapa. Tais neurônios permitem a visualização e a criação dos agrupamentos da Rede de Kohonen utilizando-se linhas que unem estes neurônios.

Nas figuras 3.8A e 3.8B encontram-se representados a U-matriz e o mapa de densidade, respectivamente, de uma Rede Neural de Kohonen de dimensão 10×10 , com a formação de 8 agrupamentos determinados pelos neurônios com número reduzido de padrões de entrada associados. No mapa de densidade, encontram-se representados os números de padrões de entrada associados a cada neurônio do mapa.

Após a determinação dos “vales” no mapa de densidade, cada Agrupamento possui um neurônio mais representativo, isto é, um neurônio com maior número de padrões de entrada associados ao mesmo. Tal neurônio é denominado “pico” do Agrupamento, P_A .

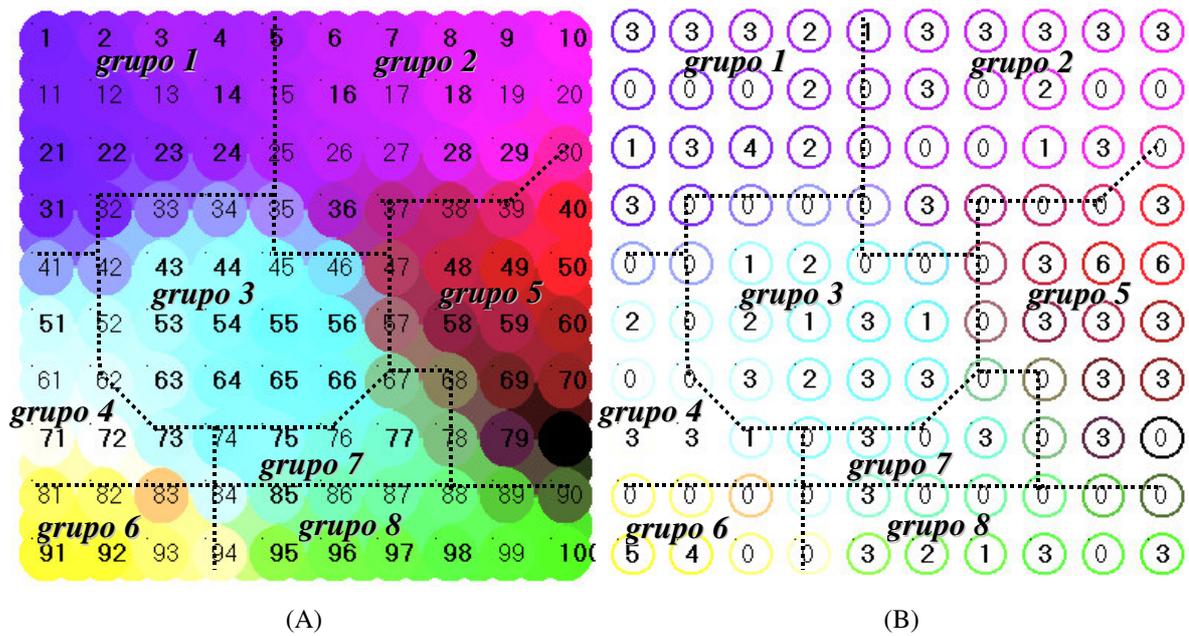


FIGURA 3.8 – EXEMPLOS DA REPRESENTAÇÃO DA U-MATRIZ (A) E DO MAPA DE DENSIDADE (B) PARA O PROBLEMA DE ALOCAÇÃO DE SALAS, COM FORMAÇÃO DE 8 AGRUPAMENTOS

A determinação dos neurônios pertencentes a cada agrupamento pode ser feita através do seguinte algoritmo (XU; LI, 2002):

Algoritmo para a criação de agrupamentos através de mapas de densidade

Passo 1: Encontre a densidade de cada neurônio do mapa M . O conjunto de neurônios com densidade não nula forma um conjunto P , com cardinalidade n_p . Classifique estes neurônios em ordem crescente, através da densidade dos mesmos. Vá ao Passo 2.

Passo 2: Determine a distância entre cada par de neurônios i e j do conjunto P :

$$dist(i, j) = \|w(i) - w(j)\|. \quad (3.2.19)$$

Se $dist(i, j) > E$ (onde E representa um erro usado para determinar a tolerância de distância entre as características de neurônios do mesmo agrupamento), então i e j são considerados membros de agrupamentos distintos. Repita este passo até concluir a comparação entre os elementos do conjunto P . Vá ao Passo 3.

Passo 3: O número de agrupamentos encontrados é n_A . Para cada agrupamento A , o neurônio com maior densidade é considerado centróide do agrupamento. Pare.

Com as técnicas descritas para o treinamento, a quantização de erros, e a visualização de agrupamentos, a construção de matrizes de custos para o problema da Designação pode ser feita através da expressão (3.2.16) ou da (3.2.17). A seguir, é apresentada a técnica proposta neste trabalho para resolver o problema da designação, utilizando matrizes construídas pela Rede de Kohonen.

3.3 PROPOSTA DE RESOLUÇÃO DO PROBLEMA DA DESIGNAÇÃO UTILIZANDO REDES NEURAIIS RECORRENTES

Nesta seção a Rede Neural Recorrente de Wang é apresentada para resolver o problema da Designação e, também a utilização de uma nova proposta para acelerar a convergência desta rede é definida.

A Rede Recorrente proposta por Wang (1992) converge para soluções infactíveis em problemas com múltiplas soluções ótimas globais ou soluções ótimas locais muito próximas da solução ótima global. A técnica proposta neste trabalho utiliza o princípio *Winner Takes All*, que além de acelerar a convergência de Rede de Wang resolve os problemas de infactibilidade da mesma (SIQUEIRA; SCHEER; STEINER, 2005).

3.3.1 A Rede Neural Recorrente de Wang para o problema da Designação

As Redes Neurais Recorrentes possuem memórias adaptativas, com processamento temporal, e estados que evoluem através de equações não lineares (VIEIRA; LEMOS; LEE, 2003). A estrutura da Rede Neural Recorrente utilizada neste trabalho para resolver o problema da Designação, baseia-se na forma matricial da formulação do problema da Designação, mostrada a seguir (HUNG; WANG, 2003):

$$\text{Minimizar } c = c^T x \quad (3.3.1)$$

$$\text{Sujeito a } Ax = b \quad (3.3.2)$$

$$x_{ij} \geq 0, \quad j = 1, 2, \dots, n \quad (3.3.3)$$

onde os vetores c^T e x contêm todas as linhas da matriz de custos c e da matriz com elementos de decisão x_{ij} , ou seja, $c^T = (c_{11}, c_{12}, \dots, c_{1n}, c_{21}, c_{22}, \dots, c_{2n}, \dots, c_{n1}, c_{n2}, \dots, c_{nm})$ e $x = (x_{11}, x_{12},$

..., x_{1n} , x_{21} , x_{22} , ..., x_{2n} , ..., x_{n1} , x_{n2} , ..., x_{nn}). O vetor b contém o número 1 em todas as posições, e a matriz A tem a seguinte forma:

$$A = \begin{bmatrix} I & I & \dots & I \\ B_1 & B_2 & \dots & B_n \end{bmatrix} \in \mathfrak{R}^{2n \times n^2}$$

onde I é a matriz identidade, de ordem $n \times n$, e cada matriz B_i , para $i = 1, 2, \dots, n$, contém zeros, com exceção da i -ésima linha que contém o número 1 em todas as posições. A figura 3.9 mostra a representação da estrutura da Rede Neural de Wang.

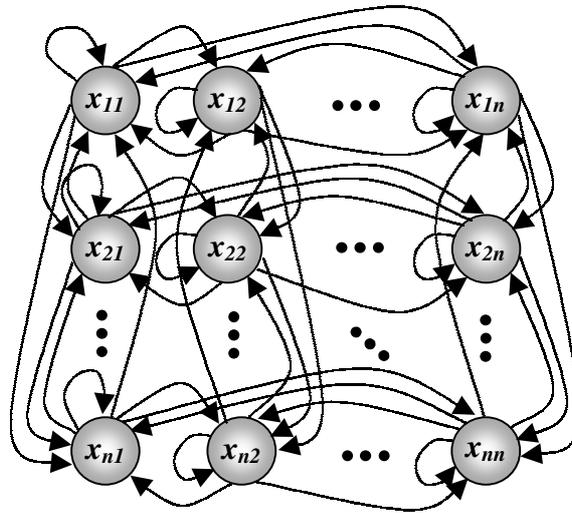


FIGURA 3.9 – REPRESENTAÇÃO DA ESTRUTURA DA REDE NEURAL RECORRENTE DE WANG

A Rede Neural Recorrente proposta por Wang (WANG, 1992, WANG, 1997; HUNG; WANG, 2003) é caracterizada pela seguinte equação diferencial:

$$\frac{du_{ij}(t)}{dt} = -\eta \sum_{k=1}^n x_{ik}(t) - \eta \sum_{l=1}^n x_{lj}(t) + \eta \theta_{ij} - \lambda c_{ij} e^{-\frac{t}{\tau}}, \quad (3.3.4)$$

$$x_{ij} = g(u_{ij}(t)), \quad (3.3.5)$$

onde o estado de equilíbrio desta Rede Neural é uma solução para o problema da Designação. O vetor limiar é definido como $\theta = A^T b = (2, 2, \dots, 2) \in \mathfrak{R}^{n^2 \times 1}$. Os valores dos elementos do vetor x são determinados por $g(u_{ij}(t))$ que pode ser a função sigmoideal, ou a função limiar em rampa (FAUSETT, 1994). Neste trabalho, optou-se a utilização da função sigmoideal para a determinação do vetor x .

Os parâmetros η , λ e τ são constantes, e escolhidos empiricamente (HUNG e WANG, 2003), afetando a convergência da rede. O parâmetro η serve para penalizar violações do conjunto de restrições do problema definido por (3.2.1)-(3.2.4). Os parâmetros λ e τ controlam a minimização da função objetivo do problema da Designação (3.2.1).

A forma matricial para esta rede é deduzida facilmente, onde as expressões (3.3.4) e (3.3.5) podem ser escritas da seguinte forma:

$$\frac{du(t)}{dt} = -\eta(Wx(t) - \theta) - \lambda ce^{-\frac{t}{\tau}}, \quad (3.3.6)$$

$$x = g(u(t)), \quad (3.3.7)$$

onde $W = A^T A$. As propriedades de convergência desta rede encontram-se demonstradas nas seguintes publicações: Wang (1993a, 1993b, 1994, 1997); Xia, Wang e Hung (1999), Liu, Hu e Wang (2004) e Hung e Wang (2003).

3.3.2 Múltiplas soluções ótimas globais e soluções ótimas locais próximas da solução ótima global

Em algumas matrizes de custos, as soluções ótimas estão muito próximas, ou então várias soluções ótimas são admissíveis. Nestes casos, a Rede Neural proposta por Wang, apresentada na seção 3.3.1, não converge para uma solução factível, como pode ser visto no exemplo dado a seguir. Seja a matriz de custos c dada abaixo:

$$c = \begin{pmatrix} 1,4 & 6,1 & 3,1 & 0,4 & 2,2 & 4,4 & 0,1 & 0 \\ 1,3 & 0,2 & 3,3 & 0,2 & 1,2 & 0,4 & 1,5 & 8,2 \\ 1,7 & 2,9 & 2,8 & 0,1 & 9,8 & 0,4 & 4,2 & 6,8 \\ 0,5 & 0 & 0 & 1 & 1,1 & 0,9 & 2,4 & 0,9 \\ 2,8 & 7,7 & 1,2 & 0,5 & 0,9 & 6 & 4,6 & 5,9 \\ 0,5 & 0,2 & 0 & 1,8 & 8,5 & 4,9 & 4,4 & 0,9 \\ 0,6 & 0,7 & 0,7 & 6,9 & 0,1 & 0 & 3,2 & 3,8 \\ 1,4 & 5,4 & 3,7 & 1,1 & 3 & 0 & 1,3 & 0,7 \end{pmatrix}, \quad (3.3.8)$$

após 6.300 iterações utilizando a Rede Neural de Wang apresenta a seguinte solução:

$$x^* = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0,5 & 0 & 0,5 & 0 & 0 \\ 0,5 & 0 & 0,5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0,5 & 0,5 & 0 & 0 & 0 \\ 0,5 & 0 & 0,5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,5 & 0,5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Esta solução não é factível, pois alguns elementos x_{ij} violam o conjunto de restrições (3.2.4), mostrando que a Rede Neural de Wang necessita de ajustes para estes casos. Duas soluções ótimas deste problema são as seguintes:

$$\tilde{x}^* = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad \bar{x}^* = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

A simples decisão de colocar o valor unitário em qualquer um dos elementos x_{ij} que possui valor 0,5 na solução x^* pode torná-la infactível ou determinar uma solução ótima local.

Em um outro exemplo, a matriz de custos \bar{c} é dada por:

$$\bar{c} = \begin{pmatrix} 0,6 & 0,9 & 1 & 2,7 & 0 & 1 & 0,1 & 0 \\ 1 & 0 & 0,3 & 0,4 & 5,5 & 0,3 & 3,4 & 5,5 \\ 0 & 2,2 & 0,2 & 0,6 & 0,2 & 0,2 & 0 & 0,2 \\ 0,3 & 0,2 & 0 & 0,1 & 1,5 & 0 & 0,1 & 1,5 \\ 0,6 & 0,9 & 1 & 2,7 & 0 & 1 & 0,1 & 0 \\ 0 & 2,2 & 0,2 & 0,6 & 0,2 & 0,2 & 0 & 0,2 \\ 0,6 & 0,9 & 1 & 2,7 & 0 & 1 & 0,1 & 0 \\ 0 & 2,2 & 0,2 & 0,6 & 0,2 & 0,2 & 0 & 0,2 \end{pmatrix},$$

cuja solução encontrada pela Rede Neural de Wang, após 4.715 iterações é:

$$x^* = \begin{pmatrix} 0 & 0 & 0 & 0 & 0,33 & 0 & 0,33 & 0,33 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0,5 & 0 & 0,25 & 0 & 0 & 0,25 & 0 & 0 \\ 0 & 0 & 0,5 & 0 & 0 & 0,5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,33 & 0 & 0,33 & 0,33 \\ 0,5 & 0 & 0,25 & 0 & 0 & 0,25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,33 & 0 & 0,33 & 0,33 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

e a solução ótima global para a matriz \bar{c} é:

$$\bar{x}^* = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}.$$

Um ajuste possível na solução x^* pode ser feito da seguinte forma: considerando a solução x^* , escolhe-se o maior elemento da linha i , x_{ij}^* ; atribui-se o valor 1 a este elemento, e aos demais elementos da linha i e da coluna j , atribui-se o valor nulo. Desta forma, a solução x^* torna-se:

$$x^* = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

ou seja, uma solução ótima local é encontrada. A linha 8 e a coluna 4 não possuem elementos não nulos para efetuar a designação, e como o custo $c_{84} = 0,6$, esta designação pode ser feita, resultando na seguinte solução:

$$x^* = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix},$$

cujo custo total é igual a 0,9, enquanto que a solução ótima global tem custo total igual a 0,6. Em matrizes com dimensão bem maior do que a matriz considerada neste exemplo, o número de linhas e colunas sem designação pode ser maior, o que torna difícil a decisão sobre quais designações devem ser feitas.

Outro ajuste que pode ser feito na solução x^* é a modificação dos elementos da matriz de custos, eliminando empates nos custos correspondentes das variáveis x_{ij} que possuem solução não-unitária. Desta forma, pode-se encontrar uma solução ótima local quando as modificações não são feitas de forma adequada.

Portanto, estas decisões podem acarretar em resultados não satisfatórios, além de aumentar consideravelmente o tempo de processamento da Rede Neural. Com o objetivo de resolver este problema, uma técnica que faz um “ajuste” neste problema é mostrada a seguir, permitindo o alcance da solução ótima global em mais de 60% dos casos testados e, ainda, com convergência bem mais rápida se comparada à Rede Neural de Wang sem a utilização desta técnica.

3.3.3 Uso do princípio *Winner Takes All* aplicado à Rede Neural Recorrente de Wang

O método proposto neste trabalho utiliza uma técnica baseada no princípio *Winner Takes All*, acelerando a convergência da Rede Neural de Wang, além de corrigir eventuais problemas que podem surgir devido a múltiplas soluções ótimas ou soluções ótimas muito próximas (STEINER; SIQUEIRA; SCHEER, 2005).

O termo da equação (3.3.6), $Wx(t) - \theta$, mede a violação das restrições do problema da Designação. Após um certo número de iterações, este termo não sofre mudanças substanciais em seu valor, evidenciando o fato de que as restrições do problema estão quase satisfeitas.

Neste momento, o método proposto nesta seção pode ser aplicado. Quando o valor deste termo é nulo, as restrições do problema da Designação estão satisfeitas.

O valor máximo do vetor $\varepsilon = |Wx(t) - \theta|$ é 2, pois a solução inicial da Rede de Wang é inactível, determinada pelo vetor nulo x . Após algumas iterações, os valores dos elementos de x são modificados para que as restrições do problema sejam satisfeitas, e quando o valor de $|Wx(t) - \theta|$ for próximo de zero, em todas as coordenadas, as restrições do problema da Designação estão satisfeitas.

Seja δ um vetor utilizado para determinar o momento em que a técnica *Winner Takes All* deve ser aplicada. Desta maneira, quando todos os elementos de x satisfazem a condição $|Wx(t) - \theta| \leq \delta$, onde $\delta_i \in [0, 2]$, a técnica proposta pode ser usada (SIQUEIRA; SCHEER; STEINER, 2005). Vale lembrar que as melhores soluções apresentadas pela técnica *Winner Takes All* são encontradas quando os valores δ_i são mais próximos de zero, ou seja, valores altos para o vetor δ_i garantem convergência precoce da rede, com soluções factíveis de qualidade inferior.

O algoritmo da técnica proposta *Winner Takes All* é apresentado a seguir.

Algoritmo baseado no princípio *Winner Takes All* para a aplicação da Rede Neural de Wang para o problema da Designação

Passo 1: Encontre uma solução x para o problema da Designação, utilizando-se a Rede Neural de Wang. Se $|Wx(t) - \theta| \leq \delta$, então vá ao Passo 2. Caso contrário, encontre outra solução x .

Passo 2: Considere a matriz de decisão x após um certo número de iterações da Rede Neural de Wang. Seja a matriz \bar{x} , onde $\bar{x} = x$ e $m = 1$. Vá ao passo 3.

Passo 3: Encontre o m -ésimo maior elemento da matriz de decisão, \bar{x}_{kl} . O valor deste elemento é substituído pela metade da soma de todos os valores dos elementos da linha k e da coluna l da matriz x , ou seja,

$$\bar{x}_{kl} = \frac{1}{2} \left(\sum_{i=1}^n x_{il} + \sum_{j=1}^n x_{kj} \right). \quad (3.3.9)$$

Os demais elementos da linha k e coluna l tornam-se nulos na matriz \bar{x} . Vá ao passo 4.

Passo 4: Se $m \leq n$, então faça $m = m + 1$, e vá ao Passo 3. Caso contrário, vá ao Passo 5.

Passo 5: Se uma boa aproximação para uma solução factível do problema da Designação for encontrada, então pare. Caso contrário, faça $x = \bar{x}$, execute a Rede Neural de Wang novamente, e vá ao Passo 2.

Uma boa aproximação para uma solução do problema da Designação consiste em valores da matriz x que possuem elementos próximos dos valores 0 ou 1. Um erro de tolerância pode ser definido no início da execução do algoritmo proposto, com o objetivo de medir os valores das soluções encontradas e determinar se são boas aproximações para soluções do problema da Designação.

Um exemplo do funcionamento do algoritmo proposto nesta seção, é apresentado no exemplo ilustrativo dado a seguir.

3.3.3.1 Exemplo Ilustrativo

Considerando-se a matriz a seguir, que consiste em uma solução parcial do problema definido pela matriz c em (3.3.8), após 14 iterações, o maior elemento da matriz \bar{x} está na linha 1, coluna 7, ou seja, $k = 1$ e $l = 7$:

$$\bar{x} = \begin{pmatrix} 0,0808 & 0,0011 & 0,0168 & 0,1083 & 0,0514 & 0,0033 & *0,422 & 0,3551 \\ 0,0056 & 0,2827 & 0,0168 & 0,1525 & 0,1484 & 0,1648 & 0,0866 & 0 \\ 0,1754 & 0,0709 & 0,0688 & 0,3449 & 0 & 0,3438 & 0,0425 & 0,0024 \\ 0,1456 & 0,2412 & 0,2184 & 0,0521 & 0,1131 & 0,0747 & 0,0598 & 0,1571 \\ 0,0711 & 0 & 0,2674 & 0,272 & 0,3931 & 0,0024 & 0,0306 & 0,0061 \\ 0,2037 & 0,2823 & 0,2956 & 0,0366 & 0 & 0,0025 & 0,0136 & 0,2186 \\ 0,1681 & 0,174 & 0,1562 & 0 & 0,3053 & 0,2016 & 0,0369 & 0,0144 \\ 0,1142 & 0,0031 & 0,0138 & 0,0829 & 0,0353 & 0,2592 & 0,251 & 0,2907 \end{pmatrix}.$$

Após atualizar o elemento x_{17} através da expressão (3.3.9), o resultado dado abaixo é encontrado. Encontra-se o segundo maior elemento de \bar{x} , que está na linha 5 e coluna 5, $k = l = 5$:

$$\bar{x} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1,0412 & 0 \\ 0,0056 & 0,2827 & 0,0168 & 0,1525 & 0,1484 & 0,1648 & 0 & 0 \\ 0,1754 & 0,0709 & 0,0688 & 0,3449 & 0 & 0,3438 & 0 & 0,0024 \\ 0,1456 & 0,2412 & 0,2184 & 0,0521 & 0,1131 & 0,0747 & 0 & 0,1571 \\ 0,0711 & 0 & 0,2674 & 0,272 & *0,393 & 0,0024 & 0 & 0,0061 \\ 0,2037 & 0,2823 & 0,2956 & 0,0366 & 0 & 0,0025 & 0 & 0,2186 \\ 0,1681 & 0,174 & 0,1562 & 0 & 0,3053 & 0,2016 & 0 & 0,0144 \\ 0,1142 & 0,0031 & 0,0138 & 0,0829 & 0,0353 & 0,2592 & 0 & 0,2907 \end{pmatrix}.$$

Após a atualização do elemento x_{55} , prossegue-se com a atualização do terceiro maior elemento da matriz \bar{x} , que está na linha 3 e coluna 4, $k = 3$ e $l = 4$.

$$\bar{x} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1,0412 & 0 \\ 0,0056 & 0,2827 & 0,0168 & 0,1525 & 0 & 0,1648 & 0 & 0 \\ 0,1754 & 0,0709 & 0,0688 & *0,345 & 0 & 0,3438 & 0 & 0,0024 \\ 0,1456 & 0,2412 & 0,2184 & 0,0521 & 0 & 0,0747 & 0 & 0,1571 \\ 0 & 0 & 0 & 0 & 1,0446 & 0 & 0 & 0 \\ 0,2037 & 0,2823 & 0,2956 & 0,0366 & 0 & 0,0025 & 0 & 0,2186 \\ 0,1681 & 0,174 & 0,1562 & 0 & 0 & 0,2016 & 0 & 0,0144 \\ 0,1142 & 0,0031 & 0,0138 & 0,0829 & 0 & 0,2592 & 0 & 0,2907 \end{pmatrix}$$

Após atualizar todos os elementos de \bar{x} , obtém-se a seguinte solução:

$$\bar{x} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1,0412 & 0 \\ 0 & 1,0564 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1,0491 & 0 & 0 & 0 & 0 \\ 1,0632 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1,0446 & 0 & 0 & 0 \\ 0 & 0 & 1,0533 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,0544 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1,0473 \end{pmatrix}.$$

Esta solução \bar{x} é, então, apresentada novamente à Rede Neural de Wang, e depois de encontrar outra solução x , calcula-se uma nova solução \bar{x} . Este procedimento é feito até que uma boa aproximação de uma solução factível para o problema da Designação seja encontrada. Neste exemplo, após 5 iterações do algoritmo *Winner Takes All*, a matriz \bar{x} apresenta uma aproximação para uma solução ótima do problema:

$$\bar{x} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0,9992 & 0 \\ 0 & 0,9996 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0,9994 & 0 & 0 & 0 & 0 \\ 0,999 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,9985 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1,0003 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,9991 \end{pmatrix}.$$

Dois importantes aspectos desta técnica que devem ser observados são os seguintes: o reduzido número de iterações necessárias para encontrar uma solução factível para o problema da Designação (aproximadamente 1% do número necessário para a convergência da Rede Neural original de Wang) e, além disso, a ausência de problemas com relação às matrizes com múltiplas soluções ótimas.

Os ajustes dos parâmetros da Rede Recorrente de Wang são imprescindíveis para garantir a convergência desta técnica, sendo que algumas formas de ajustá-los são apresentadas a seguir.

3.3.4 Algumas metodologias para ajustar os parâmetros da Rede Neural de Wang

Neste trabalho, os parâmetros utilizados desempenham papel fundamental para a convergência da Rede Neural Recorrente de Wang. Em todas as matrizes testadas, considerou-se $\eta = 1$, e os parâmetros τ e λ foram calculados de diversas maneiras (SIQUEIRA; SCHEER; STEINER, 2005), descritas a seguir.

Uma das formas mais usuais de calcular o parâmetro λ para o problema da Designação pode ser encontrada em Wang (1992), onde λ é dado por:

$$\lambda = \eta / C_{\max}, \quad (3.3.10)$$

onde $C_{\max} = \max\{c_{ij}; i, j = 1, 2, \dots, n\}$. Desta forma, o último termo da expressão (3.3.8) possui valor máximo η para o maior custo da matriz c , auxiliando o processo de minimização da função objetivo do problema da Designação.

A utilização de medidas de dispersão entre os elementos da matriz c mostrou-se eficiente para ajustar os parâmetros τ e λ em matrizes com grande dispersão entre seus custos,

de acordo com os resultados apresentados no Capítulo V. Considerando-se σ como o desvio padrão entre os elementos da matriz de custos c , o parâmetro λ pode ser definido como:

$$\lambda = \frac{\eta}{\sigma} \quad (3.3.11)$$

Outra maneira de ajustar λ é considerá-lo um vetor, definido por:

$$\bar{\lambda} = \eta \left(\frac{1}{\sigma_1}, \frac{1}{\sigma_2}, \dots, \frac{1}{\sigma_n} \right), \quad (3.3.12)$$

onde σ_i , para $i = 1, 2, \dots, n$, representa o desvio padrão de cada linha da matriz de custos c . Cada elemento do vetor $\bar{\lambda}$ é utilizado para atualizar a linha correspondente da matriz de decisão x . Esta forma de calcular λ mostrou-se muito eficiente em matrizes de custos com grande dispersão entre seus valores, conforme mostram os resultados apresentados no Capítulo V deste trabalho. Desta forma, as matrizes com linhas que possuem os valores dos custos com desvio padrão muito grande, possuem valores diferenciados neste parâmetro, penalizando as linhas com grande dispersão entre os valores de seus custos.

Uma variação da expressão (3.3.10), que utiliza o mesmo princípio da expressão (3.3.12), é definir λ como o vetor dado por:

$$\bar{\lambda} = \eta \left(\frac{1}{c_{1 \max}}, \frac{1}{c_{2 \max}}, \dots, \frac{1}{c_{n \max}} \right), \quad (3.3.13)$$

onde $c_{i \max} = \max\{c_{ij}; j = 1, 2, \dots, n\}$. Esta definição para λ também produz bons resultados em matrizes com grande dispersão entre seus elementos, pois neste tipo de matriz os custos máximos de cada linha possuem grandes diferenças, e esta definição de λ cria determinadas faixas de atuação para este parâmetro, facilitando o alcance da otimalidade, pois o nível de penalidade para os custos máximos de cada linha da matriz de custos será similar. Esta mesma justificativa serve para a definição de λ em (3.3.12).

O parâmetro τ depende do número necessário de iterações para a convergência da Rede Neural. Quando a técnica de ajuste *Winner Takes All*, apresentada na seção anterior, não é utilizada, o número necessário de iterações para a convergência da Rede Neural de Wang varia entre 1.000 e 15.000 iterações (WANG, 1997). Neste caso, τ é uma constante, tal que:

$$1.000 \leq \tau \leq 15.000. \quad (3.3.14)$$

Quando o ajuste *Winner Takes All* é utilizado, o número necessário de iterações varia entre 5 e 300. Portanto, o valor de τ é considerado no intervalo:

$$5 \leq \tau \leq 300. \quad (3.3.15)$$

Neste trabalho, foram utilizadas outras duas formas de ajuste do parâmetro τ , além de considerá-lo constante, nos intervalos mostrados nas expressões (3.3.14) e (3.3.15). Em uma das técnicas, τ é dado por:

$$\bar{\tau} = \frac{1}{\mu} (\mu_1 \sigma_1, \mu_2 \sigma_2, \dots, \mu_n \sigma_n), \quad (3.3.16)$$

onde μ_i é a média dos elementos da i -ésima linha da matriz de custos c , σ_i é o desvio padrão dos elementos da i -ésima linha de c , e μ é a média entre os valores de todos os elementos da matriz c . Esta definição utiliza a razão entre as médias de cada linha da matriz e da matriz toda e o desvio padrão de cada linha da matriz, mostrando-se eficiente para definir τ em matrizes com grande dispersão entre seus elementos.

A segunda proposta de ajuste para τ utiliza a definição da Rede Neural de Wang. Quando $c_{ij} = c_{\max}$ e $Wx(t) - \theta \cong 0$, o termo $-\lambda_i c_{ij} \exp(-t/\tau_i) = k_i$ deve ser tal que, $g(k_i) \cong 0$. Logo, a função sigmoïdal g pode ser aplicada aos termos k_i , com o objetivo de encontrar um limite para o parâmetro τ_i :

$$g(k_i) = \frac{1}{1 + e^{-\beta k_i}} = \phi_i,$$

onde ϕ é um vetor com valores próximos de zero. Isolando-se o valor de k_i , obtém-se:

$$k_i = \frac{-\ln\left(\frac{1}{\phi_i} - 1\right)}{\beta}. \quad (3.3.17)$$

Utilizando-se o valor de $k_i = \lambda_i c_{ij} \exp(-t/\tau_i)$ da equação (3.3.17), e isolando-se o valor do parâmetro τ_i , obtém-se:

$$\tau_i = \frac{-t}{\ln\left(\frac{-k_i}{\lambda_i c_{\max}}\right)}. \quad (3.3.18)$$

As aplicações dos parâmetros descritos nesta seção (3.3.10)-(3.3.18), e os resultados encontrados com tais parâmetros são apresentados no Capítulo V a seguir.

No capítulo IV um caso de estudo é apresentado, no qual a metodologia proposta neste trabalho foi implementada. Os resultados desta metodologia são apresentados no Capítulo V deste trabalho. No Capítulo VI, são apresentados os resultados da aplicação da Rede Neural de Wang com o princípio *Winner Takes All* para o problema do Caixeiro Viajante.

CAPÍTULO IV

4 ESTUDO DE CASO

4.1 DESCRIÇÃO DO PROBLEMA

A metodologia proposta no Capítulo III deste trabalho é implementada neste capítulo na resolução de um problema real, com dimensão considerável e com a obtenção de resultados considerados satisfatórios. O problema em questão consiste em alocar disciplinas de graduação e pós-graduação da Universidade Federal do Paraná nas salas de aula disponíveis de um dos campus da Universidade, denominado Centro Politécnico.

Até o ano 2000, esta alocação de salas era feita manualmente, exigindo um tempo considerável dos membros da Comissão responsável por esta tarefa. No primeiro semestre do ano 2000, a alocação de salas de um dos blocos do Centro Politécnico foi feita utilizando-se o modelo da Designação, com a resolução através do método Húngaro. Os resultados foram satisfatórios, e a implementação desta metodologia de resolução para os demais blocos ocorreu no ano seguinte.

Com a criação de novos cursos na UFPR nos últimos anos, a demanda por salas de aula e, conseqüentemente, a dimensão do problema aumentou consideravelmente.

O grafo bi-partido utilizado para a modelagem deste problema possui os grupos de vértices definidos como: V' - conjunto de disciplinas que necessitam ser alocadas em um determinado horário h_i ; e V'' - conjunto de salas de aula disponíveis no horário h_i . O cálculo do custo da designação de uma disciplina d_i para uma sala s_j necessita da verificação da disponibilidade da sala s_j desde o horário de início até o término da aula da disciplina d_i ; se a sala s_j não está disponível em todos os horários verificados para d_i , o custo para esta designação é considerado ∞ :

$$c_{ij} = \infty. \quad (4.1.1)$$

As disciplinas do departamento de Desenho, e algumas disciplinas de cursos como Arquitetura, Geografia, Engenharia Civil e Engenharia Cartográfica, necessitam de carteiras de desenho, e não podem ser alocadas em salas com carteiras de braço. A designação de uma disciplina deste tipo para salas de carteira de braço tem custo ∞ .

Outra restrição considerada neste problema é a escolha dos blocos de salas de aula pelos cursos. Cada curso tem a preferência para alocação de suas disciplinas em um determinado bloco, e estas designações devem ter um custo reduzido em relação às designações destas disciplinas para salas semelhantes dos demais blocos. A utilização da Rede de Kohonen para fazer os agrupamentos faz esta separação dos blocos, e tais custos são reduzidos para estes casos.

Existe a escolha dos blocos em que um determinado curso não deve ser alocado, e nestes casos, os custos para tais designações são representados por ∞ .

Quando a capacidade da sala de aula i é menor do que o número de alunos matriculados na disciplina j , tem-se o custo $c_{ij} = \infty$.

Após a criação da matriz de custos para a Alocação de salas de aula para todos os horários da semana, o modelo da designação é aplicado para decidir as alocações das salas de todas as disciplinas consideradas.

Sejam n_T o número total de disciplinas, n_i o número de disciplinas com horário de início h , s_T o número total de salas de aula e s_i o número de salas de aula disponíveis no horário h . O algoritmo básico para alocar as n_T disciplinas nas s_T salas de aula é dado a seguir:

Algoritmo para Alocação de salas para disciplinas

- Passo 1: Determine os custos das designações (utilizando Rede de Kohonen ou as distâncias Euclidianas entre os vetores de características) entre as s_T salas e as n_T disciplinas. Vá ao Passo 2.
- Passo 2: Se todos os horários já foram executados, então pare. Caso contrário, escolha um determinado horário h ainda não executado. Selecione as salas disponíveis neste horário (s_h), e as disciplinas que necessitam ser alocadas (d_h) e que possuem horário de início h . Vá ao Passo 3.

Passo 3: Aplique o modelo da Designação (método Húngaro, *Matching*, ou Rede Neural de Wang) para alocar as d_h disciplinas nas s_h salas. Atualize a matriz de custos, colocando custo ∞ para as salas utilizadas para a alocação das d_h disciplinas do horário h até o horário do final da aula de cada disciplina alocada. Vá ao Passo 2.

Nas próximas seções são apresentadas as adaptações utilizadas na metodologia proposta neste trabalho para resolver o problema de Alocação de salas para disciplinas.

4.2 APLICAÇÃO DA REDE DE KOHONEN PARA CRIAR A MATRIZ DE CUSTOS DO PROBLEMA

Na primeira fase da alocação de salas para disciplinas, as matrizes de custos são definidas. A proposta deste trabalho é utilizar a Rede de Kohonen para criar os agrupamentos dos conjuntos de salas (M') e de disciplinas (M''), e calcular os custos destas designações através da expressão (3.2.16) ou (3.2.17).

4.2.1 Dados de entrada

As informações utilizadas para a construção da matriz de custos do problema são:

- 1) para cada disciplina:
 - a) o número de alunos matriculados;
 - b) o tipo de carteiras necessário para a disciplina;
 - c) o bloco de preferência do curso;
 - d) o bloco de proibição da alocação da disciplina.
- 2) para cada sala de aula:
 - a) a capacidade da sala;
 - b) o tipo de carteiras existente na sala;
 - c) o bloco onde localiza-se a sala.

Para a criação dos mapas Auto-Organizáveis M' e M'' (para as disciplinas e salas, respectivamente), as informações utilizadas são as três primeiras listadas acima, tanto para as salas quanto para as disciplinas. A última condição verificada na criação da matriz de custos é a designação de uma disciplina para uma sala localizada em um bloco de proibição. O custo

da designação de uma disciplina em um bloco de proibição possui o valor alterado para ∞ , e, em caso contrário, este custo permanece inalterado.

Os vetores dos padrões de entrada têm a seguinte característica para as disciplinas:

$$p(\text{número de alunos, tipo de carteira, bloco de preferência}), \quad (4.1.2)$$

e para as salas de aula:

$$p(\text{capacidade, tipo de carteira, bloco}). \quad (4.1.3)$$

Para o treinamento da Rede de Kohonen foi utilizada a metodologia *holdout* (WITTEN; FRANK, 2000), criando-se um arquivo com 2/3 das disciplinas consideradas para este problema, escolhidas aleatoriamente. O banco de dados utilizado neste trabalho é o grupo de disciplinas do primeiro semestre do ano letivo de 2005. Existem 650 disciplinas neste banco de dados, e são utilizadas as informações de 433 disciplinas (equivalente a 2/3 das 650 disciplinas) na fase de treinamento da rede. As disciplinas restantes representam o conjunto de testes da rede.

Com o objetivo de evitar a prioridade da alocação de salas para os dados de entrada com valores muito altos, os padrões de entrada utilizados são vetores com valores entre 0 e 1 para as três coordenadas (JAIN; MURTY; FLYNN, 1999). A padronização dos dados para as capacidades foi feita dividindo-se a primeira coordenada de todos os vetores pela capacidade da maior sala entre as disponíveis, que neste caso é de 115 carteiras. Desta forma, a primeira coordenada de todos os vetores de entrada possui valores entre 0 e 1.

A segunda característica utilizada para definir os agrupamentos da Rede de Kohonen é o tipo de carteira da sala. Quando uma disciplina necessita uma sala para desenho, a segunda coordenada do padrão de entrada possui valor 1, e em caso contrário, valor nulo. Desta forma, o custo da designação de salas sem carteira de braço é menor para as disciplinas que possuem esta necessidade, enquanto que o custo para a designação das disciplinas que não precisam de salas especiais para as salas com carteira de braço também é reduzido, forçando estas designações. Como é impossível utilizar as salas com carteiras de braço para disciplinas de desenho, após o cálculo de todos os custos, as designações das disciplinas de desenho para as salas com carteiras de braço são proibidas, utilizando-se custo ∞ .

A terceira característica utilizada para a criação dos agrupamentos é a escolha dos blocos didáticos de preferência para cada disciplina. Esta escolha é feita atribuindo-se à

terceira coordenada do vetor de entrada de dados valores entre 0 e 1, de acordo com a localização de cada bloco de salas de aula. Na figura 4.1, encontra-se representada a planta dos prédios que possuem salas de aula utilizadas neste trabalho, onde os blocos utilizados estão destacados e os respectivos nomes destes blocos são: PA, CT, PC, PD, PE, PF, PG e PH.

As disciplinas dos três setores que utilizam estes blocos são alocadas em blocos mais próximos das coordenações de seus respectivos cursos. Os blocos mais próximos das coordenações dos cursos dos setores de Ciências Exatas e de Ciências da Terra são PA, CT e PC e do Setor de Tecnologia são os blocos PD, PE, PF, PG e PH.



FIGURA 4.1 – REPRESENTAÇÃO DA VISTA SUPERIOR DOS PRÉDIOS QUE POSSUEM AS SALAS DE AULA UTILIZADAS.

FONTE: DEPARTAMENTO DE GEOMÁTICA – SETOR DE CIÊNCIAS DA TERRA - UFPR

O critério utilizado para determinar valores para a terceira coordenada de cada vetor de entrada foi a distância entre os blocos didáticos disponíveis. A matriz das distâncias entre os blocos, em metros, está colocada na Tabela 4.1.

TABELA 4.1 – DISTÂNCIAS ENTRE OS BLOCOS DIDÁTICOS DO CENTRO POLITÉCNICO – UFPR

	PA	CT	PC	PD	PE	PF	PG	PH
PA	0	76,5	27,2	100,3	127,5	154,7	181,9	209,1
CT	76,5	0	44,2	98,6	125,8	153	180,2	207,4
PC	27,2	44,2	0	107,1	134,3	161,5	188,7	215,9
PD	100,3	98,6	107,1	0	27,2	54,4	81,6	108,8
PE	127,5	125,8	134,3	27,2	0	27,2	54,4	81,6
PF	154,7	153	161,5	54,4	27,2	0	27,2	54,4
PG	181,9	180,2	188,7	81,6	54,4	27,2	0	27,2
PH	209,1	207,4	215,9	108,8	81,6	54,4	27,2	0

FONTE: DEPARTAMENTO DE GEOMÁTICA – SETOR DE CIÊNCIAS DA TERRA - UFPR

Os blocos didáticos mais distantes entre si são os blocos PC e PH, com distância $d_{\max} = 215,9\text{m}$. Utilizando-se estes blocos extremos como referência, com a atribuição do valor 0 para o bloco PC e 1 para o bloco PH, o cálculo dos coeficientes correspondentes para cada bloco pode ser feito utilizando-se a seguinte expressão:

$$coef_{bloco} = \frac{d(bloco, PC)}{d_{\max}}, \quad (4.1.4)$$

e o resultado da representação de cada bloco através da expressão (4.1.4) é apresentada na Figura 4.2.



FIGURA 4.2 – REPRESENTAÇÃO DA LOCALIZAÇÃO DE CADA BLOCO DIDÁTICO, UTILIZADA COMO TERCEIRA COORDENADA DOS PADRÕES DE ENTRADA DA REDE

A aplicação dos mapas Auto-Organizáveis de Kohonen para dados reais do problema de alocação de salas para disciplinas é mostrada no Capítulo V, onde os padrões de entrada foram utilizados nos formatos definidos nesta seção, e a matriz de custos para o problema da Designação é construída utilizando-se a metodologia proposta no Capítulo III. A resolução do problema da Designação das salas disponíveis para as disciplinas é feita através da Rede Neural de Wang com o princípio *Winner Takes All*, e os resultados desta resolução são discutidos no Capítulo V também.

CAPÍTULO V

5 ANÁLISES DE RESULTADOS DA METODOLOGIA PROPOSTA PARA O PROBLEMA DA DESIGNAÇÃO LINEAR

5.1 INTRODUÇÃO

Neste capítulo encontram-se os resultados da aplicação das técnicas propostas no Capítulo III deste trabalho ao problema descrito no Capítulo IV, assim como alguns testes sobre a aplicação da técnica *Winner Takes All* na Rede Neural Recorrente de Wang para o problema da Designação.

Na próxima seção são mostrados os resultados da técnica *Winner Takes All* na Rede Neural Recorrente de Wang em matrizes com custos aleatórios para o problema da Designação.

5.2 RESULTADOS DA APLICAÇÃO DA TÉCNICA *WINNER TAKES ALL* EM MATRIZES COM CUSTOS ALEATÓRIOS PARA O PROBLEMA DA DESIGNAÇÃO

Neste trabalho, 73 matrizes (com dimensões variando de 3×3 até 20×20) foram utilizadas para testar as técnicas de ajustes de parâmetros apresentadas na seção 3.6 deste trabalho, além da proposta de ajuste *Winner Takes All* aplicada à Rede Neural de Wang para o problema da Designação. Estas matrizes foram geradas aleatoriamente, com alguns casos de múltiplas soluções ótimas e de soluções ótimas muito próximas.

Na tabela 5.1, encontram-se as médias dos resultados para as 32 matrizes testadas que possuem único ótimo global. O erro médio utilizado para comparação dos resultados obtidos neste trabalho em todas as técnicas é dado pela seguinte expressão:

$$\text{erro médio} = 100 \times \left(\frac{\text{solução obtida} - \text{solução ótima}}{\text{solução ótima}} \right). \quad (5.1.1)$$

Para ajustar λ , as expressões utilizadas foram as seguintes expressões: (3.3.10) na primeira e na última coluna da Tabela 5.1; (3.3.11) na segunda e quarta colunas; (3.3.13) na terceira coluna; (3.3.12) na quinta coluna. Para calcular τ , utilizou-se as seguintes expressões: (3.3.15) nas três primeiras colunas; (3.3.16) na quarta coluna; (3.3.18) na quinta coluna; e (3.3.14) na última coluna. Os resultados da aplicação da Rede Neural de Wang, sem a utilização do ajuste proposto neste trabalho, encontram-se na última (sexta) coluna desta tabela.

TABELA 5.1 – RESULTADOS PARA AS 32 MATRIZES COM ÚNICO ÓTIMO GLOBAL

λ	$\lambda = \eta/C_{\max}$	$\lambda = \eta/\sigma$	$\lambda_i = \eta/c_{i \max}$	$\lambda = \eta/\sigma$	$\lambda_i = \eta/\sigma_i$	$\lambda = \eta/C_{\max}$
τ	$5 \leq \tau \leq 300$	$5 \leq \tau \leq 300$	$5 \leq \tau \leq 300$	$\tau_i = \frac{\mu_i}{\mu} \sigma_i$	$\tau_i = \frac{-t}{\ln\left(\frac{-k_i}{\lambda_i c_{\max}}\right)}$	$1.000 \leq \tau \leq 15.000$
ótimal. global	27	30	25	27	30	32
ótimal. local	5	2	7	5	2	0
infactibilidade	0	0	0	0	0	0
ótimal. global (%)	84	94	78	84	94	100
erro médio (%)	3,26	2,75	1,09	1,55	0,46	0,00
iterações	31	42	33	39	42	3034

Os resultados para este grupo de matrizes foram considerados satisfatórios, pois a metodologia proposta neste trabalho, juntamente com os parâmetros definidos no Capítulo III estabelecem condições para que as soluções ótimas globais sejam encontradas em mais de 78% destas matrizes, e o erro médio para estas matrizes encontra-se abaixo de 3,26%.

Os ajustes dos parâmetros que definem melhores resultados para a técnica *Winner Takes All* são os que utilizam o desvio padrão entre os custos para λ , e as expressões (3.3.16) e (3.3.18) para τ ; estes resultados encontram-se na segunda e na quinta colunas da Tabela 5.1. A utilização dos parâmetros na forma vetorial também mostrou-se eficiente para estas matrizes, de acordo com os resultados da terceira, quarta e quinta colunas da Tabela 5.1.

O erro médio encontrado neste grupo de matrizes varia entre 0,46% a 3,26%, demonstrando que as soluções ótimas locais encontradas estão próximas das soluções ótimas globais. O percentual de soluções ótimas globais encontradas para este grupo de matrizes está

sempre acima de 78%, o que mostra que a técnica proposta é eficiente para resolver o problema da Designação com matrizes com única solução ótima global.

Na tabela 5.2, encontram-se os resultados para as matrizes com múltiplas soluções ótimas e/ou soluções ótimas próximas.

Os resultados mostrados na Tabela 5.2 mostram que as técnicas de dispersão entre os elementos da matriz c são mais eficientes para o uso da técnica de ajuste *Winner Takes All*. A determinação do parâmetro λ com a utilização do desvio padrão entre os custos resulta em melhores resultados para este grupo de matrizes, com 95% das soluções ótimas encontradas (quinta coluna da Tabela 5.2).

TABELA 5.2 – RESULTADOS PARA AS 41 MATRIZES COM MÚLTIPLAS SOLUÇÕES ÓTIMAS

λ	$\lambda = \eta/C_{\max}$	$\lambda = \eta/\sigma$	$\lambda_i = \eta/c_{i \max}$	$\lambda = \eta/\sigma$	$\lambda_i = \eta/\sigma_i$	$\lambda = \eta/C_{\max}$
τ	$5 \leq \tau \leq 300$ $5 \leq \tau \leq 300$ $5 \leq \tau \leq 300$ $\tau_i = \frac{\mu_i}{\mu} \sigma_i$ $\tau_i = \frac{-t}{\ln\left(\frac{-k_i}{\lambda_i c_{\max}}\right)}$ $1.000 \leq \tau \leq 15.000$					
ótimal. global	25	35	26	29	39	0
ótimal. local	16	6	15	12	2	0
infactibilidade	0	0	0	0	0	41
ótimal. global (%)	61	85	63	71	95	0
erro médio (%)	3,78	1,09	6,88	5,62	0,13	-
iterações (média)	37	42	40	89	43	5220

O erro médio encontrado neste grupo de matrizes encontra-se entre 0,13% e 6,88%, reafirmando a eficiência da técnica proposta para resolver o problema da Designação. A Rede Neural de Wang “pura” obteve soluções infactíveis em todas as matrizes testadas que possuem múltiplas soluções ótimas, como pode ser visto na sexta coluna da Tabela 5.2. Os resultados para as 73 matrizes testadas neste trabalho são apresentados na Tabela 5.3.

TABELA 5.3 – RESULTADOS PARA AS 73 MATRIZES TESTADAS

λ	$\lambda = \eta/C_{\max}$	$\lambda = \eta/\sigma$	$\lambda_i = \eta/c_{i \max}$	$\lambda = \eta/\sigma$	$\lambda_i = \eta/\sigma_i$	$\lambda = \eta/C_{\max}$
τ	$5 \leq \tau \leq 300$ $5 \leq \tau \leq 300$ $5 \leq \tau \leq 300$ $\tau_i = \frac{\mu_i}{\mu} \sigma_i$ $\tau_i = \frac{-t}{\ln\left(\frac{-k_i}{\lambda_i c_{\max}}\right)}$ $1.000 \leq \tau \leq 15.000$					
ótimal. global	52	65	51	56	69	32
ótimal. local	21	8	22	17	4	0
infactibilidade	0	0	0	0	0	41
ótimal. global (%)	71	89	70	77	95	44
erro médio (%)	3,51	1,96	3,85	3,49	0,3	-
iterações (média)	35	42	37	67	37	4259

O número médio de iterações necessárias para a convergência da Rede de Wang com a técnica de ajuste proposta varia entre 0,82% e 1,5% do número de iterações necessárias para a convergência da Rede de Wang pura, reduzindo consideravelmente o tempo de processamento da rede, com resultados satisfatórios.

A Rede Neural de Wang pura tem convergência mais lenta quando são aplicados os ajustes dos parâmetros descritos pelas expressões (3.3.11)-(3.3.13) para λ , e pelas expressões (3.3.15)-(3.3.18) para τ .

Nas próximas seções 5.3 e 5.4 são apresentados os resultados da aplicação das duas técnicas propostas no Capítulo III (Rede de Kohonen para definir os custos do problema e a Rede de Wang com o princípio *Winner Takes All* para resolver o problema) para a resolução do problema definido no Capítulo IV deste trabalho.

5.3 RESULTADOS OBTIDOS COM A APLICAÇÃO DA REDE DE KOHONEN AO PROBLEMA DE ALOCAÇÃO DE SALAS DE AULA DA UFPR

A alocação das 650 disciplinas para o 1º Semestre do ano letivo de 2005 foi utilizada neste trabalho, com 433 modelos para o treinamento da Rede de Kohonen, e 46 salas disponíveis nos blocos citados na seção 4.2 do Capítulo IV deste trabalho.

As dimensões dos mapas testados variam de 3×3 a 26×26 , com vizinhança Gaussiana do tipo quadrada, e com a utilização da atualização exponencial para as taxas de aprendizagem e do raio da vizinhança dos mapas, dadas pelas expressões (3.2.11) e (3.2.10), respectivamente. Na figura 5.1, encontra-se a representação dos mapas de densidade e da U-matriz, respectivamente, de um mapa 10×10 com 433 dados de entrada, após 9.600 iterações de treinamento da rede. Nesta figura, os nove agrupamentos resultantes estão representados através do sistema de cores primárias *RGB* (*Red-Green-Blue*: Vermelho-Verde-Azul), de acordo com os valores das três coordenadas dos pesos resultantes para cada neurônio do mapa.

Na Tabela 5.4 encontram-se os significados dos elementos de cada grupo criado no mapa representado na Figura 5.1, onde são representados os valores mínimo (m) e máximo (M) de cada coordenada dos vetores $p(p_1, p_2, p_3)$ de cada agrupamento.

Por exemplo, o agrupamento 9 representa o grupo de disciplinas com escolha de sala sem carteiras de braço, com bloco de preferência entre os blocos PD e PH, com número de alunos matriculados variando entre 35 e 70.

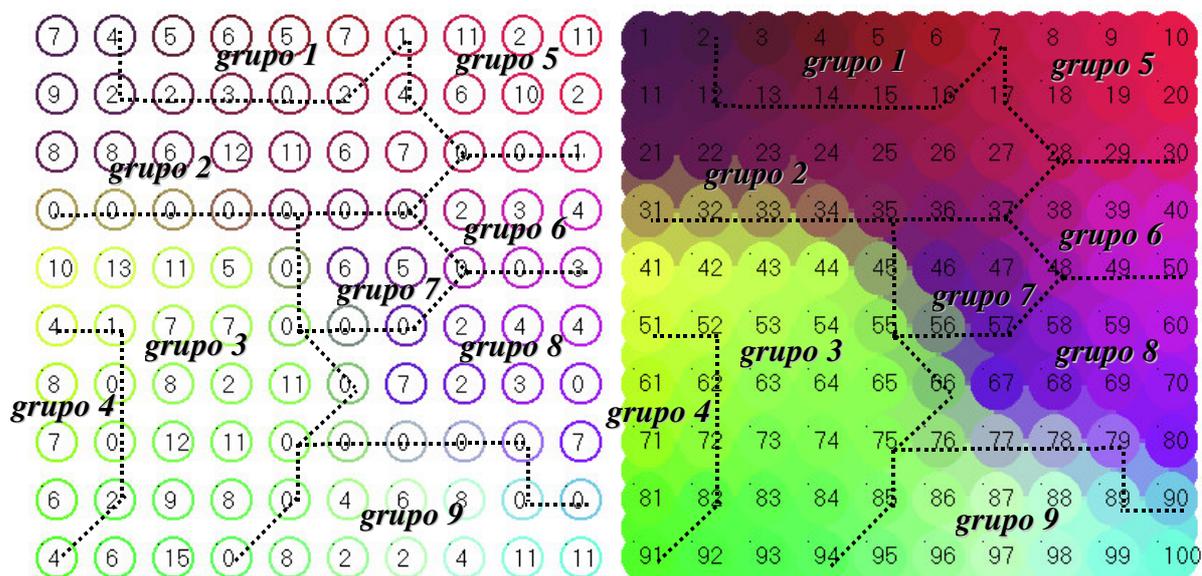


FIGURA 5.1 – MAPA DE DENSIDADE E U-MATRIZ DE UMA REDE DE KOHONEN 10 × 10, COM 433 PADRÕES DE ENTRADA

TABELA 5.4 – SIGNIFICADO DE CADA AGRUPAMENTO CRIADO NO EXEMPLO DA FIGURA 5.1

	Grupo 1		Grupo 2		Grupo 3		Grupo 4		Grupo 5	
	m	M	m	M	m	M	m	M	m	M
p_1	0,3	1	0,3	1	0,5	1	0,3	1	0,3	1
capac.	35	115	35	115	60	115	35	115	35	115
p_2	0		0		1		1		0	
cart.	qualquer		qualquer		desenho		desenho		qualquer	
p_3	0		0,1	0,5	0	0,5	0		0,1	0,5
bloco	PC		PA, CT, PD		PA, CT, PD		PC		PC, PA, CT, PD	
	Grupo 6		Grupo 7		Grupo 8		Grupo 9			
p_1	0,5	0,3	0,5	0,5	0,5	0,6	0,3	0,6		
capac.	60	35	60	60	60	70	35	70		
p_2	0		0		1		1			
cart.	qualquer		qualquer		desenho		desenho			
p_3	0		0,1	0,5	0,5	1	0,5	1		
bloco	PD		PA, CT, PC		PD, PE, PF, PG, PH		PD, PE, PF, PG, PH			

NOTA: m – VALOR MÍNIMO DE CADA COORDENADA; M – VALOR MÁXIMO DE CADA COORDENADA; **capac.** – CAPACIDADE; **cart.** – TIPO DE CARTEIRA; (p_1, p_2, p_3) – VETOR DOS PADRÕES DE ENTRADA DA REDE.

Após o treinamento da rede, os conjuntos das 650 disciplinas e das 46 salas de aula passam pela classificação de seus elementos a partir dos pesos encontrados no treinamento da rede, criando-se, respectivamente, os mapas M' e M'' para tais conjuntos. O custo da

designação de uma sala i para uma disciplina j é dado pela expressão (3.2.16), que mede a distância Euclidiana entre os neurônios vencedores da sala i e da disciplina j .

Os testes para determinar o melhor mapa envolvem o cálculo dos erros definidos no Capítulo IV anterior para três conjuntos de dados: as disciplinas escolhidas para o treinamento da rede (433), o conjunto de treinamento das disciplinas (217) e as salas disponíveis (46). Nas figuras 5.2, 5.3 e 5.4, encontram-se os erros topológico, de quantização e médio quadrático, respectivamente, referentes aos mapas gerados com os dados do problema considerado neste trabalho. Estes erros medem a qualidade de mapas com dimensão variando entre 3 e 26, para os três conjuntos de dados do problema.

Os mapas criados possuem vizinhança do tipo Gaussiana, com topologia do tipo quadrada, a entrada dos dados na fase de treinamento dos mapas é aleatória e a atualização da taxa de aprendizagem e do raio de vizinhança é exponencial. O raio de vizinhança inicial é igual ao número de neurônios do mapa, e a taxa de aprendizagem inicial é de 0,3.

De acordo com os erros topológicos encontrados para os três grupos de dados do problema de Alocação de salas, pode-se afirmar que tais erros são estáveis e mínimos para os mapas de dimensão maior do que 23×23 , conforme mostra a Figura 5.2.

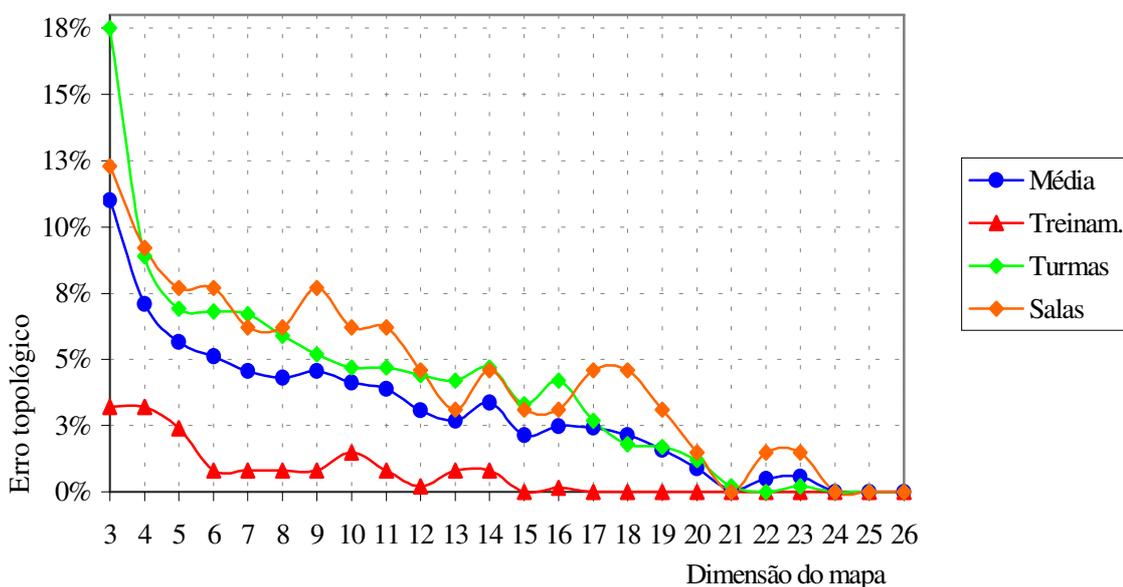


FIGURA 5.2 – ERRO TOPOLÓGICO PARA OS TRÊS GRUPOS DE DADOS

Os erros de quantização encontrados nos mapas testados possuem maior estabilidade do que os erros topológicos destes mapas, e apresentam melhores resultados para as classificações dos dados para os mapas com dimensão maior do que 14×14 , conforme

mostra a Figura 5.3. Os mapas com dimensão maior do que 14×14 possuem erros de quantização satisfatórios, menores do que $2,5 \times 10^{-3}$ para o conjunto de treinamento, $0,06$ para o conjunto de testes das salas, e $0,025$ para o conjunto de testes das disciplinas. Os erros médios quadráticos para estes mapas são inferiores a 5×10^{-5} para o conjunto de treinamento, 1×10^{-3} para o conjunto de testes das salas de aula, e $1,5 \times 10^{-4}$ para o conjunto de testes das disciplinas (Figura 5.4).

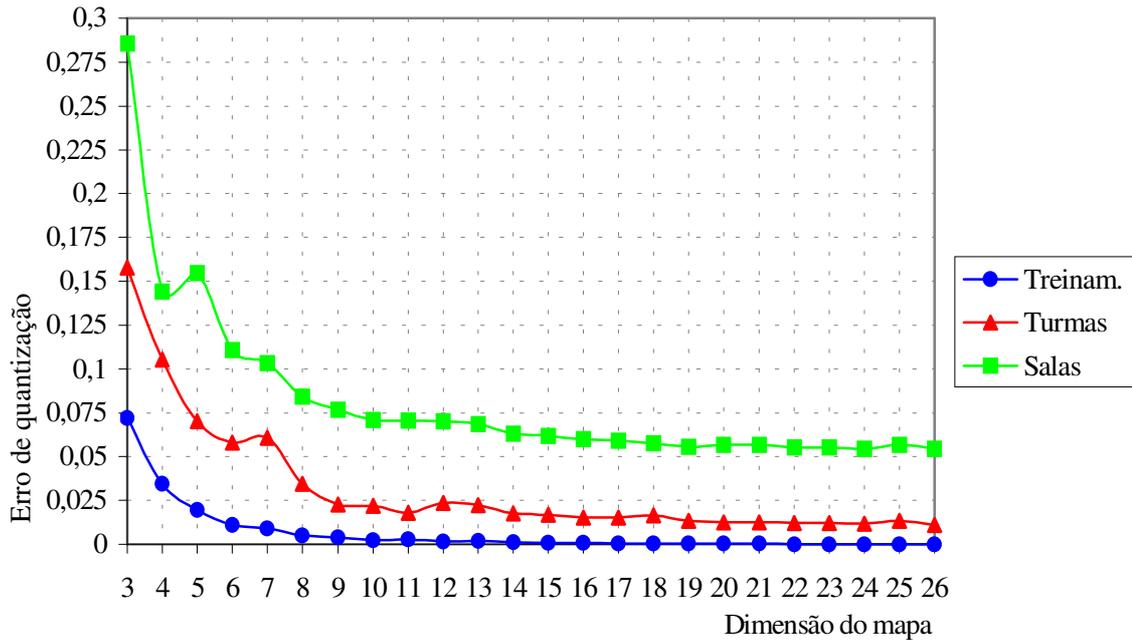


FIGURA 5.3 – ERRO DE QUANTIZAÇÃO PARA OS TRÊS GRUPOS DE DADOS

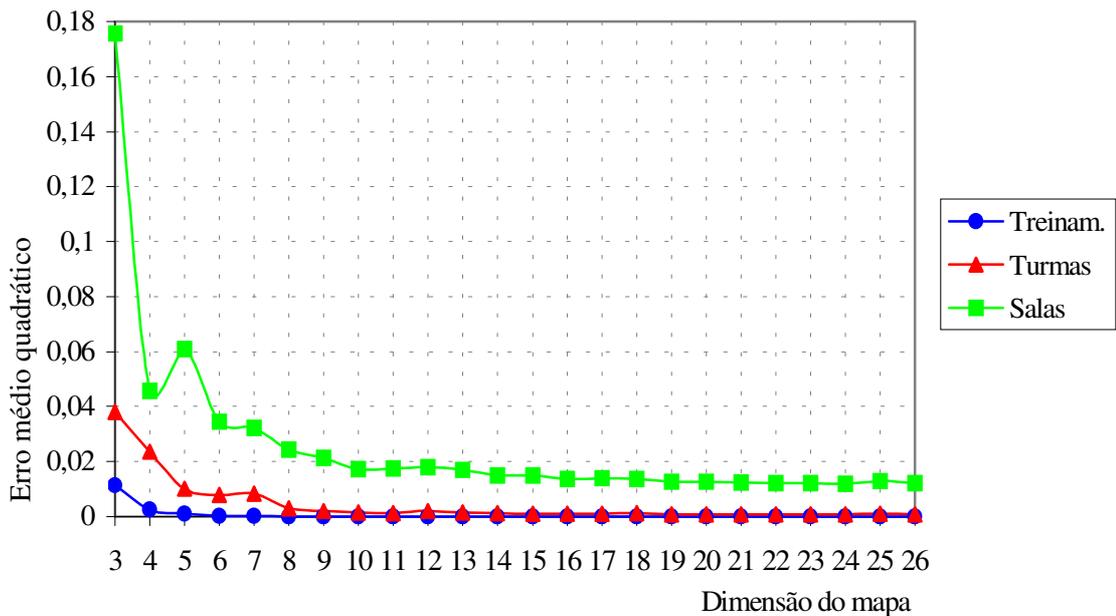


FIGURA 5.4 – ERRO MÉDIO QUADRÁTICO PARA OS TRÊS GRUPOS DE DADOS

A escolha do melhor mapa para o problema em estudo deve levar em conta os três erros encontrados. Na próxima seção são apresentados os custos finais obtidos através da aplicação da Rede Neural de Wang com o princípio *Winner Takes All* nos mapas citados nesta seção.

5.4 APLICAÇÃO DA REDE NEURAL DE WANG COM O PRINCÍPIO *WINNER TAKES ALL* PARA O PROBLEMA DE ALOCAÇÃO DE SALAS DA UFPR

A escolha do melhor mapa obtido para a obtenção da melhor designação possível para os dados utilizados neste trabalho é mostrada nesta seção.

A solução ótima para os dados utilizados neste trabalho para o problema da Designação das salas de aula para as disciplinas é de 41,46, o que significa um custo médio para cada designação de 0,064.

Os parâmetros utilizados para a resolução do problema da Designação com a Rede de Wang e o princípio *Winner Takes All* são os seguintes: $\eta = 1$, $\lambda_i = \eta/d_i$ e as expressões (3.3.17) e (3.3.18) para o parâmetro τ , onde d_i representa o desvio padrão entre os elementos da linha i da matriz de custos do problema de cada horário dos dias de semana. Esta é a escolha de parâmetros que produz melhores resultados para o problema da Designação, de acordo com os resultados mostrados na seção 5.2.

Os mapas que possuem os menores erros de quantização e topológico produzem melhores resultados para a designação final (Figura 5.5). Para os dados do problema mostrado neste trabalho, os melhores mapas são os que possuem dimensões 23×23 , 24×24 , 25×25 e 26×26 , com custos finais de 41,84; 41,82; 41,76 e 41,71, respectivamente.

Os mapas com dimensões maiores do que 15×15 fornecem soluções ótimas locais bem próximas da solução ótima global para o problema, com erro médio inferior a 3%, como mostra a Figura 5.5.

No próximo capítulo, a metodologia proposta neste trabalho para a resolução do problema da Designação é aplicada ao problema do Caixeiro Viajante, utilizando os parâmetros que determinam melhores resultados para o problema da Designação.

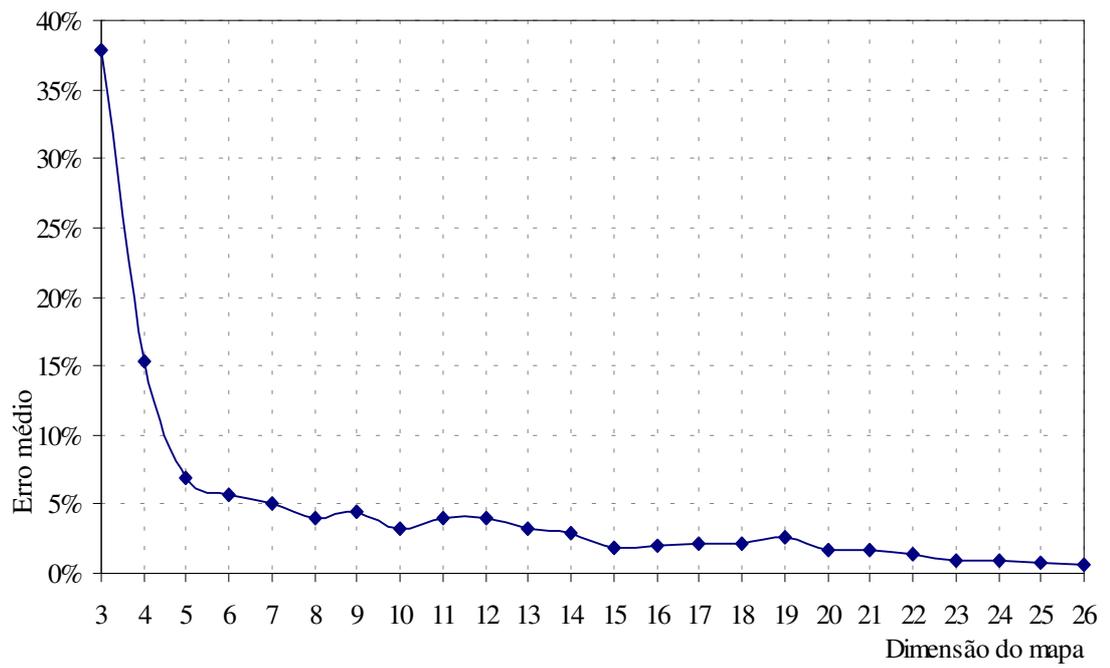


FIGURA 5.5 – ERRO MÉDIO ENCONTRADO PARA A ALOCAÇÃO DAS DISCIPLINAS PARA AS SALAS DE AULA

CAPÍTULO VI

6 RESOLUÇÃO DO PROBLEMA DO CAIXEIRO VIAJANTE COM A APLICAÇÃO DA REDE NEURAL DE WANG COM O PRINCÍPIO *WINNER TAKES ALL*

6.1 DESCRIÇÃO DO PROBLEMA DO CAIXEIRO VIAJANTE

Seja um conjunto de pontos representando n cidades. O problema do Caixeiro Viajante consiste na determinação de uma rota que inicia em uma cidade, passa por cada cidade do conjunto apenas uma vez, e retorna à cidade inicial da rota perfazendo uma distância total mínima. Esta rota é denominada ciclo Hamiltoniano de custo mínimo.

A representação deste problema pode ser feita através de um grafo bi-partido G , onde cada subconjunto de vértices deste grafo contém as n cidades consideradas. Os custos para a designação de uma cidade i para uma cidade j , onde $i \neq j$, são dados pela distância Euclidiana entre as cidades no problema simétrico, ou pela distância mínima real entre as cidades no problema assimétrico. A distância real entre duas cidades leva em conta diversos fatores, como por exemplo os sentidos das ruas utilizadas para determinar a viagem entre estas cidades.

Existem diversas aplicações práticas para este problema, tais como Roteamento de Veículos (LAPORTE, 1992, AHUJA; MAGNANTI; ORLIN, 1993) e Problemas de Perfuração (ONWUBOLU; CLERC, 2004).

O método Húngaro pode ser utilizado para resolver o problema do Caixeiro Viajante. Se a solução do problema da Designação forma uma rota viável para o problema do Caixeiro Viajante, então esta solução é ótima para o Caixeiro Viajante. Caso contrário, uma variante do método *Branch and Bound* (MURTY, 1985) pode ser utilizada para determinar rotas viáveis para o problema do Caixeiro Viajante, onde são definidos vários problemas da Designação a

partir da solução inicial, encontrada através do Método Húngaro. A melhor solução viável destes problemas definidos é a solução ótima do problema do Caixeiro Viajante.

A Rede Neural Recorrente de Wang, com o princípio *Winner Takes All* (SIQUEIRA; SCHEER; STEINER, 2005) pode ser aplicada na resolução do problema do Caixeiro Viajante, resolvendo-se este problema como um Problema da Designação através da Rede de Wang, e utilizando-se o princípio *Winner Takes All* nas soluções encontradas com a Rede de Wang, com a restrição de que as soluções modificadas formem uma rota viável para o problema do Caixeiro Viajante. Os parâmetros utilizados para a Rede de Wang são os que apresentam as melhores soluções para o problema da Designação, conforme mostram os resultados apresentados no Capítulo V deste trabalho.

As soluções encontradas com a técnica proposta neste trabalho são comparadas com as soluções dos mapas Auto-Organizáveis e Têmpera Simulada para o problema simétrico, e com heurísticas de inserção e remoção de arcos para o problema assimétrico. A técnica de Busca Local *2-opt* (BIANCHI; KNOWLES; BOWLER, 2005) é utilizada para melhorar as soluções encontradas com a técnica proposta neste trabalho, e os dados utilizados para as comparações feitas são do banco de dados *TSPLIB*.

6.1.1 Formulação do problema do Caixeiro Viajante

O problema do Caixeiro Viajante pode ser formulado matematicamente através do seguinte Modelo de Programação Linear Inteira Binária (MURTY, 1985):

$$\text{Minimizar } C = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij} \quad (6.1.1)$$

$$\text{Sujeito a } \sum_{i=1}^n x_{ij} = 1, \quad j = 1, 2, \dots, n \quad (6.1.2)$$

$$\sum_{j=1}^n x_{ij} = 1, \quad i = 1, 2, \dots, n \quad (6.1.3)$$

$$x_{ij} \in \{0, 1\}, \quad i, j = 1, 2, \dots, n \quad (6.1.4)$$

$$\tilde{x} \text{ forma um ciclo Hamiltoniano} \quad (6.1.5)$$

onde c_{ij} e x_{ij} são, respectivamente, os custos e as variáveis de decisão associados à designação do elemento i à posição j . Quando $x_{ij} = 1$, o elemento i é designado à posição j , o que significa que a rota possui um trecho que é a seqüência da cidade i para a cidade j . Para iniciar a

resolução deste problema, deve-se fazer $c_{ii} = \infty$ (para $i = 1, 2, \dots, n$), com o objetivo de eliminar as soluções sem sentido $x_{ii} = 1$. O vetor \tilde{x} possui toda a seqüência da rota encontrada, ou seja, a solução para o problema do Caixeiro Viajante.

O problema da Designação possui a mesma formulação anterior, com exceção da restrição (6.1.5). A função objetivo (6.1.1) representa o custo total a ser minimizado. Os conjuntos de restrições (6.1.2) e (6.1.3) garantem que cada cidade i será designada para exatamente uma cidade j . O conjunto (6.1.4) representa as restrições de integralidade das variáveis zero-um x_{ij} . A restrição (6.1.5) garante que na rota final cada cidade será visitada uma vez, sem a formação de sub-rotas.

Na próxima seção é apresentada a metodologia modificada para resolver o problema do Caixeiro Viajante com a Rede Neural de Wang com o princípio *Winner Takes All*.

6.2 SOLUÇÃO DO PROBLEMA DO CAIXEIRO VIAJANTE UTILIZANDO A REDE NEURAL DE WANG COM O PRINCÍPIO *WINNER TAKES ALL*

A formulação dada por (6.1.1)-(6.1.5) permite que a Rede Neural de Wang possa ser utilizada para resolver o problema do Caixeiro Viajante. Entretanto, algumas mudanças devem ser feitas no princípio *Winner Takes All*, para que a restrição (6.1.5) seja satisfeita, criando assim rotas viáveis para o problema em questão.

Os parâmetros escolhidos para a Rede Neural de Wang são aqueles que determinam melhores resultados para o problema da Designação, conforme mostram os resultados apresentados no Capítulo V deste trabalho. O parâmetro η é considerado igual a 1 em todos os casos testados, o parâmetro λ é considerado como um vetor, definido pela expressão (3.3.12), e o parâmetro λ é definido pelas expressões (3.3.17) e (3.3.18).

O mesmo critério utilizado para iniciar a aplicação do critério *Winner Takes All* para o problema da Designação é usado no problema do Caixeiro Viajante, ou seja, quando todos os elementos do vetor x satisfazem $|Wx(t) - \theta| \leq \delta$, onde $\delta \in [0, 2]$, o princípio proposto pode ser utilizado, e seu algoritmo é apresentado a seguir.

Algoritmo baseado no princípio *Winner Takes All* para a aplicação da Rede Neural de Wang para o problema do Caixeiro Viajante

Passo 1: Determine um número máximo de rotas r_{\max} . Encontre uma solução x do problema da Designação, utilizando a Rede Neural de Wang. Se $|Wx(t) - \theta| \leq \delta$, então vá para o passo 2. Caso contrário, encontre uma outra solução x .

Passo 2: Dada a matriz de decisão, considere a matriz \bar{x} , onde $\bar{x} = x$, $m = 1$, e vá ao passo 3.

Passo 3: Escolha uma linha k da matriz de decisão \bar{x} . Faça $p = k$, $\tilde{x}(m) = k$ e vá ao passo 4.

Passo 4: Encontre o maior elemento da linha k , \bar{x}_{kl} . O valor deste elemento é dado pela metade da soma de todos os elementos da linha k e coluna l da matriz x , ou seja,

$$\bar{x}_{kl} = \frac{1}{2} \left(\sum_{i=1}^n x_{il} + \sum_{j=1}^n x_{kj} \right). \quad (6.2.1)$$

Os outros elementos da linha k e coluna l tornam-se nulos. Para evitar a formação de sub-rotas, os elementos da coluna k devem ser também nulos. Faça $\tilde{x}(m+1) = l$, para dar continuidade à rota do Caixeiro Viajante, faça $k = l$, e vá ao passo 5.

Passo 5: Se $m < n$, então faça $m = m + 1$ e vá ao passo 4. Caso contrário, faça

$$\bar{x}_{kp} = \frac{1}{2} \left(\sum_{i=1}^n x_{ip} + \sum_{j=1}^n x_{kj} \right), \quad (6.2.2)$$

$\tilde{x}(n+1) = p$, determine o custo da rota, C , e vá ao passo 6.

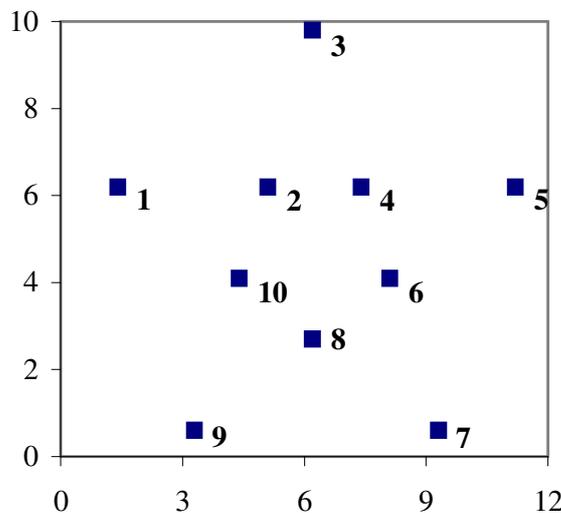
Passo 6: Se $C < C_{\min}$, então faça $C_{\min} = C$, $x_{\min} = \bar{x}$ e $x = \bar{x}$. Caso contrário, faça $x = x_{\min}$. Faça $r = r + 1$. Se $r < r_{\max}$, então execute novamente a Rede de Wang, e vá ao passo 2, caso contrário, pare: a solução factível encontrada é x_{\min} .

Esta técnica proposta pode ser aplicada tanto em problemas simétricos quanto assimétricos. Na próxima seção é apresentado um exemplo da aplicação desta técnica para um problema simétrico, com 10 cidades.

6.2.1 Exemplo Ilustrativo

Considere o problema do Caixeiro Viajante simétrico, com as coordenadas de 10 cidades, como mostra a figura 6.1. Após 21 iterações, a Rede Neural de Wang apresenta a seguinte solução para o problema da Designação:

$$\bar{x} = \begin{pmatrix} 0 & 0,2557 & 0,2436 & 0,0063 & 0,0004 & 0,0011 & 0,0004 & 0,0073 & 0,2198 & 0,2654 \\ 0,2630 & 0 & 0,2744 & 0,2060 & 0,0012 & 0,0094 & 0,0001 & 0,0097 & 0,0017 & 0,2345 \\ 0,2334 & 0,2722 & 0 & 0,2669 & 0,2108 & 0,0072 & 0,0005 & 0,0011 & 0,0005 & 0,0074 \\ 0,0013 & 0,2114 & 0,2649 & 0 & 0,2509 & 0,2510 & 0,0017 & 0,0093 & 0,0001 & 0,0094 \\ 0,0004 & 0,0058 & 0,2125 & 0,2492 & 0 & 0,2650 & 0,2597 & 0,0061 & 0,0004 & 0,0010 \\ 0,0001 & 0,0093 & 0,0016 & 0,2445 & 0,2723 & 0 & 0,2717 & 0,1907 & 0,0014 & 0,0083 \\ 0,0004 & 0,0011 & 0,0005 & 0,0074 & 0,2624 & 0,2655 & 0 & 0,2606 & 0,1963 & 0,0059 \\ 0,0017 & 0,0096 & 0,0001 & 0,0092 & 0,0014 & 0,1860 & 0,2613 & 0 & 0,3188 & 0,2120 \\ 0,2213 & 0,0071 & 0,0005 & 0,0011 & 0,0004 & 0,0065 & 0,2031 & 0,3037 & 0 & 0,2562 \\ *0,278 & 0,2279 & 0,0019 & 0,0094 & 0,0001 & 0,0084 & 0,0014 & 0,2116 & 0,2611 & 0 \end{pmatrix}$$



cidade	x	y
1	1,4	6,2
2	5,1	6,2
3	6,2	9,8
4	7,4	6,2
5	11,2	6,2
6	8,1	4,1
7	9,3	0,6
8	6,2	2,7
9	3,3	0,6
10	4,4	4,1

FIGURA 6.1 – EXEMPLO DO PROBLEMA DO CAIXEIRO VIAJANTE COM 10 CIDADES

Nesta matriz de decisão, uma cidade é escolhida para iniciar a rota, por exemplo, a cidade 10, ou seja, $p = 10$. Na linha p da matriz de decisão, escolhe-se o maior elemento, definindo o destino do Caixeiro Viajante ao sair da cidade p . O maior elemento da linha p encontra-se na coluna 1, logo, $k = p = 10$ e $l = 1$. Após atualizar a matriz de decisão \bar{x} através da equação (6.2.1), a rota prossegue, com $k = 1$:

$$\bar{x} = \begin{pmatrix} 0 & *0,256 & 0,2436 & 0,0063 & 0,0004 & 0,0011 & 0,0004 & 0,0073 & 0,2198 & 0 \\ 0 & 0 & 0,2744 & 0,2060 & 0,0012 & 0,0094 & 0,0001 & 0,0097 & 0,0017 & 0 \\ 0 & 0,2722 & 0 & 0,2669 & 0,2108 & 0,0072 & 0,0005 & 0,0011 & 0,0005 & 0 \\ 0 & 0,2114 & 0,2649 & 0 & 0,2509 & 0,2510 & 0,0017 & 0,0093 & 0,0001 & 0 \\ 0 & 0,0058 & 0,2125 & 0,2492 & 0 & 0,2650 & 0,2597 & 0,0061 & 0,0004 & 0 \\ 0 & 0,0093 & 0,0016 & 0,2445 & 0,2723 & 0 & 0,2717 & 0,1907 & 0,0014 & 0 \\ 0 & 0,0011 & 0,0005 & 0,0074 & 0,2624 & 0,2655 & 0 & 0,2606 & 0,1963 & 0 \\ 0 & 0,0096 & 0,0001 & 0,0092 & 0,0014 & 0,1860 & 0,2613 & 0 & 0,3188 & 0 \\ 0 & 0,0071 & 0,0005 & 0,0011 & 0,0004 & 0,0065 & 0,2031 & 0,3037 & 0 & 0 \\ 0,9999 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

O maior elemento da linha 1 na matriz \bar{x} encontra-se na coluna 2, logo, $l = 2$. Este procedimento é feito, até que todas as linhas sejam atualizadas, definindo a rota: $\tilde{x} = (10, 1, 2, 3, 4, 6, 5, 7, 8, 9, 10)$, mostrada na figura 6.2A, com custo de 37,79 que representa um erro médio de 1,7%.

$$\bar{x} = \begin{pmatrix} 0 & 0,9999 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0,9998 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0,9998 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0,9999 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0,9998 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,9999 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,9999 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,9997 \\ 0,9999 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Esta solução é apresentada à Rede Neural de Wang, fazendo-se $x = \bar{x}$. Após 7 iterações, a Rede de Wang apresenta a seguinte matriz de decisão:

$$\bar{x} = \begin{pmatrix} 0 & 0,2557 & 0,2436 & 0,0063 & 0,0004 & 0,0011 & 0,0004 & 0,0073 & 0,2198 & 0,2654 \\ 0,2630 & 0 & 0,2744 & 0,2062 & 0,0011 & 0,0094 & 0 & 0,0097 & 0,0017 & 0,2345 \\ 0,2334 & 0,2722 & 0 & 0,2669 & 0,2108 & 0,0072 & 0,0005 & 0,0011 & 0,0005 & 0,0074 \\ 0,0013 & 0,2114 & 0,2649 & 0 & 0,2511 & 0,2508 & 0,0017 & 0,0093 & 0,0001 & 0,0094 \\ 0,0004 & 0,0058 & 0,2125 & 0,2492 & 0 & 0,2650 & 0,2597 & 0,0061 & 0,0004 & 0,0010 \\ 0,0001 & 0,0093 & 0,0016 & 0,2445 & 0,2723 & 0 & 0,2717 & 0,1907 & 0,0014 & 0,0083 \\ 0,0004 & 0,0011 & 0,0005 & 0,0074 & 0,2624 & 0,2655 & 0 & 0,2608 & 0,1963 & 0,0059 \\ 0,0017 & 0,0095 & 0,0001 & 0,0092 & 0,0014 & 0,1860 & 0,2613 & 0 & 0,3189 & 0,2120 \\ 0,2213 & 0,0071 & 0,0005 & 0,0011 & 0,0004 & 0,0065 & 0,2031 & 0,3037 & 0 & 0,2562 \\ 0,2783 & 0,2279 & 0,0019 & 0,0094 & 0 & 0,0083 & 0,0013 & 0,2116 & 0,2614 & 0 \end{pmatrix}$$

Através do princípio *Winner Takes All*, uma aproximação para a solução ótima deste problema é encontrada, com a rota: $\tilde{x} = (10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10)$, com custo de 37,159 (Figura 6.2B).

$$\bar{x} = \begin{pmatrix} 0 & 1,0002 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0,9999 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1,0002 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1,0001 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0,9998 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,9999 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1,0001 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Na próxima seção são mostrados os resultados da aplicação desta técnica para alguns problemas do *TSPLIB*, tanto para problemas simétricos quanto assimétricos.

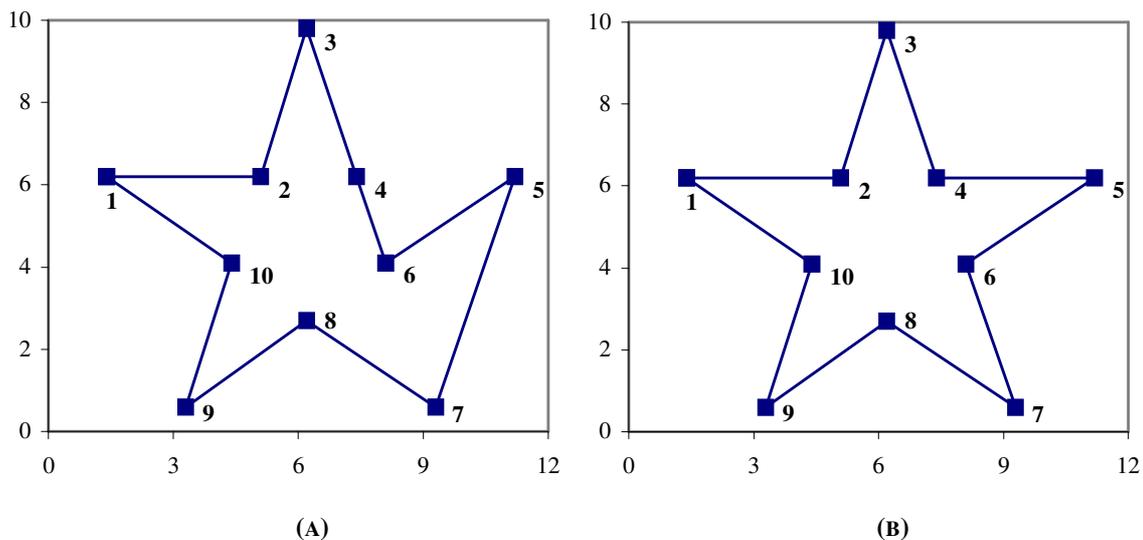


FIGURA 6.2 - (A) SOLUÇÃO FACTÍVEL, ENCONTRADA COM O MÉTODO PROPOSTO, COM ERRO MÉDIO DE 1,7%.
(B) SOLUÇÃO ÓTIMA, ENCONTRADA ATRAVÉS DO MÉTODO PROPOSTO

6.3 RESULTADOS DA APLICAÇÃO DA TÉCNICA PROPOSTA EM PROBLEMAS DO *TSPLIB*

Os resultados encontrados com a técnica proposta neste trabalho para os casos simétricos do problema do Caixeiro Viajante do banco de dados *TSPLIB* são comparados com resultados de mapas Auto-Organizáveis de Kohonen e Têmpera Simulada, e os casos assimétricos são comparados com heurísticas de remoção e inserção de arcos.

Nas Tabelas 6.1 e 6.2 encontram-se, respectivamente, as descrições dos problemas simétricos e assimétricos utilizados neste trabalho. Com exceção dos problemas *laurene10* e *katta06*, todos os problemas citados pertencem ao banco de dados *TSPLIB*.

Para problemas simétricos, os métodos utilizados como comparação da técnica proposta neste trabalho são os seguintes: o método que envolve métodos estatísticos entre os pesos dos neurônios de um mapa de Kohonen (ARAS; OOMEN; ALTINEL, 1999), com a versão global (*KniesG*), e a versão local (*KniesL*); a técnica Têmpera Simulada (*TS*) (BUDINICH, 1996), com a utilização da técnica de melhoria *2-opt*; o mapa Auto-Organizável de Budinich, apresentado em (BUDINICH, 1996); o mapa Auto-Organizável expandido (*ESOM*) (LEUNG; JIN; XU, 2004); o mapa Auto-Organizável eficiente e integrado (*eISOM*) (JIN et. al, 2003); a técnica mapa Auto-Organizável eficiente (*SETSP*) (VIEIRA; DORIA NETO; COSTA, 2002); e a rede Cooperativa Adaptativa de Kohonen (*CAN*) (COCHRANE; BEASLEY, 2003). Um breve relato sobre estas técnicas encontra-se no Capítulo II deste

trabalho. Na tabela 6.3 são mostrados os erros médios das técnicas acima citadas, além da técnica ‘pura’ proposta neste trabalho (onde a coluna ‘max’ indica os melhores resultados e a coluna ‘min’ indica os piores resultados para cada problema) e da técnica proposta com o algoritmo de melhoria *2-opt*.

TABELA 6.1 - PROBLEMAS SIMÉTRICOS DO CAIXEIRO VIAJANTE UTILIZADOS NESTE TRABALHO

nome do PCV	número de cidades	descrição do problema
laurene10	10	Exemplo do livro de Laurene Fausett, 1994, pág. 336
burma14	14	14 cidades de Burma (coordenadas geográficas)
ulysses16	16	odisséia de Ulysses (Groetschel e Padberg)
ulysses22	22	odisséia de Ulysses (Groetschel e Padberg)
bays29	29	29 cidades da Bavaria (distâncias geográficas)
swiss42	42	42 cidades da Suíça (Fricker)
att48	48	48 capitais dos Estados Unidos (Padberg/Rinaldi)
eil51	51	problema de 51 cidades (Christofides/Elon)
berlin52	52	52 locações em Berlin (Alemanha) (Groetschel)
st70	70	problema de 70 cidades (Smith/Thompson)
eil76	76	problema de 76 cidades (Christofides/Elon)
pr76	76	problema de 76 cidades (Padberg/Rinaldi)
gr96	96	África: subproblema do problema de 666 cidades (Groetschel)
kroC100	100	problema de 100 cidades C (Krolak/Felts/Nelson)
kroD100	100	problema de 100 cidades D (Krolak/Felts/Nelson)
rd100	100	problema de 100 cidades aleatórias (Reinelt)
eil101	101	problema de 101 cidades (Christofides/Elon)
lin105	105	problema de 105 cidades (subproblema de lin318)
pr107	107	problema de 107 cidades (Padberg/Rinaldi)
pr124	124	problema de 124 cidades (Padberg/Rinaldi)
bier127	127	127 fábricas de cerveja em Augsburg (Alemanha) (Juenger/Reinelt)
ch130	130	problema de 130 cidades (Churritz)
pr136	136	problema de 136 cidades (Padberg/Rinaldi)
pr152	152	problema de 152 cidades (Padberg/Rinaldi)
u159	159	problema de perfuração (drilling problem) (Reinelt)
rat195	195	rattled grid (pulleyblank)
d198	198	problema de perfuração (drilling problem) (Reinelt)
kroA200	200	problema de 200 cidades A (Krolak/Felts/Nelson)
gr202	202	subproblema do problema de 666 cidades (Groetschel)
tsp225	225	Um problema PCV (Reinelt)
a280	280	problema de perfuração (drilling problem) (Ludwig)
pr299	299	problema de 299 cidades (Padberg/Rinaldi)
lin318	318	problema de 318 cidades (Lin/Kernighan)
rd400	400	problema de 400 cidades aleatórias (Reinelt)
pcb442	442	problema de perfuração (drilling problem) (Groetschel/Juenger/Reinelt)
att532	532	problema de 532 cidades (Padberg/Rinaldi)

TABELA 6.2 - PROBLEMAS ASSIMÉTRICOS DO CAIXEIRO VIAJANTE UTILIZADOS NESTE TRABALHO

nome do PCV	número de cidades	descrição do problema
katta06	6	exemplo do livro de Katta Murty, 1985, pág. 450
br17	17	problema proposto por Bruno Repetto
ftv33	33	problema proposto por Matteo Fischetti
ftv35	35	problema proposto por Matteo Fischetti
ftv38	38	problema proposto por Matteo Fischetti
pr43	43	problema proposto por Bruno Repetto (Pekny)
ftv44	44	problema proposto por Matteo Fischetti
ftv47	47	problema proposto por Matteo Fischetti
ry48p	48	problema proposto por Matteo Fischetti
ft53	53	problema proposto por Matteo Fischetti
ftv55	55	problema proposto por Matteo Fischetti
ftv64	64	problema proposto por Matteo Fischetti
ft70	70	problema proposto por Matteo Fischetti
ftv70	70	problema proposto por Matteo Fischetti
kro124p	100	problema proposto por Matteo Fischetti
ftv170	170	problema proposto por Matteo Fischetti
rgb323	323	aplicação no problema de guindaste de empilhador (Ascheuer)
rgb358	358	aplicação no problema de guindaste de empilhador (Ascheuer)
rgb403	403	aplicação no problema de guindaste de empilhador (Ascheuer)
rgb443	443	aplicação no problema de guindaste de empilhador (Ascheuer)

TABELA 6.3 – RESULTADOS DOS EXPERIMENTOS PARA O PCV SIMÉTRICO COM AS TÉCNICAS: *KNIESG*, *KNIESL*, *TS*, *SOM* DE BUDINICH, *ESOM*, *EISOM*, *SETSP*, *CAN*, *RNRW* COM *WTA*. AS SOLUÇÕES EM NEGRITO MOSTRAM OS MELHORES RESULTADOS PARA CADA PROBLEMA, DESCONSIDERANDO-SE OS RESULTADOS DA *RNRW* COM A TÉCNICA *2-OPT*.

nome do PCV	número de cidades	solução ótima	erro médio para 7 algoritmos (%)								erro médio (%) <i>RNRW</i> com <i>WTA</i>		
			<i>KniesG</i>	<i>KniesL</i>	<i>TS</i>	Budinich	<i>ESom</i>	<i>EiSom</i>	<i>Setsp</i>	<i>CAN</i>	max	min	<i>2opt</i>
eil51	51	430	2,86	2,86	2,33	3,10	2,10	2,56	2,22	0,94	1,16	1,16	0
st70	70	678,6	2,33	1,51	2,14	1,70	2,09	NC	1,60	1,33	4,04	2,71	0
eil76	76	545,4	5,48	4,98	5,54	5,32	3,89	NC	4,23	2,04	2,49	1,03	0
gr96	96	514	NC	NC	4,12	2,09	1,03	NC	NC	NC	6,61	4,28	0
rd100	100	7910	2,62	2,09	3,26	3,16	1,96	NC	2,60	1,23	7,17	6,83	0,08
eil101	101	629	5,63	4,66	5,74	5,24	3,43	3,59	NC	1,11	7,95	3,02	0,48
lin105	105	14383	1,29	1,98	1,87	1,71	0,25	NC	1,30	0	5,94	4,33	0,20
pr107	107	44303	0,42	0,73	1,54	1,32	1,48	NC	0,41	0,17	3,14	3,14	0
pr124	124	59030	0,49	0,08	1,26	1,62	0,67	NC	NC	2,36	2,63	0,33	0
bier127	127	118282	3,08	2,76	3,52	3,61	1,70	NC	1,85	0,69	5,08	4,22	0,37
pr136	136	96772	5,15	4,53	4,90	5,20	4,31	NC	4,40	3,94	6,86	5,99	1,21
pr152	152	73682	1,29	0,97	2,64	2,04	0,89	NC	1,17	0,74	3,27	3,23	0
rat195	195	2323	11,92	12,24	13,29	11,48	7,13	NC	11,19	5,27	8,82	5,55	3,31
kroa200	200	29368	6,57	5,72	5,61	6,13	2,91	1,64	3,12	0,92	12,25	8,95	0,62
lin318	318	42029	NC	NC	7,56	8,19	4,11	2,05	NC	2,65	8,65	8,35	1,90
pcb442	442	50784	10,45	11,07	9,15	8,43	7,43	6,11	10,16	5,89	13,18	9,16	2,87
att532	532	27686	6,8	6,74	5,38	5,67	4,95	3,35	NC	3,32	15,43	14,58	1,28

Os resultados da técnica proposta neste trabalho com a melhoria *2-opt* são melhores do que os resultados das outras técnicas, com exceção do problema lin105. Os erros médios

variam entre 0% e 3,31%, conforme mostra a última coluna da Tabela 6.3. Sem a técnica de melhoria, os resultados para o problema *eil76* são melhores do que os resultados das demais técnicas. A figura 6.3 mostra os resultados da aplicação da técnica proposta para o problema *pcb442* com e sem a técnica *2-opt*.

Para o problema assimétrico, as técnicas utilizadas como comparação da técnica proposta neste trabalho são as seguintes (GLOVER et al., 2001): os métodos Caminho de Karp-Steele (*KSP*) e Karp-Steele geral (*GKS*); a Contração Recursiva de Caminhos (*RPC*); a heurística de Contração ou Determinação de Caminhos (*COP*); a heurística “gulosa” (*GR*); e a heurística de Inserção Aleatória (*RI*). Na tabela 6.4, são mostrados os erros médios das técnicas descritas acima, além da técnica “pura” proposta neste trabalho, e da técnica proposta com técnica de melhoria *2-opt*.

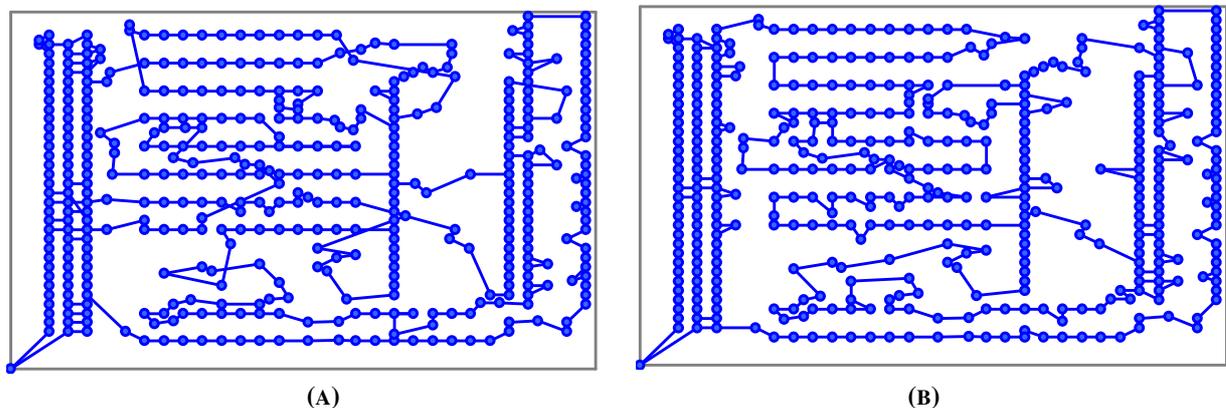


FIGURA 6.3 - SOLUÇÕES FACTÍVEIS ENCONTRADAS PARA O PCB442 ATRAVÉS: (A) DO MÉTODO PROPOSTO PURO, COM ERRO MÉDIO DE 9,16%; (B) DO MÉTODO PROPOSTO COM A TÉCNICA 2-OPT, COM ERRO MÉDIO DE 2,87%.

Os resultados da técnica “pura” proposta neste trabalho são melhores ou equivalentes aos resultados das demais heurísticas citadas acima para os problemas *br17*, *ftv33*, *ftv44*, *ft53*, *ft70* e *kro124p*, conforme mostra a Tabela 6.4. Com a aplicação da técnica *2-opt* na técnica proposta, os resultados encontrados são melhores para os problemas *br17*, *ftv33*, *pr43*, *ry48p*, *ftv44*, *ft53*, *ft70* e *kro124p*, com erros médios variando entre 0% e 16,14%.

Nas figuras 6.4 e 6.5 encontram-se as visualizações do *software* desenvolvido neste trabalho para o problema do Caixeiro Viajante para os problemas simétricos *bier127* e *att532*, respectivamente. Grafos de outros problemas simétricos do *TSPLIB* encontram-se no Apêndice I deste trabalho.

TABELA 6.4 – RESULTADOS DOS EXPERIMENTOS PARA O PCV ASSIMÉTRICO COM AS TÉCNICAS: *GR*, *RI*, *KSP*, *GKS*, *RPC*, *COP* E RNRW COM WTA. AS SOLUÇÕES EM NEGRITO MOSTRAM OS MELHORES RESULTADOS PARA CADA PROBLEMA, DESCONSIDERANDO-SE OS RESULTADOS DA RNRW COM A TÉCNICA 2-OPT.

nome do PCV	número de cidades	solução ótima	erro médio para 5 algoritmos (%)						erro médio (%) RNRW com WTA			
			<i>GR</i>	<i>RI</i>	<i>KSP</i>	<i>GKS</i>	<i>RPC</i>	<i>COP</i>	max	min	<i>2opt</i>	
br17	17	39	102,56	0	0	0	0	0	0	0	0	0
ftv33	33	1286	31,34	11,82	13,14	8,09	21,62	9,49	7,00	0	0	0
ftv35	35	1473	24,37	9,37	1,56	1,09	21,18	1,56	5,70	3,12	3,12	3,12
ftv38	38	1530	14,84	10,20	1,50	1,05	25,69	3,59	3,79	3,73	3,01	3,01
pr43	43	5620	3,59	0,30	0,11	0,32	0,66	0,68	0,46	0,29	0,05	0,05
ftv44	44	1613	18,78	14,07	7,69	5,33	22,26	10,66	2,60	2,60	2,60	2,60
ftv47	47	1776	11,88	12,16	3,04	1,69	28,72	8,73	8,05	3,83	3,83	3,83
ry48p	48	14422	32,55	11,66	7,23	4,52	29,50	7,97	6,39	5,59	1,24	1,24
ft53	53	6905	80,84	24,82	12,99	12,31	18,64	15,68	3,23	2,65	2,65	2,65
ftv55	55	1608	25,93	15,30	3,05	3,05	33,27	4,79	12,19	11,19	6,03	6,03
ftv64	64	1839	25,77	18,49	3,81	2,61	29,09	1,96	2,50	2,50	2,50	2,50
ft70	70	38673	14,84	9,32	1,88	2,84	5,89	1,90	2,43	1,74	1,74	1,74
ftv70	70	1950	31,85	16,15	3,33	2,87	22,77	1,85	8,87	8,77	8,56	8,56
kro124p	100	36230	21,01	12,17	16,95	8,69	23,06	8,79	10,52	7,66	7,66	7,66
ftv170	170	2755	32,05	28,97	2,40	1,38	25,66	3,59	14,66	12,16	12,16	12,16
rbg323	323	1326	8,52	29,34	0	0	0,53	0	16,44	16,14	16,14	16,14
rbg358	358	1163	7,74	42,48	0	0	2,32	0,26	22,01	12,73	8,17	8,17
rbg403	403	2465	0,85	9,17	0	0	0,69	0,20	4,71	4,71	4,71	4,71
rbg443	443	2720	0,92	10,48	0	0	0	0	8,05	8,05	2,17	2,17

Nas tabelas 6.5 e 6.6 encontram-se os resultados detalhados de todos os problemas simétricos e assimétricos utilizados neste trabalho, respectivamente. Os tempos computacionais colocados nestas tabelas indicam o desempenho da técnica proposta neste trabalho em segundos, e foram obtidos em um computador *Pentium IV*, 1,6GHz, com 256Mb de memória RAM.

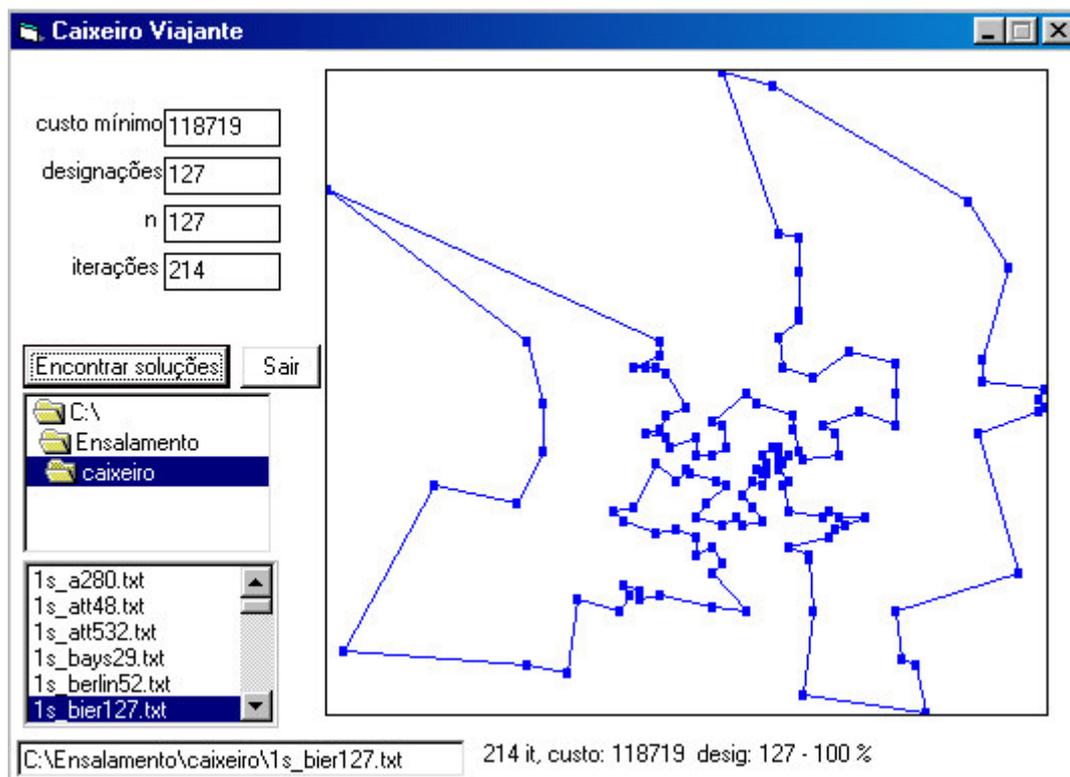
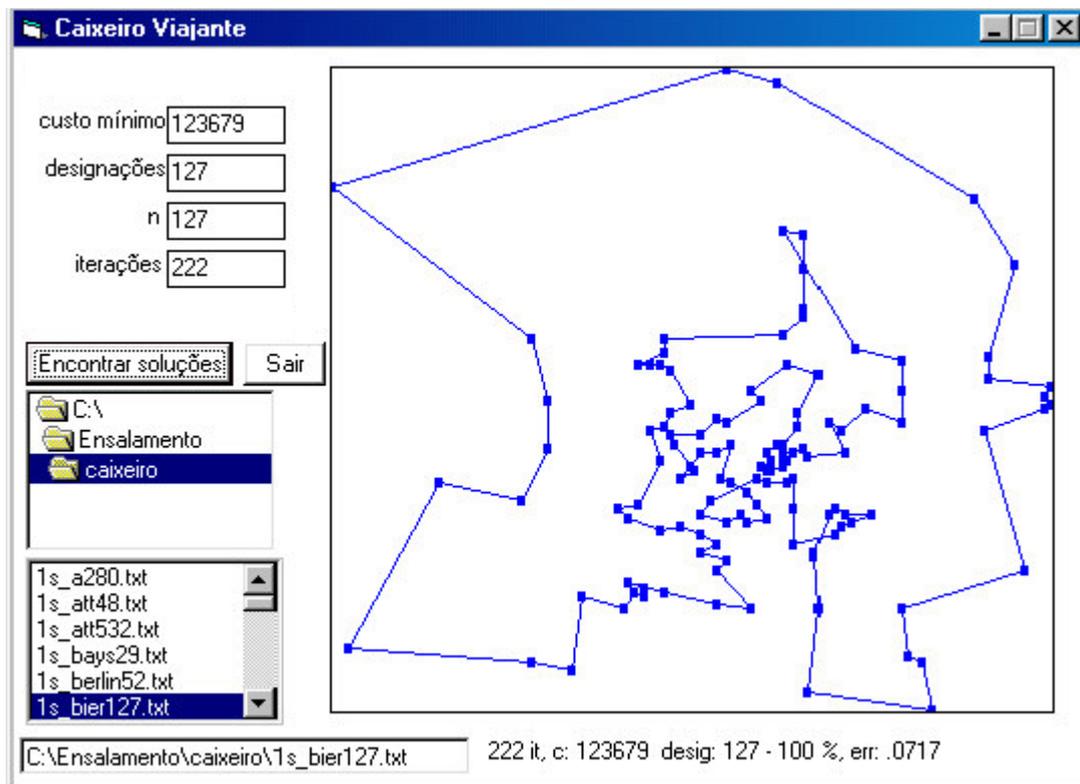


FIGURA 6.4 – RESULTADOS DO PROBLEMA BIER127 COM A TÉCNICA “PURA” PROPOSTA NESTE TRABALHO E COM A TÉCNICA DE MELHORIA 2-OPT

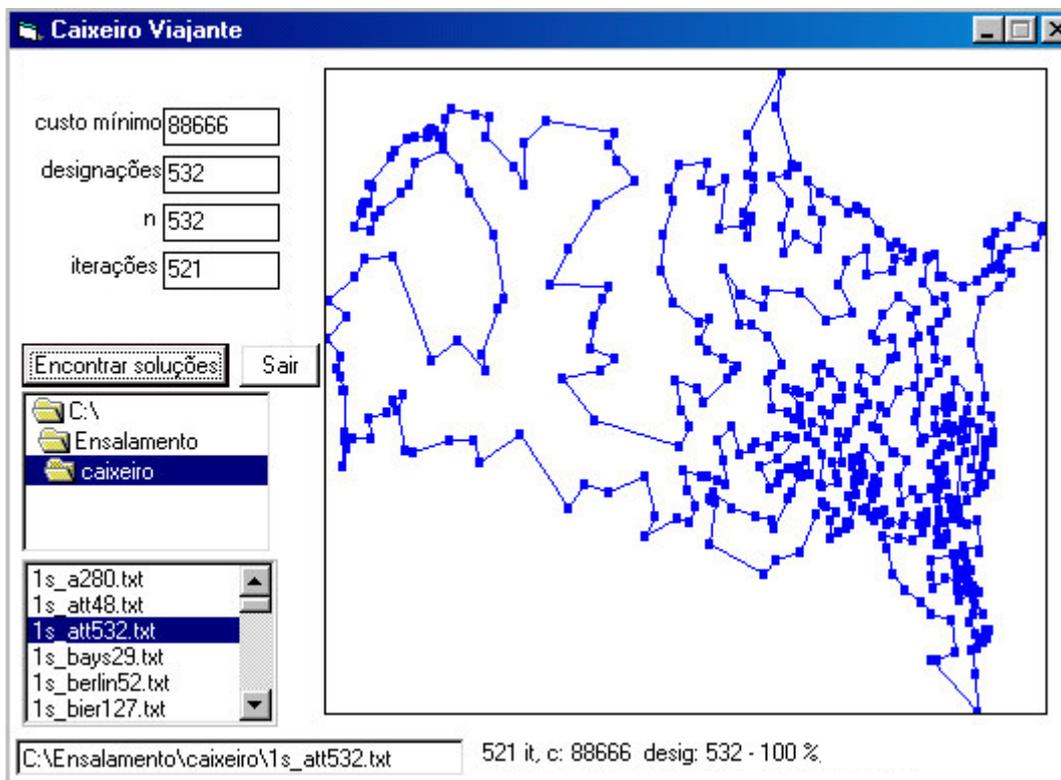
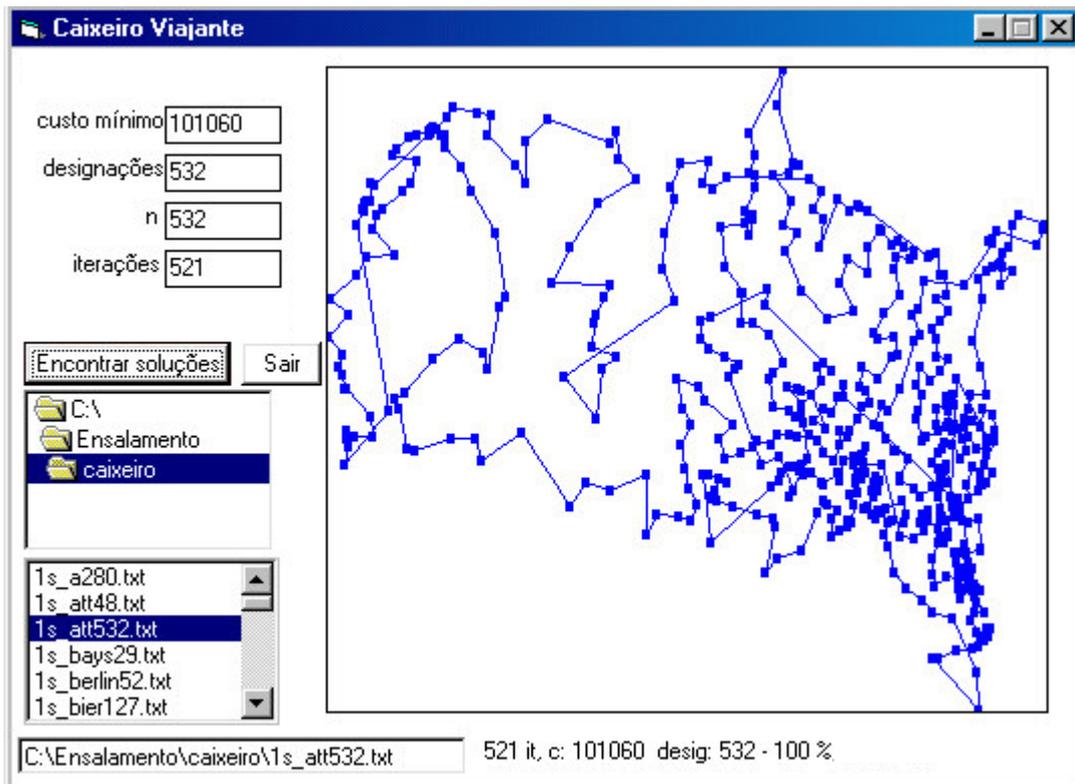


FIGURA 6.5 – RESULTADOS DO PROBLEMA ATT532 COM A TÉCNICA “PURA” PROPOSTA NESTE TRABALHO E COM A TÉCNICA DE MELHORIA 2-OPT

TABELA 6.5 – RESULTADOS PARA TODOS OS PROBLEMAS SIMÉTRICOS DO *TSPLIB* QUE FORAM UTILIZADOS NESTE TRABALHO

nome do PCV	n	solução ótima	pior solução	maior erro (%)	melhor solução	menor erro (%)	solução 2-opt	erro 2-opt (%)	tempo RNRW (seg)	tempo RNRW +2-opt (seg)
laurene10	10	2,71	2,84	4,797	2,71	0	2,71	0	0	0
burma14	14	30	31	3,333	30	0	30	0	0	0
ulysses16	16	74,1	75,5	1,889	74,8	0,945	74,1	0	0	0
ulysses22	22	75,6	79	4,497	75,6	0	75,6	0	1	1
bays29	29	2020	2089	3,416	2048	1,386	2020	0	1	1
swiss42	42	1273	1302	2,278	1279	0,471	1273	0	2	3
att48	48	33522	34860	3,991	34367	2,521	33522	0	4	4
eil51	51	430	439	2,093	435	1,163	430	0	5	5
berlin52	52	7542	8154	8,115	7657	1,525	7542	0	6	6
st70	70	678,6	706	4,038	697	2,711	678,6	0	17	17
eil76	76	545,4	548	0,477	548	0,477	545,4	0	15	16
pr76	76	108159	111329	2,931	110768	2,412	108159	0	10	14
gr96	96	514	536	4,280	527	2,529	514	0	30	35
kroC100	100	20749	21830	5,210	21771	4,926	20749	0	40	45
kroD100	100	21294	23327	9,547	23039	8,195	21440	0,686	35	43
rd100	100	7910	8477	7,168	8450	6,827	7916	0,076	38	40
eil101	101	629	679	7,949	648	3,021	632	0,477	40	42
lin105	105	14383	15238	5,945	15006	4,332	14412	0,202	40	45
pr107	107	44303	45694	3,140	45612	2,955	44303	0	45	50
pr124	124	59030	60583	2,631	59227	0,334	59030	0	50	70
bier127	127	118282	124214	5,015	123272	4,219	118719	0,369	51	67
ch130	130	6110	6434	5,303	6418	5,041	6195	1,391	54	64
pr136	136	96772	103409	6,858	102564	5,985	97947	1,214	70	75
pr152	152	73682	76093	3,272	76065	3,234	73682	0	84	92
u159	159	42080	45106	7,191	44744	6,331	42080	0	165	180
rat195	195	2323	2478	6,672	2452	5,553	2400	3,315	460	462
d198	198	15780	17516	11,001	17425	10,425	15972	1,217	425	452
kroA200	200	29368	32965	12,248	31995	8,945	29551	0,623	464	472
gr202	202	490	513	4,694	502	2,449	490	0	480	486
tsp225	225	3916	4247	8,453	4154	6,078	3957	1,047	521	533
a280	280	2586	2906	12,374	2900	12,142	2664	3,016	1191	1215
pr299	299	48191	54034	12,125	54034	12,125	49744	3,223	1600	1695
lin318	318	42029	45663	8,646	45538	8,349	42829	1,903	1645	1761
rd400	400	15281	17618	15,294	17268	13,003	15772	3,213	4924	5014
pcb442	442	50783,6	57475	13,176	55436	9,161	52240	2,868	5915	6061
att532	532	87550	101060	15,431	100317	14,583	88666	1,275	14250	14440

Nas figuras 6.6 e 6.7 encontram-se os gráficos com os melhores e piores resultados da técnica ‘pura’ aplicada neste trabalho para os problemas simétricos e assimétricos mostrados nas Tabelas 6.5 e 6.6, além dos resultados da técnica proposta com a melhoria *2-opt*.

TABELA 6.6 – RESULTADOS PARA TODOS OS PROBLEMAS ASSIMÉTRICOS DO *TSPLIB* QUE FORAM UTILIZADOS NESTE TRABALHO

nome do PCV	n	solução ótima	pior solução	maior erro (%)	melhor solução	menor erro (%)	solução 2-opt	erro 2-opt (%)	tempo RNRW (seg)	tempo RNRW +2-opt (seg)
katta06	6	63	64	1,587	63	0	63	0	0	0
br17	17	39	40	2,564	39	0	39	0	0	1
ftv33	33	1286	1358	5,599	1336	3,888	1286	0	1	2
ftv35	35	1473	1557	5,703	1519	3,123	1512	2,648	2	2
ftv38	38	1530	1588	3,791	1587	3,725	1576	3,007	3	3
pr43	43	5620	5646	0,463	5636	0,285	5622	0,036	2	3
ftv44	44	1613	1682	4,278	1655	2,604	1655	2,604	3	3
ftv47	47	1776	1919	8,052	1844	3,829	1844	3,829	3	4
ry48p	48	14422	15343	6,386	15228	5,589	14787	2,531	3	4
ft53	53	6905	7128	3,230	7088	2,650	7088	2,650	8	9
ftv55	55	1608	1802	12,065	1683	4,664	1683	4,664	5	6
ftv64	64	1839	1885	2,501	1885	2,501	1885	2,501	8	10
ft70	70	38673	40226	4,016	39345	1,738	39345	1,738	13	15
ftv70	70	1950	2123	8,872	2049	5,077	2049	5,077	10	13
kro124p	100	36230	39960	10,295	39005	7,659	39005	7,659	35	43
ftv170	170	2755	3159	14,664	3090	12,160	3090	12,160	285	324
rgb323	323	1326	1544	16,440	1540	16,139	1540	16,139	1702	1865
rgb358	358	1163	1419	22,012	1311	12,726	1258	8,169	2914	3008
rgb403	403	2465	2581	4,706	2581	4,706	2492	1,095	3987	4125
rgb443	443	2720	2939	8,051	2939	8,051	2779	2,169	6204	6442

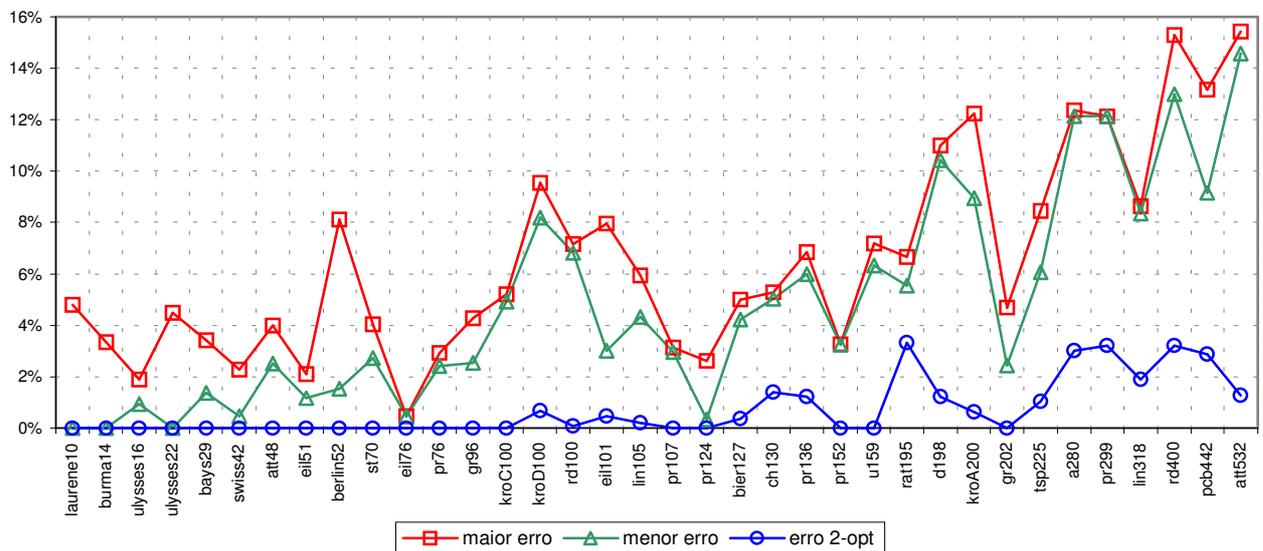


FIGURA 6.6 – ERROS MÉDIOS ENCONTRADOS NOS PROBLEMAS SIMÉTRICOS DO CAIXEIRO VIAJANTE

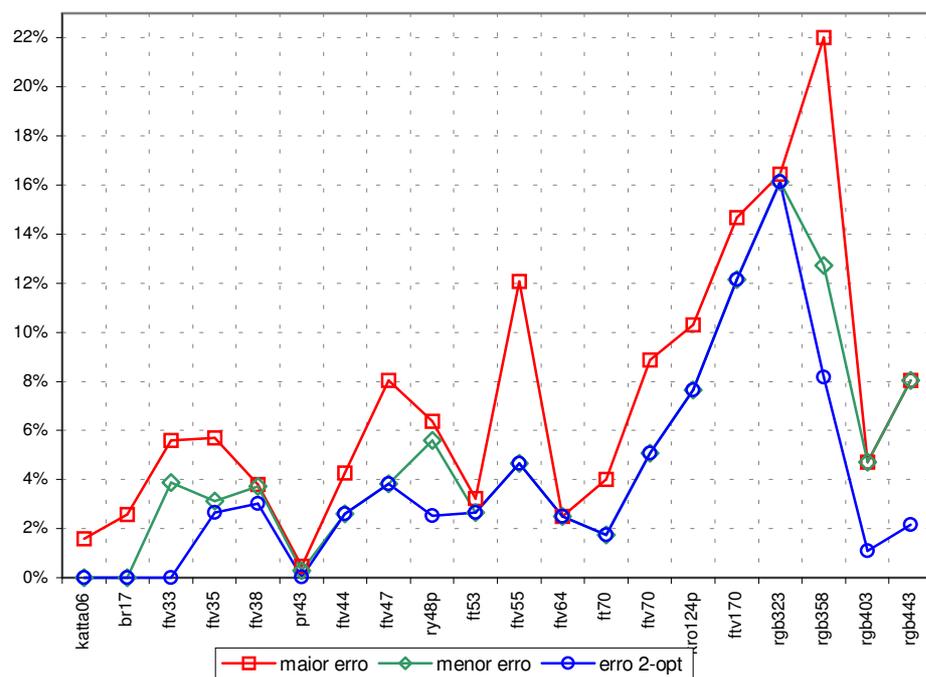


FIGURA 6.7 – ERROS MÉDIOS ENCONTRADOS NOS PROBLEMAS ASSIMÉTRICOS DO CAIXEIRO VIAJANTE

Os erros apresentados nas Figuras 6.6 e 6.7 mostram resultados satisfatórios para os problemas utilizados como teste para a técnica proposta neste trabalho, principalmente quando o algoritmo de melhoria *2-opt* é utilizado, onde os erros médios são todos menores do que 4% nos problemas simétricos e 16% nos problemas assimétricos.

Foram escolhidos 7 problemas simétricos (gr96, kroC100, kroD100, eil101, lin105, rd100 e pr107) que possuem tamanhos similares para mostrar a melhoria da qualidade das soluções encontradas com a técnica proposta de acordo com o número de iterações utilizadas. A técnica *Winner Takes All* começa a ser aplicada a partir da 110^a iteração para quase todos estes problemas, e a visualização da evolução dos erros médios a partir desta iteração é mostrada nas Figuras 6.8 e 6.9, onde aparecem os erros médios da técnica “pura” e da técnica com o algoritmo de melhoria *2-opt*.

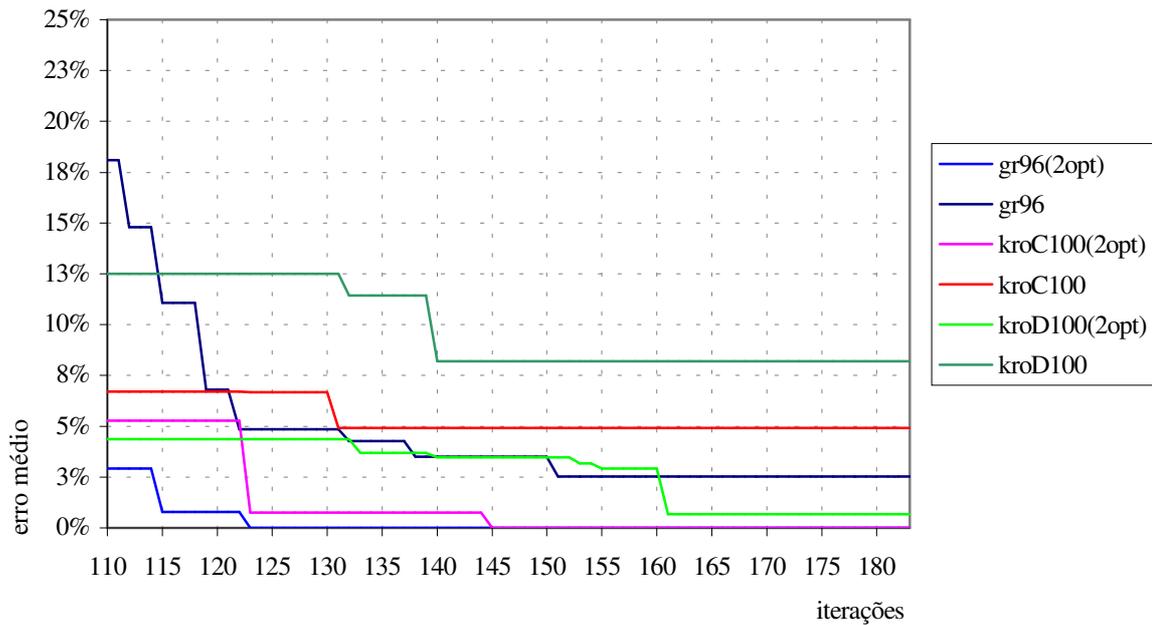


FIGURA 6.8 – QUALIDADE DAS SOLUÇÕES DE ACORDO COM O NÚMERO DE ITERAÇÕES PARA OS PROBLEMAS GR96, KROC100 E KROD100, COM E SEM A TÉCNICA DE MELHORIA 2-OPT.

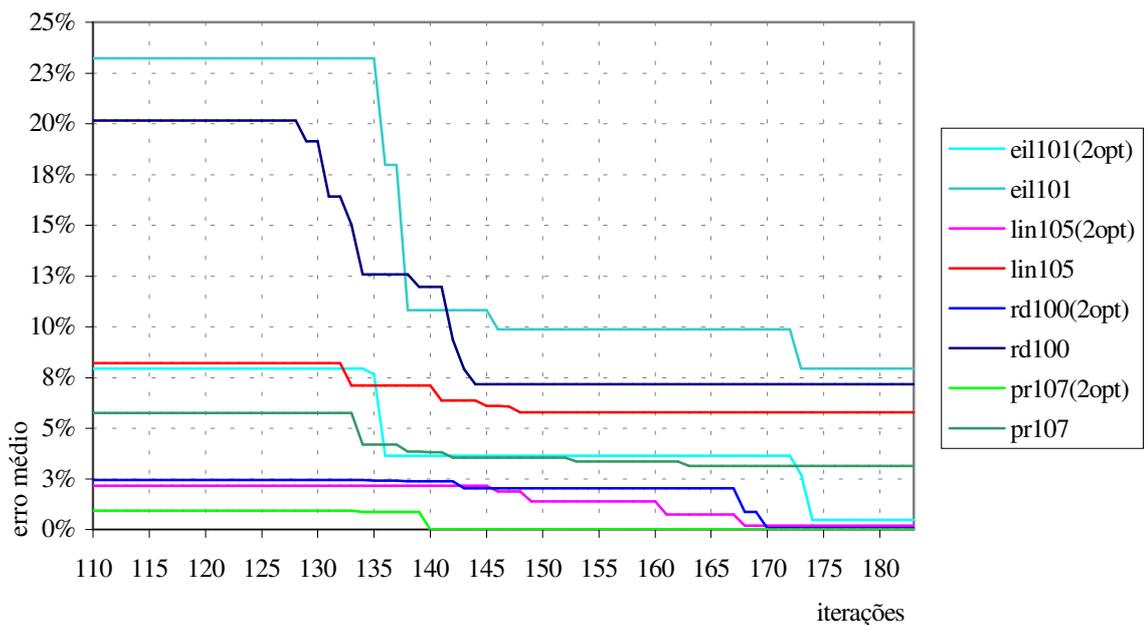


FIGURA 6.9 – QUALIDADE DAS SOLUÇÕES DE ACORDO COM O NÚMERO DE ITERAÇÕES PARA OS PROBLEMAS EIL101, LIN105, RD100 E PR107, COM E SEM A TÉCNICA DE MELHORIA 2-OPT.

De acordo com os resultados encontrados, a partir da 180ª iteração as soluções dos problemas indicados nas figuras 6.8 e 6.9 não apresentam melhorias, encontrando-se boas aproximações para as soluções ótimas globais conhecidas destes problemas, conforme mostram as tabelas de resultados apresentadas nesta seção.

No próximo capítulo são apresentadas as conclusões deste trabalho e as sugestões para futuros trabalhos.

CAPÍTULO VII

7 CONCLUSÕES E TRABALHOS FUTUROS

7.1 RESULTADOS

Neste trabalho, uma técnica baseada no princípio *Winner Takes All* foi aplicada na Rede Neural de Wang para a resolução dos problemas da Designação e do Caixeiro Viajante. A eficiência desta técnica foi comprovada com os seguintes resultados:

- em 73 matrizes com custos aleatórios, com dimensões variando entre 3×3 e 20×20 , para o problema da Designação, com a obtenção de soluções ótimas globais em mais de 61% dos casos testados e soluções ótimas locais com erros médios inferiores a 6%;
- na implementação desta metodologia, com a determinação dos custos para o problema da Designação através dos mapas Auto-Organizáveis de Kohonen, no problema de Alocação de salas de aula da UFPR, com resultados satisfatórios; e
- na aplicação desta metodologia com o princípio *Winner Takes All* modificado para resolver o problema do Caixeiro Viajante, com boas aproximações para 35 problemas simétricos e 20 assimétricos do *TSPLIB*.

De acordo com os resultados mostrados na seção 5.2, comprova-se que a técnica de ajuste baseada no princípio *Winner Takes All*, aplicada à Rede Neural Recorrente de Wang produz resultados satisfatórios para o problema da Designação, principalmente em matrizes com múltiplas soluções ótimas, onde a melhor escolha dos parâmetros determina erros médios inferiores a 0,5% e a soluções ótimas globais para 95% das matrizes testadas.

A adequada utilização dos parâmetros para a Rede Neural de Wang é essencial, e com as propostas mostradas no Capítulo III deste trabalho para a obtenção destes parâmetros, os resultados mostram a eficiência desta técnica para problemas de grande escala, pois o número

médio de iterações necessário para encontrar soluções factíveis para o problema da Designação, é reduzido consideravelmente comparando-se com a Rede Neural original de Wang.

A aplicação da Rede de Kohonen na determinação dos elementos da matriz de custos mostra resultados satisfatórios para mapas com dimensão maior do que 15×15 para os dados utilizados na alocação de salas de aula. As soluções factíveis encontradas com estes mapas possuem erro médio inferior a 3%, o que mostra a eficiência da metodologia proposta neste trabalho para a utilização das duas Redes Neurais para resolver o problema da Designação.

A implementação do princípio *Winner Takes All* modificado para resolver o problema do Caixeiro Viajante mostra mais uma aplicação da metodologia proposta, onde a Rede Neural de Wang é utilizada com os parâmetros que definem melhores resultados para os testes feitos para o problema da Designação, conforme mostram os resultados apresentados no Capítulo V.

Os problemas utilizados para a aplicação da técnica proposta para o problema do Caixeiro Viajante são do banco de dados *TSPLIB*, e tratam-se de problemas utilizados para comparação de desempenhos de diversas heurísticas. As comparações feitas com outras heurísticas mostram que a técnica proposta neste trabalho obtém melhores resultados em diversos problemas deste banco de dados, com erros médios abaixo de 16,14%.

Vale lembrar que melhorias consideráveis foram obtidas quando a técnica *2-opt* foi utilizada nas soluções apresentadas pela técnica proposta neste trabalho, conforme mostram os resultados apresentados no Capítulo VI.

Uma grande vantagem da implementação da técnica mostrada neste trabalho é a possibilidade de resolver tanto problemas do Caixeiro Viajante simétricos quanto assimétricos.

7.2 SUGESTÕES PARA FUTUROS TRABALHOS

Algumas sugestões para futuros trabalhos são as seguintes:

- Definir novos parâmetros para a Rede Neural de Wang, utilizando outras medidas de comparação entre os elementos da matriz de custos do problema da Designação;
- Aplicar a metodologia proposta para o problema da Designação (Rede de Kohonen para os custos e Rede de Wang com *Winner Takes All* para a resolução do

problema) em outros problemas reais, tais como determinação de tarefas para funcionários, horários para professores, escalonamento de tarefas, e problemas semelhantes;

- Aplicar outras heurísticas de melhoria para as soluções da técnica proposta para o problema do Caixeiro Viajante, tais como *3-opt*, *1-shift* ou outra técnica similar;
- Aplicar a metodologia proposta para o problema do Caixeiro Viajante em outros problemas reais, tais como roteamento de veículos, problemas de perfuração de placas e similares;
- Criar outras formas de definição do princípio *Winner Takes All*, comparando os resultados com os problemas do *TSPLIB*. Neste trabalho este princípio foi utilizado com o critério do maior elemento de cada linha/coluna da matriz de decisão; uma outra forma de implementar esta técnica é a aplicação do critério de Vogel para determinar o vencedor de cada linha/coluna da matriz de decisão.

APÊNDICE I

GRAFOS DE ALGUNS PROBLEMAS DO *TSPLIB* OBTIDOS COM A TÉCNICA PROPOSTA NESTE TRABALHO

Seguem abaixo os grafos obtidos pela técnica “pura” proposta neste trabalho, e a técnica de melhoria *2-opt*.

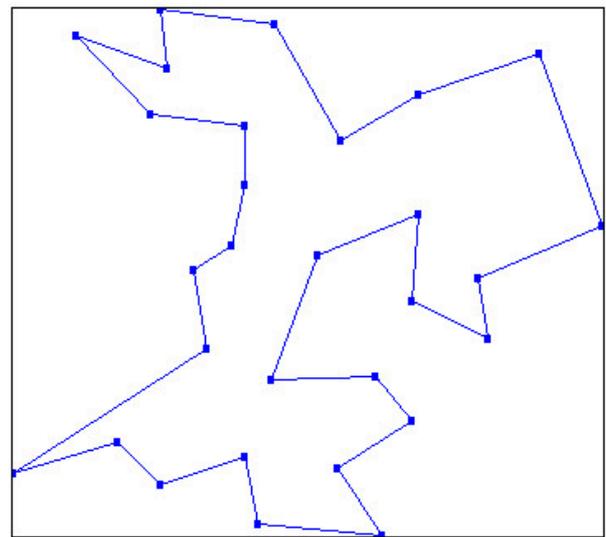
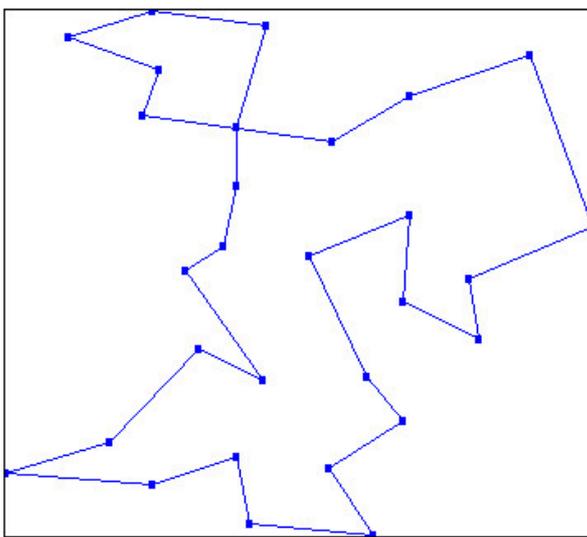


FIGURA A1 – SOLUÇÃO PARA O PROBLEMA BAYS29 COM A TÉCNICA “PURA” (ERRO DE 3,41%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2-OPT (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA)

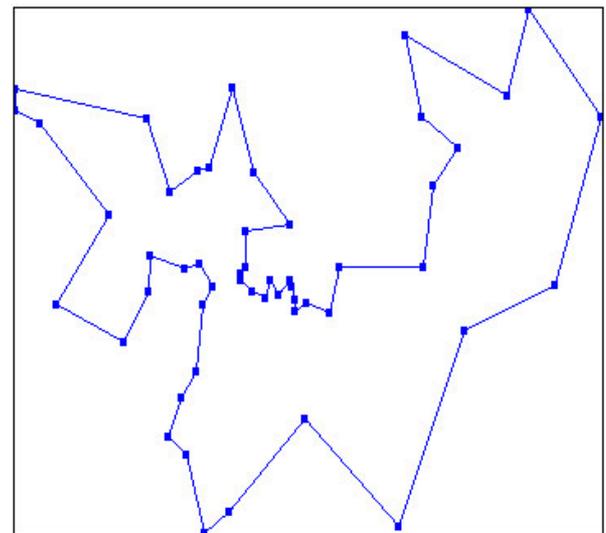
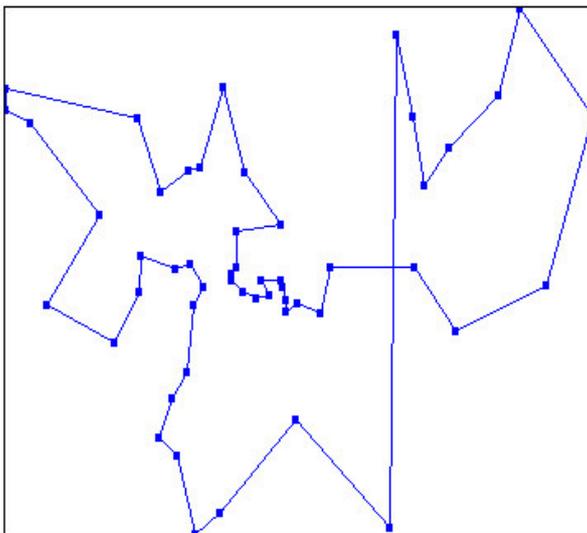


FIGURA A2 – SOLUÇÃO PARA O PROBLEMA BERLIN52 COM A TÉCNICA “PURA” (ERRO DE 6,96%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2-OPT (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA)

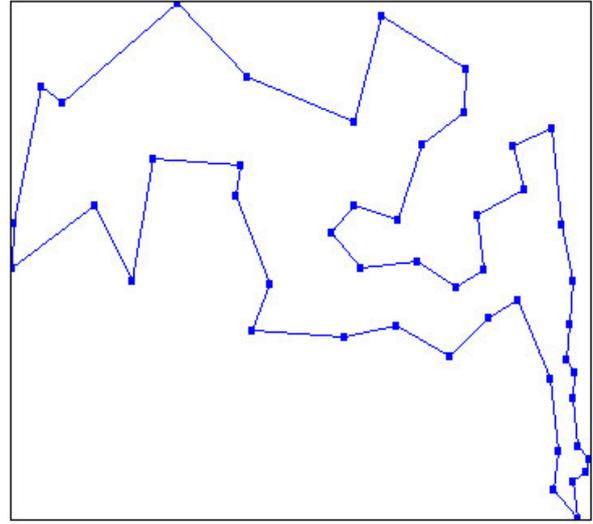
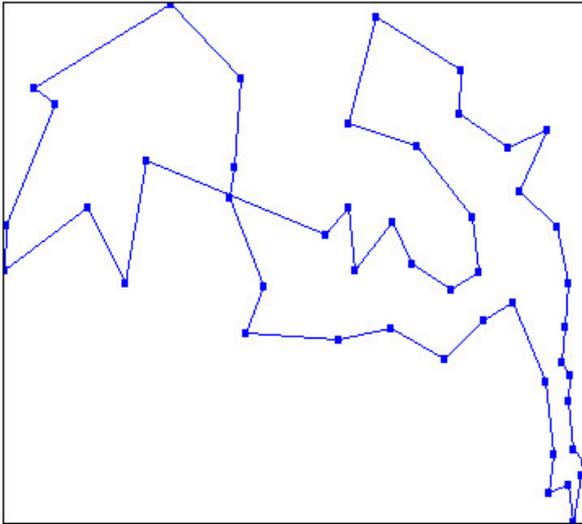


FIGURA A3 – SOLUÇÃO PARA O PROBLEMA ATT48 COM A TÉCNICA “PURA” (ERRO DE 3,99%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2-OPT (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA)

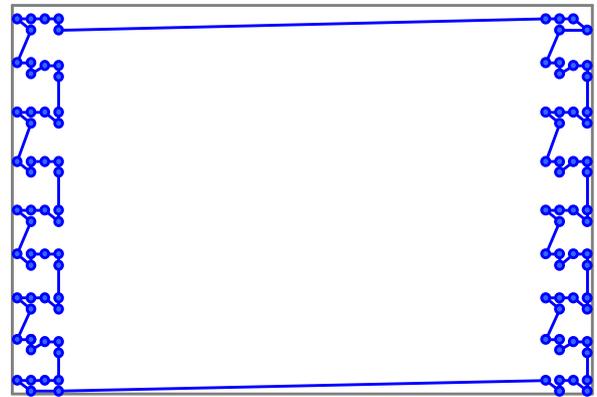
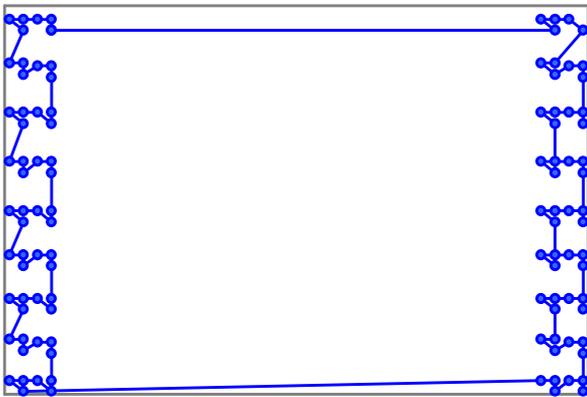


FIGURA A4 – SOLUÇÃO PARA O PROBLEMA PR107 COM A TÉCNICA “PURA” (ERRO DE 3,1%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2-OPT (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA)

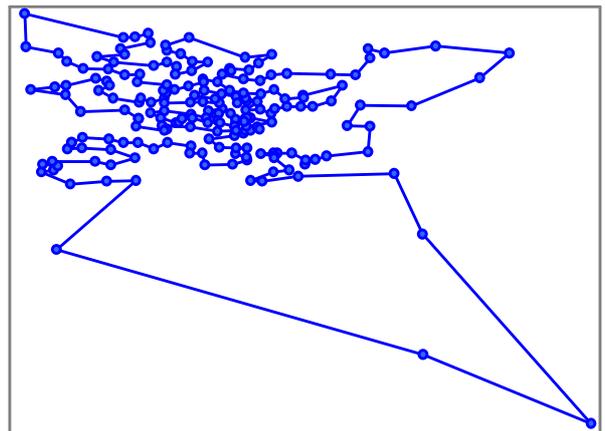
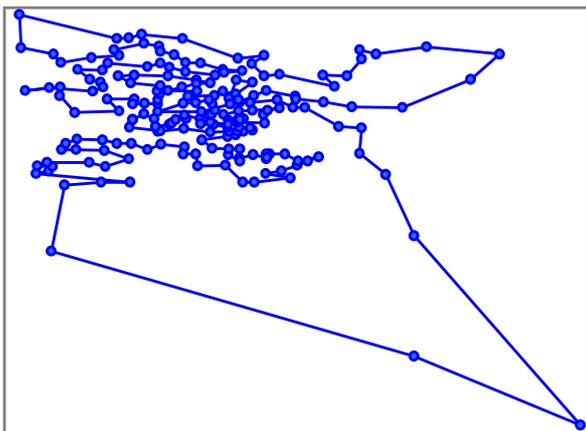


FIGURA A5 – SOLUÇÃO PARA O PROBLEMA GR202 COM A TÉCNICA “PURA” (ERRO DE 4,7%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2-OPT (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA)

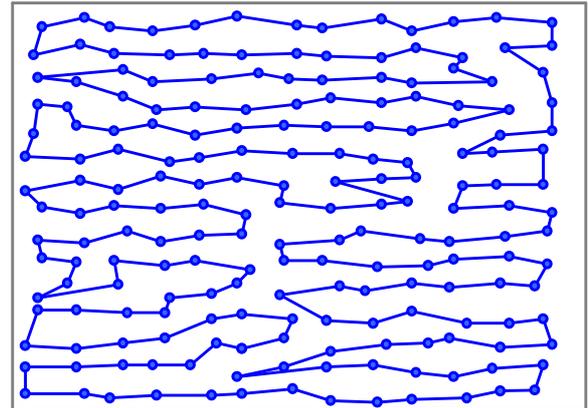
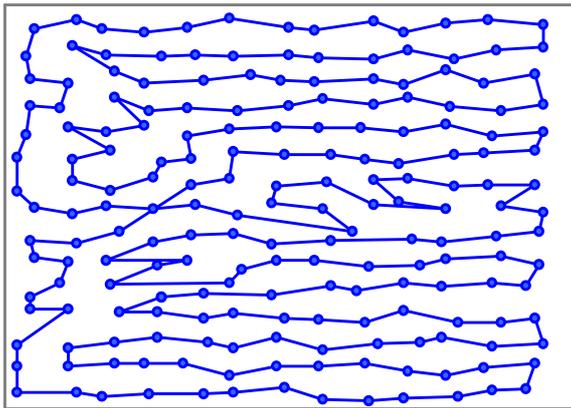


FIGURA A6 – SOLUÇÃO PARA O PROBLEMA RAT195 COM A TÉCNICA “PURA” (ERRO DE 6,67%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2-OPT (ERRO DE 3,31%)

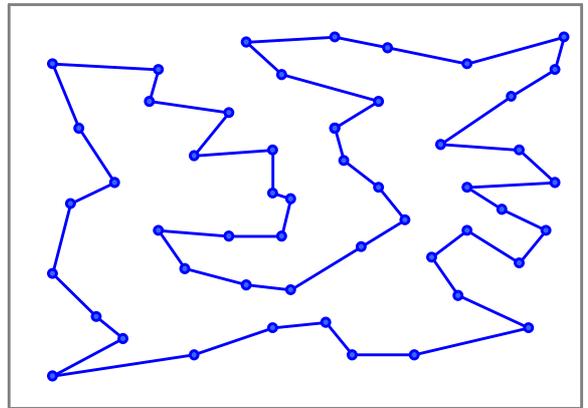
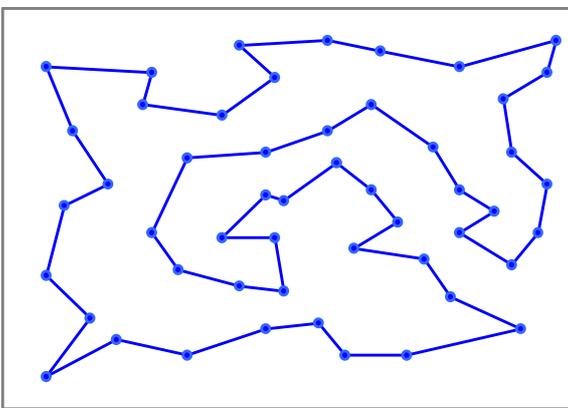


FIGURA A7 – SOLUÇÃO PARA O PROBLEMA EIL51 COM A TÉCNICA “PURA” (ERRO DE 1,38%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2-OPT (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA)

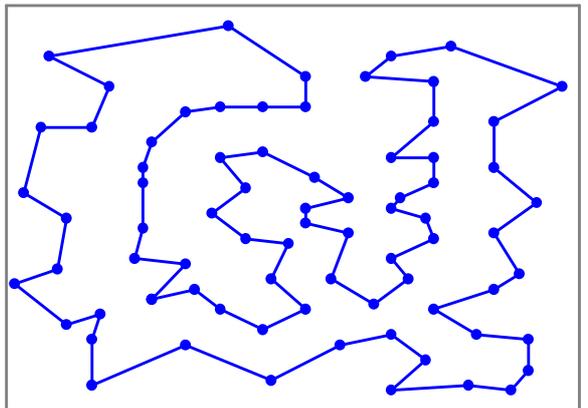
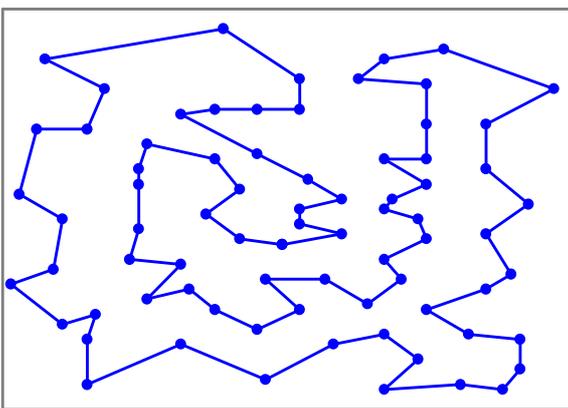


FIGURA A8 – SOLUÇÃO PARA O PROBLEMA EIL76 COM A TÉCNICA “PURA” (ERRO DE 0,48%), E COM A TÉCNICA QUE UTILIZA A MELHORIA 2-OPT (SOLUÇÃO ÓTIMA GLOBAL ENCONTRADA)

REFERÊNCIAS

- AFFENZELLER, M.; WANGER, S., **A Self-Adaptive Model for Selective Pressure Handling within the Theory of Genetic Algorithms**. In: Computer Aided Systems Theory, 2003, Las Palmas de Gran Canaria, Spain: Proceedings of Computer Aided Systems Theory - EUROCAST 2003, Springer Verlag: Lecture Notes in Computer Science, v. 2809, n. 1, p. 384-393.
- AHUJA, R.K.; MANGNANTI, T.L.; ORLIN, J.B., **Network Flows: Theory, Algorithms and Applications**. New Jersey: Prentice Hall, 1993.
- ARAS, N.; OOMMEN, B.J.; ALTINEL, I.K., **The Kohonen network incorporating explicit statistics and its application to the traveling salesman problem**. Neural Networks, v. 12, n. 9, p. 1273-1284, nov. 1999.
- AUFFERMANN, W.F.; NGAN, S.C.; HU, X., **Cluster significance testing using the Bootstrap**. NeuroImage, v. 17, n. 2, p. 583-591, out. 2002.
- BAZARAA, M.S.; JARVIS, J.J.; SHERALI, H.D., **Linear Programming and Network Flows**. 2nd ed. New York: Wiley, 1990.
- BECCALI, M.; CELLURA, M.; LO BRANO, V.; MARVUGLIA, A., **Forecasting daily urban electric load profiles using Artificial Neural Networks**. Energy Conversion and Management, v. 45, n. 18-19, p. 2879-2900, nov. 2004.
- BELTZER, A.I.; SATO, T., **Neural classification of finite elements**. Computers and Structures, v. 81, n. 24-25, p. 2331-2335, set. 2003.
- BIANCHI, L.; KNOWLES, J.; BOWLER, J., **Local search for the probabilistic traveling salesman problem: Correction to the 2-p-opt and 1-shift algorithms**. European Journal of Operational Research, v. 162, n. 1, p. 206-219, abr. 2005.
- BONABEAU, E.; HÉNAUX, F., **Self organizing maps for drawing large graphs**. Information Processing Letters, v. 67, n. 4, p. 177-184, ago. 1998.
- BROSSE, S.; GIRAUDEL, J.L.; LEK, S., **Utilization of non-supervised neural networks and principal component analysis to study fish assemblages**. Ecological Modelling, v. 146, n. 1-3, p. 159-166, dez. 2001.

BUDINICH, M., **A Self-Organizing Neural Network for the Traveling Salesman problem that is competitive with Simulated Annealing.** Neural Computing, v. 8, n. 2, p. 416-424, fev. 1996.

BURKE, E.K.; COWLING, P.I.; KEUTHEN, R., **Effective Local and Guided variable neighborhood Search Methods for the Asymmetric Traveling Salesman Problem.** In: Evolutionary Computing Workshop, 2001, Como, Italy: Applications of Evolutionary Computing - EvoWorkshop 2001, Springer Verlag: Lecture Notes on Computer Science, v. 2037, n. 1, p. 203-212.

CARRASCO, M.P.; PATO, M.V., **A comparison of discrete and continuous neural network approaches to solve the class/teacher timetabling problem.** European Journal of Operational Research, v. 153, n. 1, p. 65-79, fev. 2004.

CHOI, I.C.; KIM, S.I.; KIM, H.S., **A genetic algorithm with a mixed region search for the asymmetric traveling salesman problem.** Computers & Operations Research, v. 30, n. 5, p. 773-786, abr. 2003.

CHRISTOFIDES, N., **Graph Theory: an algorithmic approach.** Academic Press: New York, 1975.

CHU, P.C.; BEASLEY, J.E., **A genetic algorithm for the generalized assignment problem.** Computers and Operational Research, v. 24, n. 1, p. 17-23, jan. 1997.

CHU, S.C.; RODDICK, J.F.; PAN, J.S., **Ant colony system with communication strategies.** Information Sciences, v. 167, n. 1-4, p. 63-76, dez. 2004.

COCHRANE, E.M., BEASLEY, J.E., **The Co-Adaptive Neural Network Approach to the Euclidean Travelling Salesman Problem.** Neural Networks, v. 16, n. 10, p. 1499-1525, dez. 2003.

COPPINI, G.; DICIOTTI, S.; VALLI, G., **Matching of medical images by self-organizing neural networks.** Pattern Recognition Letters, v. 25, n. 3, p. 341-352, fev. 2004.

COTTRELL, M., FORT, J.C., PAGÈS, G., **Theoretical aspects of the SOM algorithm.** Neurocomputing, v. 21, n. 1-3, p. 119-138, nov. 1998.

DELGADO, M.L.P; MARTÍN, Q.M., **Diseño de una red neuronal autoasociativa para resolver el problema de viajante de comercio (Traveling Salesman Problem, TSP).** In: Congreso Nacional de Estadística e Investigación Operativa, 27., 2003, Lleida: Proceedings of 27º Congreso Nacional de Estadística e Investigación Operativa.

DORIGO, M.; GARAMBARDELLA, L.M., **Ant Colonies for the Traveling Salesman Problem**. Biosystems, v. 43, n. 2, p. 73-81, jul. 1997a.

DORIGO, M.; GARAMBARDELLA, L.M., **Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem**. IEEE Transactions on Evolutionary Computation, v. 1, n. 1, p. 53-66, abr. 1997b.

FAUSETT, L., **Fundamentals of Neural Networks**. New Jersey: Prentice Hall, 1994.

FENG, G.; DOULIGERIS, C., **Using Hopfield Networks to solve Traveling Salesman Problems based on Stable State Analysis Technique**. In: International Joint Conference on Neural Networks, 2000, Como, Italy: Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks - IJCNN 2000, IEEE press, v. 6, p. 521-526.

FLANAGAN, J. A., **Self-organization in Kohonen SOM's**. Neural Networks, v. 9, n. 7, p. 1185-1197, out. 1996.

FLECK, E.; SANTOS, R.A.; PEDREIRA, C.E.; SOUZA, E.M., **Modelagem de agrupamento de dados sísmicos usando mapas de Kohonen**. In: International Congress of The Brazilian Geophysical Society, 8., 2003, Rio de Janeiro: Proceedings of the Eighth International Congress of The Brazilian Geophysical Society.

GENDREAU, M.; LAPORTE, G.; SEMET, F., **A tabu search heuristic for the undirected selective traveling salesman problem**. European Journal of Operational Research, v. 106, n. 2-3, p. 539-545, abr. 1998.

GHAZIRI, H.; OSMAN, I.H., **A neural network algorithm for the traveling salesman problem with backhauls**. Computers and Industrial Engineering, v. 44 n. 2, p. 267-281, fev. 2003.

GLOVER, F.; GUTIN, G.; YEO, A.; ZVEROVICH, A., **Construction heuristics for the asymmetric TSP**. European Journal of Operational Research, v. 129, n. 3, p. 555-568, mar. 2001.

GOLDBERG, D.E. **Genetic Algorithms in Search, Optimization and Machine Learning**. Menlo Park, Addison-Wesley, CA, 1986.

GOMES, L.C.T.; VON ZUBEN, F.J., **Uma abordagem baseada em mapas Auto-Organizáveis de Kohonen aplicada ao problema de Roteamento de Veículos**. In: Simpósio Brasileiro de Automação Inteligente, 5., 2001, Canela: Anais do V Simpósio Brasileiro de Automação Inteligente.

HAKEN, H.; SCHANZ, M.; STARKE, J., **Treatment of combinatorial optimization problems using selection equation with cost terms. Part I. Two dimensional assignment problems.** Physica D, v. 134, n. 2, p. 227-241, out. 1999.

HARPER, P.R.; SENNA, V.; VIEIRA, I.T.; SHAHANI, A.K., **A genetic algorithm for the project assignment problem.** Computers & Operations Research, v. 32, n. 5, p. 1255-1265, mar. 2005.

HAYKIN, S., **Neural Networks: A comprehensive Foundation.** New Jersey: Prentice Hall, 1999.

HOPFIELD, J.J.; TANK, D.W., **Neural Computation of decisions in optimization problems.** Biological Cybernetics, Heidelberg, v. 52, n. 3, p. 141-152, 1985.

HSIEH, K.H.; TIEN, F.C., **Self-organizing feature maps for solving location-allocation problems with rectilinear distances.** Computers & Operational Research, v. 31, n. 7, p. 1017-1031, jun. 2004.

HU, Y.C.; CHEN, R.S.; HSU, Y.T.; TZENG, G.H., **Grey self-organizing feature maps.** Neurocomputing, v. 48, n. 3, p.863-877, out. 2002.

HUNG, D.L.; WANG, J., **Digital Hardware realization of a Recurrent Neural Network for solving the Assignment Problem.** Neurocomputing, v. 51, p. 447-461, abr. 2003.

JAIN, A.K.; MURTY, M.N.; FLYNN, P.J., **Data Clustering: A Review.** ACM Computing Surveys, New York, v. 31, n. 3, p. 264-323, set. 1999.

JI, P.; LEE, W.B.; LI, H., **A new algorithm for the assignment problem: an alternative for the Hungarian method.** Computers Operational Research, London, v. 24, n. 11, p. 1017-1023, nov. 1997.

JIN, H.D.; LEUNG, K.S.; WONG, M.L.; XU, Z.B., **An Efficient Self-Organizing Map Designed by Genetic Algorithms for the Traveling Salesman Problem.** IEEE Transactions On Systems, Man, And Cybernetics - Part B: Cybernetics, v. 33, n. 6, dez. 2003.

KANGAS, J.; KOHONEN, T., **Developments and applications of the self-organizing maps and related algorithms.** Mathematics and Computers in Simulation, Amsterdam, v. 41, n. 1-2, p. 3-12, jun. 1996.

KIANG, M.Y., **Extending the Kohonen's self-organizing map networks for clustering analysis.** Computational Statistics & Data Analysis, v. 38, n. 2, p. 161-180, dez. 2001.

KIVILUOTO, K., **Topology Preservation in Self-Organizing Maps**. In: International Conference on Neural Networks, 1996, New Jersey: IEEE Neural Networks Council, p. 294-299.

KOBAYASHI, K., **Introducing a Clustering Technique into Recurrent Neural Networks for Solving Large-scale Traveling Salesman Problems**. In: International Conference on Artificial Neural Networks, 8., 1998, Skovde: Proceedings of the 8th International Conference on Artificial Neural Networks – ICANN'98, Springer Verlag, v. 2, pp. 935 -940.

KOHONEN, T., **The self-organizing map**. Neurocomputing, v. 21, n. 1-3, p. 1-6, nov. 1998.

_____ **Self-organizing maps**. Berlin: Springer Verlag, 2001.

KOWALSKI, C.T.; KOWALSKA, T.O., **Neural networks application for induction motor faults diagnosis**. Mathematics and computing in Simulation, v. 63, n. 2-3, p. 435-448, nov. 2003.

LAPORTE, G., **The vehicle routing problem: An overview of exact and approximate algorithms**. European Journal of Operational Research, v. 59, n. 3, p. 345-358, jun.1992.

LEE, J.A.; VERLEYSEN, M., **Self-organizing maps with recursive neighborhood adaptation**. Neural Networks, v. 15, n. 8-9, p. 993-1003, out. 2002.

LEUNG, K.S.; JIN, H.D.; XU, Z.B., **An expanding self-organizing neural network for the traveling salesman problem**. Neurocomputing, v. 62, p. 267-292, dez. 2004.

LIN, S., KERNIGHAN, B.W., **An effective heuristic algorithm for the traveling salesman problem**. Operational Research, v. 21, p. 498-516, 1973.

LINGO, **Manual do LINGO**. Lindo Systems, inc. N. Dayton, Chicago, Illinois, 1999.

LIU, D.; HU, S.; WANG, J., **Global Output Convergence of a Class of Continuous-Time Recurrent Neural Networks with Time-Varying Thresholds**. IEEE Transactions on Circuits and Systems, v. 51, n. 4, p. 161-167, abr. 2004.

LIU, G.; HE, Y.; FANG, Y.; QIU, Y., **A novel adaptive search strategy of intensification and diversification in tabu search**. In: Neural Networks and Signal Processing, 2003, Nanjing, China: Proceedings of Neural Networks and Signal Processing, IEEE press, v. 1, p. 428-431.

LOZANO, S.; GUERRERO, F.; ONIEVA, L.; LARRAÑETA, J., **Kohonen maps for solving a class of location-allocation problems**. European Journal of Operational Research, v. 108, n. 1, p. 106-117, jul. 1998.

MARTINS, M.P.; GUIMARÃES, L.N.F.; FONSECA, L.M.G., **Classificador de texturas por Redes Neurais**. In: Congresso Brasileiro de Computação, 2002, Itajaí: Anais do Congresso Brasileiro de Computação.

MARTÍN-VALDIVIA, M.; RUIZ-SEPÚLVEDA, A.; TRIGUERO-RUIZ, F., **Improving local minima of Hopfield networks with augmented Lagrange multipliers for large scale TSPs**. Neural Networks, v.13, n. 3, p. 283-285, abr. 2000.

MATSUDA, S., **“Optimal” Hopfield network for combinatorial optimization with linear cost function**. IEEE Transactions on Neural Networks, v. 9, n. 6, p. 1319-1330, nov. 1998.

MERZ, P.; FREISLEBEN, B., **Genetic Local Search for the TSP: New results**. In: IEEE International Conference on Evolutionary Computation, 1997, Indianapolis: Proceedings of IEEE International Conference on Evolutionary Computation, IEEE press, p. 159-164.

MINIEKA, E., **Optimization algorithms for networks and graphs**. Marcel Dekker: New York, 1978.

MULDER, S.A.; WUNSCH II, D.C., **Million city traveling salesman problem solution by divide and conquer clustering with adaptive resonance neural networks**. Neural Networks, v. 16, n. 5-6, p. 827-832, jun./jul. 2003.

MURTY, K.G., **Linear and combinatorial programming**. Robert E. Krieger: Florida, 1985.

NAGASHIMA, T.; KOBAYASHI, K.; OBAYASHI, M., **An Effective Solution to Large-scale Traveling Salesman Problems Using Chaotic Neural Networks**. In: International Conference on Neural Information Processing, 7., 2000, Taejon: Proceedings of 7th International Conference on Neural Information Processing – ICONIP’2000, pp.459-464.

OBACH, M.; WAGNER, R.; WERNER, H.; SCHMIDT, H.H., **Modelling population dynamics of aquatic insects with artificial neural networks**. Ecological Modelling, London, v. 146, n. 1-3, p. 207-217, dez. 2001.

ONG, J.; ABIDI, S.S.R., **Data Mining using Self-Organizing Kohonen maps: A technique for effective data clustering & visualization**. International Conference on Artificial Intelligence, v. 1, 1999, Las Vegas: Proceedings of International Conference on Artificial Intelligence.

ONWUBOLU, G.C.; CLERC, M., **Optimal path for automated drilling operations by a new heuristic approach using particle swarm optimization**. International Journal of Production Research v. 42, n. 3, p. 473-491, 2004.

PILAT, M.L.; WHITE, T., **Using Genetic Algorithms to Optimize ACS-TSP**. In: International Workshop on Ant Algorithms, 3., 2002, Brussels: Proceedings of the Third International Workshop on Ant Algorithms, Springer Verlag: Lecture Notes In Computer Science, v. 2463, p. 282-287.

QU, L.; SUN, R., **A synergetic approach to genetic algorithms for solving traveling salesman problem**. Information Sciences, v. 117, n. 3-4, p. 267-283, ago. 1999.

RADEVA, S.; RADEV, D., **Kochonen Neural Model for Destructive Seismic Waves**. In: Conference on Information Technology in Construction, 22., 2005, Dresden: Proceedings of the 22nd Conference on Information Technology in Construction.

REINELT, G., **TSPLIB – A traveling salesman problem library**. ORSA Journal on Computing, v. 3, n. 4, p. 376-384, 1991.

SARI, S.C.; SHERALI, H.D.; BHOOTRA, A., **New tighter polynomial length formulations for the asymmetric traveling salesman problem with and without precedence constraints**. Operations Research Letters, v. 33, n. 1, p. 62-70, jan. 2005.

SERPEN, G.; LIVINGSTON, D.L., **Determination of weights for relaxation Recurrent Neural Networks**. Neurocomputing, v. 34, n. 1-4, p. 145-168, set. 2000.

SERRANO-CINCA, C., **Self organizing neural networks for financial diagnosis**. Decision Support Systems, v. 17, n. 3, p. 227-238, jul. 1996.

SILVA, M.A.S., **Mapas Auto-Organizáveis na Análise Exploratória de dados Geoespaciais Multivariados**. São José dos Campos, 2004. 120 f. Dissertação (Mestrado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais: INPE.

SIQUEIRA, P.H., **Aplicação do Algoritmo do Matching no Problema de Construção de Escalas de Motoristas e Cobradores de Ônibus**. Curitiba, 1999, 77 f. Dissertação (Mestrado em Métodos Numéricos) – Universidade Federal do Paraná: UFPR.

SIQUEIRA, P.H.; CARNIERI, C.; STEINER, M.T.A.; BARBOZA, A.O., **Uma proposta de solução para o problema da construção de escalas de motoristas e cobradores de ônibus por meio do algoritmo do matching de peso máximo**. Gestão & Produção, São Carlos, v. 11, n. 2, p. 187-196, mai./ago. 2004.

SIQUEIRA, P.H.; SCHEER, S.; STEINER, M.T.A., **Application of the "Winner Takes All" Principle in Wang' s Recurrent Neural Network for the Assignment Problem** Advances in Neural Networks - ISNN 2005, Proceedings of Second International Symposium on Neural Networks, Chongqing, China, May 30 - June 1, 2005, Lecture Notes in Computer Science v. 3496, n. 1, p. 731-738, Springer Verlag, 2005.

STEINER, M.T.A.; SIQUEIRA, P.H.; SCHEER, S., **Solution of the assignment problem using Wang' s recurrent neural network with the principle "Winner Takes All"** In: Triennial Conference of the International Federation of Operational Research Societies - IFORS 2005, 17., Proceedings of IFORS 2005, July 11 – 15, Honolulu, Hawaii, 2005.

STUTZLE, T.; GRUN, A.; LINKE, S.; RUTTGER, M., **A comparison of nature inspired heuristics on the traveling salesman problem.** In: International Conference on Parallel Problem Solving from Nature, 6., 2000, Paris, France: Proceedings of International Conference on Parallel Problem Solving from Nature - PPSN-VI, Springer Verlag: Lecture Notes in Computer Science, v. 1917, n. 1, p. 661–670.

TAFNER, M.A.; XEREZ, M.; RODRÍGUEZ FILHO, I.W., **Redes Neurais Artificiais: introdução e princípios da neurocomputação.** Blumenau: Eko e FURB, 1996.

TALAVÁN, P.M.; YÁÑEZ, J., **Parameter setting of the Hopfield network applied to TSP.** Neural Networks, v.15, n. 3, p. 363-373, abr. 2002.

TSPLIB95. Disponível em: <<http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95>> Acesso em: maio de 2005.

TSUBAKITANI, S.; EVANS, J.R., **Optimizing tabu list size for the traveling salesman problem.** Computers & Operations Research, v. 25, n. 2, p. 91-97, fev. 1998.

ULTSCH, A., **Maps for the visualization of high-dimensional data Spaces.** In: Workshop on Self-Organizing Maps, 2003a, Kyushu: Proceedings of Workshop on Self-Organizing Maps - WSOM, p. 225-230.

_____ **U*-Matrix: a Tool to visualize clusters in high dimensional data.** Dept. of Mathematics and Computer Science, v. 36, University of Marburg, 2003b.

ULTSCH, A.; RÖSKE, F., **Self-organizing feature maps predicting sea levels.** Information Sciences, London, v. 144, n. 1-4, p. 91-125, jul. 2002.

VESANTO, J.; ALHONIEMI, E., **Clustering of the Self-Organizing Map.** IEEE Transactions on Neural Networks, v. 11, n. 3., p. 586-601, mai. 2000.

VIEIRA, F.C.; DORIA NETO, A.D.; COSTA, J.A., **An efficient approach of the SOM algorithm to the Traveling Salesman problem.** In: Brazilian Symposium on Neural Networks, 7., 2002a, Recife: Proceedings of VII Brazilian Symposium on Neural Networks - SBRN'02, IEEE press, p. 152.

VIEIRA, F.C.; DORIA NETO, A.D.; COSTA, J.A., **Uma abordagem eficiente do Algoritmo SOM ao problema do Caixeiro Viajante.** In: Brazilian Symposium on Neural Networks, 7., 2002b, Recife: Anais do VII Brazilian Symposium on Neural Networks – SBRN'02.

VIEIRA, F.H.T.; LEMOS, R.P.; LEE, L.L., **Alocação dinâmica de taxa de transmissão em Redes de Pacote utilizando Redes Neurais Recorrentes treinadas com algoritmos em tempo real.** Revista IEEE América Latina, v. 1, n. 1, out. 2003.

WANG, J., **Analog Neural Network for Solving the Assignment Problem.** Electronic Letters, v. 28, n. 11, p. 1047-1050, mai. 1992.

_____ **Analysis and Design of a Recurrent Neural Network for Linear Programming.** IEEE Transactions on Circuits and Systems, v. 40, n. 9, p. 613-618, set. 1993a.

_____ **Analysis and Design of a analog Sorting Network.** IEEE Transactions on Neural Networks, v. 6, n. 4, p. 962-971, 1993b.

_____ **A deterministic annealing neural network for convex programming.** Neural Networks, v. 7, n. 4, p. 629-641, 1994.

_____ **A Recurrent Neural Network for solving The Shortest Path problem.** IEEE Transactions on Circuits and Systems, v. 43, n. 6, p. 482-486, jun. 1996.

_____ **Primal and Dual Assignment Networks.** IEEE Transactions on Neural Networks, v. 8, n. 3, p. 784-790, mai. 1997.

WANG, R.L.; TANG, Z.; CAO, Q.P., **A learning method in Hopfield neural network for combinatorial optimization problem.** Neurocomputing, v. 48, n. 4, p. 1021-1024, out. 2002.

WILSON, J.M., **A genetic algorithm for the generalized assignment problem.** Journal of the Operational Research Society, v. 48, n. 1, p. 804-809, 1997

WITKOWSKI, U.; RÜPING, S.; RÜCKERT, U.; SCHÜTTE, F.; BEINEKE, S.; GROSTOLLEN, H. **System Identification Using Self-organizing Feature Maps.** In: International Conference on Artificial Neural Networks, 5., 1997, California: IEEE CS Press, Proceedings of International Conference on Artificial Neural Networks - ANN, p. 100-105.

WITTEN, I.H.; FRANK, E., **Data Mining: practical machine learning and techniques with Java implementations**. San Francisco: Morgan Kauffmann publishers, 2000.

WOOD, I.; DOWNS, T., **Demon algorithms and their Application to Optimization Problems**, In: International Joint Conference on Neural Networks, 1998, Anchorage, Alaska: Proceedings of the International Joint Conference on Neural Networks – IJCNN'98, IEEE press, v. 2, p. 1661-1666.

WU, S., CHOW, T.W.S., **Clustering of self-organizing map using a clustering validity index based on inter-cluster and intra-cluster density**. Pattern Recognition, v. 37, n. 2, p. 175-188, fev. 2004.

WYNS, B.; SETTE, S.; BOULLART, L.; BAETEN, D.; HOFFMAN, I.E.A.; KEYSER, F., **Prediction of diagnosis in patients with early arthritis using a combined Kohonen mapping and instance-based evaluation criterion**. Artificial Intelligence in Medicine, v. 31, n. 1, p. 45-55, mai. 2004.

XIA, Y.; WANG, J.; HUNG, D.L. **Recurrent Neural Networks for solving Linear Inequalities and Equations**. IEEE Transactions on Circuits and Systems, v. 46, n. 4, p. 452-463, abr. 1999.

XU, B.; LI, S., **Automatic Color Identification in Printed Fabric Images by a Fuzzy Neural Network**. AATCC Review, v. 2, n. 9, p. 42-45, 2002.

ZAIT, M.; MESSATFA, H., **A comparative study of clustering methods**. Future Generation Computer Systems, v. 13, n. 2-3, p. 149-159, nov. 1997.

ZAMPIGUI, L.M.; KAVANAU, C.L.; ZAMPIGUI, G.A., **The Kohonen self-organizing map: a tool for clustering and alignment of single particles imaged using random conical tilt**. Journal of Structural Biology, v. 146, n. 3, p. 368-380, jun. 2004.

ZHANG, X.; LI, Y., **Self-Organizing Map as a new method for clustering and data analysis**. In: International Joint Conference on Neural Networks, 1993, Nagoya: Proceedings of International Joint Conference on Neural Networks – IJCNN'93, p. 2448-2451.

ZHU, A.; YANG, S.X., **An Improved Self-Organizing Map Approach to Traveling Salesman Problem**. In: IEEE International Conference on Robotics, Intelligent Systems and Signal Processing, 2003, Changsha, China: Proceedings of the IEEE International Conference on Robotics, Intelligent Systems and Signal Processing, IEEE press, v. 1, p. 674-679.