

UNIVERSIDADE DE SÃO PAULO
FACULDADE DE ZOOTECNIA E ENGENHARIA DE ALIMENTOS

DIEGO SANTIAGO DOS SANTOS

**Utilização da Tecnologia *Bluetooth* associada a Redes Neurais
Artificiais (PMC) para monitoramento e rastreamento de suínos**

Pirassununga/SP - Brasil

2014

DIEGO SANTIAGO DOS SANTOS

Utilização da Tecnologia *Bluetooth* associada a Redes Neurais Artificiais (PMC) para monitoramento e rastreamento de suínos

VERSÃO CORRIGIDA

Dissertação apresentada à Faculdade de Zootecnia e Engenharia de Alimentos da Universidade de São Paulo, como parte dos requisitos para a obtenção do Título de Mestre em Zootecnia.

Área de concentração: Qualidade e Produtividade Animal

Orientador: Prof. Dr. Adriano Rogério Bruno Tech.

Pirassununga/SP – Brasil

2014

Dados Internacionais de Catalogação na Publicação

Serviço de Biblioteca e Informação da Faculdade de Zootecnia e Engenharia de Alimentos
da Universidade de São Paulo

S237u	<p>Santos, Diego Santiago dos Utilização da tecnologia bluetooth associada a redes neurais e artificiais (PMC) para monitoramento e rastreamento de suínos / Diego Santiago dos Santos. -- Pirassununga, 2014. 98f. Dissertação (Mestrado) -- Faculdade de Zootecnia e Engenharia de Alimentos - Universidade de São Paulo. Departamento de Ciências Básicas. Área de Concentração: Qualidade e Produtividade Animal. Orientador: Prof. Dr. Adriano Rogério Bruno Tech.</p> <p>1. Multilayer Perceptron 2. Sistema de Comunicação sem fio 3. Monitoramento de animais. I. Título.</p>
-------	---

AGRADECIMENTOS

Agradeço primeiramente a Deus, pelo dom da vida e por ter proporcionado a oportunidade de avançar nesta área do conhecimento guiando meus caminhos em todos os momentos e em todas as situações.

Agradeço ao meu pai Júlio, minha mãe Sueli, minha irmã Juliana, meu cunhado Eduardo e minha namorada Josiane pelo apoio incondicional. Em todos os momentos, sejam eles bons ou ruins, vocês estiveram ao meu lado. Que Deus retribua todo esse amor comigo e que a alegria proporcionada pela realização desse trabalho invada seus corações. Deus lhes pague.

Agradeço ao Prof. Dr. Adriano Rogério Bruno Tech pela paciência, pela sua imensa disponibilidade, dedicação e sábia orientação. Agradeço pela sua confiança nos resultados deste trabalho, e em seu nome agradeço a todos os Professores, Funcionários e Alunos do Laboratório de Física Aplicada e Computacional. Que Deus plante em nossos corações o desejo ainda maior à busca do conhecimento e da sabedoria.

Agradeço ao Prof. Dr. Marcelo Machado de Luca de Oliveira Ribeiro, Prefeito de Campus USP de Pirassununga e à Marta Raquel Bertasi Küll Guiguer, chefe da Divisão de Manutenção e Operação da Prefeitura do Campus USP de Pirassununga por acreditarem na proposta deste trabalho e permitirem que eu assistisse às aulas das disciplinas necessárias, em nome dos quais estendo meus agradecimentos a todos os funcionários do Campus USP de Pirassununga que de alguma forma colaboraram para a realização

deste trabalho, em especial aos companheiros da DVMANOPER Henrique, Fernando, Edilson e Jorge.

Agradeço à disponibilidade da ajuda na coleta dos dados da minha irmã Juliana, meu primo Marcel Braghin, do meu amigo Tiago Santos da Cruz. A atenção de vocês contribuiu sobremaneira para que os resultados desejados fossem alcançados.

Não poderia faltar o agradecimento à Roberta, Maíra, Lara, Luiza Suellen e Ana Paula. Também estou torcendo pelo sucesso de vocês em seus trabalhos.

Aos meus grandes amigos que, embora não estivessem diretamente envolvidos nas atividades deste projeto, sempre torceram pelo êxito dos trabalhos. A todos vocês meu muito obrigado.

E um agradecimento especial à Josiane Gonçalves Borges, que esteve ao meu lado em todos os momentos, seja me ajudando até altas horas na coleta de dados, seja no processamento dos mesmos, seja durante a confecção desta dissertação. A você, meu amor, muito obrigado por estar sempre comigo. Você foi fundamental para o êxito deste trabalho.

“O temor ao Senhor é o princípio da sabedoria. Os insensatos desprezam a sabedoria e a instrução”

(Provérbios 1, 7)

RESUMO

SANTOS, D. S. d. **Utilização da Tecnologia *Bluetooth* associada a Redes Neurais Artificiais (PMC) para monitoramento e rastreamento de suínos.** 2014. 98 f. Dissertação (Mestrado) – Faculdade de Zootecnia e Engenharia de Alimentos, Universidade de São Paulo, Pirassununga, 2014.

O presente trabalho teve como objetivo apresentar uma metodologia que permita encontrar o posicionamento e rastrear as diferentes localizações de um suíno em uma baia, utilizando o valor do *Receiver Signal Strength Indicator* (RSSI), entre o dispositivo móvel (suíno) e três dispositivos fixos, e uma Rede Neural Artificial do tipo *Perceptron* Multicamadas (PMC), responsável por interpretar os sinais RSSI e transformá-los em valores conhecidos, como em um plano cartesiano, com coordenadas no eixo X e eixo Y. A região de teste foi dividida em 289 pontos, sendo 286 utilizados para coleta de dados e para o treinamento da rede PMC. Para cada ponto, foram armazenados a sua posição dentro da baia e o valor RSSI entre o dispositivo móvel e os três dispositivos fixos. O processo foi repetido para 8 pontos escolhidos aleatoriamente dentro do espaço de teste e inseridos como entradas na rede PMC. Após treinamentos e operações realizadas com diversas arquiteturas foi possível concluir que àquela dotada de 10 neurônios na camada intermediária consistiu na melhor alternativa, cujos resultados de monitoramento e rastreamento das posições do dispositivo móvel foram encontradas com valores aceitáveis de exatidão.

Palavras-chave: MultiLayer Perceptron, Sistema de Comunicação sem fio, RSSI, Monitoramento de animais.

ABSTRACT

SANTOS, D. S. d. **Using Bluetooth technology associated with Artificial Neural Networks (MLP) for monitoring and tracking pigs**. 2014. 98 f. M. Sc. Dissertation – Faculdade de Zootecnia e Engenharia de Alimentos, Universidade de São Paulo, Pirassununga, 2014.

This paper aims to present a methodology to find the positioning and tracking of the different locations of a pig in a stall, using the value of the Receiver Signal Strength Indicator (RSSI), between the mobile device (pig) and three devices fixed, and an Artificial Neural Network Multilayer Perceptron (MLP), responsible for interpreting the RSSI signals and turning them into known values, such as on a Cartesian plane, with coordinates on X axis and Y axis. The test region was divided into 289 points, with 286 points used for data collection and training of PMC network, and for each point, it was stored its position inside the stall and its RSSI value between the mobile devices and the three fixed. The process was repeated for 8 points chosen randomly within the space of test and entered as inputs into the PMC network. After training and operations with various architectures it was concluded that the architecture with 10 neurons in the hidden layer was the best alternative, whose the results of monitoring and tracking the positions of mobile device were found with acceptable accuracy.

Keywords: MultiLayer Perceptron, Wireless Communication Systems, RSSI, Animal tracking

LISTA DE ILUSTRAÇÕES

Figura 1 - Diagrama esquemático de uma piconet e scatternet (modificada). Fonte: Billo (2003).	22
Figura 2 - Pilha de Protocolos Bluetooth (adaptado). Fonte: BILLO (2003).	23
Figura 3 - Neurônio Real (modificada). Fonte: Kovács (2006).	28
Figura 4 - Modelo de Neurônio artificial de McCulloth e Pitts em 1943. Fonte: Silva; Spatti e Flauzino (2010).	29
Figura 5 - Função Logística (modificada). Fonte: Braga; Carvalho e Ludermin (2000).	31
Figura 6 - Arquitetura feedforward de camada única (modificada). Fonte: Silva; Spatti e Flauzino (2010).	32
Figura 7 - Rede feedforward de múltiplas camadas (modificada). Fonte: Silva; Spatti e Flauzino (2010).	33
Figura 8 - Rede recorrente (modificada). Fonte: Silva; Spatti e Flauzino (2010).	34
Figura 9 - Rede PMC com uma camada intermediária.	36
Figura 10 - Área para obtenção dos dados de treinamento	49
Figura 11 - Área utilizada para a realização do experimento e coleta de dados, com cotas em metros.	50
Figura 12 - Screenshot da tela do sistema coletor de dados	51
Figura 13 - Localização pelo método da triangulação	52
Figura 14 - Descrição e preparo da área para o desenvolvimento do experimento.	56
Figura 15 - Evolução do Erro Quadrático Médio por épocas de treinamento para PMC com 6 neurônios na camada intermediária.	59
Figura 16 - Evolução do Erro Quadrático Médio por épocas de treinamento para PMC com 9 neurônios na camada intermediária.	61

Figura 17 - Evolução do Erro Quadrático Médio por épocas de treinamento para PMC com 10 neurônios na camada intermediária. _____	65
Figura 18 - Evolução do Erro Quadrático Médio por épocas de treinamento para PMC com 15 neurônios na camada intermediária. _____	69
Figura 19 - Evolução do Erro Quadrático Médio por épocas de treinamento para PMC com 20 neurônios na camada intermediária. _____	73
Figura 20 - Topologia de Rede com 10 neurônios na camada intermediária.	77

LISTA DE TABELAS

Tabela 1 - Comparação do posicionamento real com posicionamento experimental em centímetros – PMC com 6 neurônios na camada intermediária. _____	59
Tabela 2 - Comparação do posicionamento real com posicionamento experimental em centímetros – PMC com 9 neurônios na camada intermediária. _____	62
Tabela 3 - Comparação do posicionamento real com posicionamento experimental em centímetros – PMC com 10 neurônios na camada intermediária. _____	65
Tabela 4 - Comparação do posicionamento real com posicionamento experimental em centímetros – PMC com 15 neurônios na camada intermediária. _____	69
Tabela 5 - Comparação do posicionamento real com posicionamento experimental em centímetros – PMC com 20 neurônios na camada intermediária. _____	73
Tabela 6 - Comparação de desempenho de treinamentos e operação para as 5 arquiteturas de rede PMC utilizadas. _____	74
Tabela 7 - Comparação da distância real e distâncias estimadas em centímetros para cada arquitetura de rede treinada. _____	75
Tabela 8 - Comparação de desempenho de treinamento e operação para as 5 taxas de aprendizado utilizadas na rede PMC com 10 neurônios intermediários. _____	77
Tabela 9 - Comparação de desempenho de treinamento e operação para as 5 precisões de aprendizado utilizadas na rede PMC com 10 neurônios intermediários. _____	78
Tabela 10 - Comparação da distância real e distâncias estimadas em centímetros para diferentes taxas de aprendizado e precisões de rede PMC com 10 neurônios intermediários. _____	79

Tabela 11 - Dados para treinamento da rede PMC _____ 91

Tabela 12 - Dados para teste e operação da rede PMC _____ 98

LISTA DE ABREVIATURAS E SIGLAS

ACL – Asynchronous Connection Less

FHSS – Frequency Hopping Spread Spectrum

FSK – Frequency Shifting Keying

GFSK – Gaussian Frequency Shifting Keying

ISM – Industrial Scientific Medicine

L2CAP – Logical Link Control and Adaptation Protocol

MLP – Multilayer Perceptron

ONU – Organização das Nações Unidas

PDA – Personal Digital Assistant

PMC – Perceptron Multicamadas

RBF – Radial Basis Function

RNA – Rede Neural Artificial

RSSI – Receiver Signal Strenght Indicator

SCO – Synchronous Connection Oriented

SDP – Service Discovery Protocol

SIG – Special Interest Group

TDM – Time Division Multiplexing

LISTA DE SÍMBOLOS

cm - Centímetros

GHz - Gigahertz

MHz - Megahertz

m - Metros

m² – Metro cuadrado

SUMÁRIO

1. INTRODUÇÃO	15
2. OBJETIVOS	18
3. REVISÃO DE LITERATURA	19
3.1. Instalações típicas em suinocultura	19
3.2. Rede <i>Bluetooth</i>	20
3.2.1. Histórico	21
3.2.2. Topologias	21
3.2.3. Pilha de Protocolos	22
3.2.4. Estabelecimento da Comunicação	25
3.3. Redes Neurais Artificiais (RNA)	26
3.3.1. O neurônio biológico	27
3.3.2. O neurônio artificial	28
3.3.3. Função de ativação	30
3.3.4. Arquiteturas, topologias, tipos e processos de treinamento	31
3.3.5. A Rede Perceptron Multicamadas	34
4. MATERIAL E MÉTODOS	48
4.1. Tomada de dados	48
4.2. Processamento de dados	53
4.2.1. Obtenção da rede Neural	53
4.2.2. Criação da base de dados	55
5. RESULTADOS E DISCUSSÃO	57
5.1. Topologia com 6 neurônios na camada intermediária	57
5.2. Topologia com 9 neurônios na camada intermediária	60
5.3. Topologia com 10 neurônios na camada intermediária	63
5.4. Topologia com 15 neurônios na camada intermediária	66
5.5. Topologia com 20 neurônios na camada intermediária	70

5.6. Fechamento dos resultados apresentados _____	74
5.7. Otimização dos parâmetros de treinamento _____	77
6. CONCLUSÃO _____	81
7. Sugestões para futuros trabalhos _____	82
REFERÊNCIAS BIBLIOGRÁFICAS _____	84
APÊNDICE A — Algoritmo de Treinamento da Rede Perceptron Multicamadas _____	87
APÊNDICE B — Algoritmo de Operação da Rede Perceptron Multicamadas	89
APÊNDICE C — Tabela de dados normalizados para Treinamento da Rede Perceptron Multicamadas _____	91
APÊNDICE D — Tabela de dados normalizados para Operação da Rede Perceptron Multicamadas _____	98

1. INTRODUÇÃO

O crescimento da população humana e a capacidade de produção de alimentos são dois assuntos que tem despertado estudos há séculos. É de amplo conhecimento a Teoria de Malthus, de 1798, que chamava atenção sobre o grande incremento populacional ocorrido nos séculos anteriores, o que poderia levar a grandes problemas de alimentação, fato que não se concretizou graças ao avanço das técnicas agrícolas e ao desenvolvimento de técnicas mais eficientes de produção de alimentos.

Para produzir alimentos para uma população que superou 7 bilhões de indivíduos (ONU, 2012), é necessário o desenvolvimento de tecnologias que promovam o incremento da produção de gêneros alimentícios.

No entanto, o controle e monitoramento da produção animal não possui o mesmo desenvolvimento, quando comparado com a indústria, embora já existam comercialmente sistemas de monitoramento integrado para suínos, aves e gado leiteiro (SEVEGNANI et al., 2005).

Nesse contexto surgem os conceitos de monitoramento e rastreamento que podem ser inseridos como alternativas para o estudo de comportamento de um rebanho suíno, podendo levar a importantes conclusões a respeito do impacto comportamental sobre a qualidade e produtividade animal. Associado a esse conceito é necessário o uso de sistemas tecnológicos que permitam conhecer a localização e deslocamento de um suíno em sua baia ou em uma área determinada.

Dentro desse panorama são necessários sistemas de comunicação com ou sem fio, que possam permitir a identificação e a presença do animal, bem como um sistema que interprete a presença de modo a indicar a sua localização e o seu comportamento, dentro do espaço monitorado.

Com isso, o uso da Tecnologia *Bluetooth* consiste numa alternativa interessante, pelo seu baixo custo e pelo alcance reduzido (BILLO, 2003), permitindo fácil adaptação ao ambiente de criação do rebanho suíno. Além disso, a implementação dessa tecnologia possibilita que sejam adquiridos dados zootécnicos, provenientes de sensores distribuídos no ambiente monitorado, de forma a proporcionar o melhor conforto e bem estar possível ao animal, como temperatura, umidade, ventilação, entre outros, que constituirão em um amplo espectro de possibilidades de pesquisas.

Por se tratar de um sistema de comunicação afetado pela distância, tem-se a configuração de uma função matemática, cuja interpretação pode ser feita através da utilização das Redes Neurais Artificiais conhecidas como Redes *Perceptron* Multicamadas, com o algoritmo de correção de erro baseado no modelo matemático chamado de retropropagação do erro, ou *Backpropagation*. Uma vez que os algoritmos de treinamento dessa tecnologia são capazes de extraírem características para aproximar qualquer função contínua (BRAGA; CARVALHO; LUDERMIN, 2000), a utilização das RNA consiste numa poderosa ferramenta para a interpretação e análise da distância (deslocamento do suíno) que está ocorrendo a comunicação de sinal através da tecnologia *Bluetooth*. Por meio da triangulação entre os nós distribuídos no ambiente (fixos) e os nós móveis, acoplados aos suínos, permitirão determinar a posição do suíno no espaço ou eixo cartesiano (X, Y). O armazenamento desses das diferentes

posições encontradas permitirá a construção de uma base de dados com a série histórica das diferentes posições, permitindo que se realize o rastreamento das diferentes posições ocupadas no interior da baía.

Desta forma, o trabalho está dividido em 5 (cinco) Capítulos, os quais propiciarão ao leitor uma melhor compreensão em relação aos objetivos propostos e sua aplicação na Zootecnia de Precisão, através da utilização de Tecnologias e Sistemas de Gestão, que permitirão integrar conhecimento e tomadas de decisões que levarão à otimização da produtividade, por meio da análise de deslocamento e do comportamento dos animais monitorados, dentro de seu ambiente de criação.

2. OBJETIVOS

O objetivo do presente trabalho consiste em propor um sistema de monitoramento e rastreamento de posição de suínos, em um espaço de confinamento, por meio da medição da potência do sinal da comunicação *Bluetooth* entre os nós fixos e os nós móveis (animais monitorados), sendo estes sinais incorporados à rede como entrada, e cujas saídas forneceram as coordenadas X e Y de localização do animal, permitindo o monitoramento, rastreamento das diferentes posições e consequente análise de deslocamentos dos mesmos no tempo e no espaço.

3. REVISÃO DE LITERATURA

Os conceitos que permeiam este trabalho estão divididos em três grandes áreas: a rede sem fio *Bluetooth*, cuja potência de sinal consiste na base para a caracterização dos pontos que serão rastreados; a rede neural artificial, baseada no modelo *Perceptron* Multicamadas (PMC), com retropropagação do erro (*Backpropagation*), que foi responsável por interpretar os valores de potência dos sinais adquiridos via tecnologia *Bluetooth* e transformá-los em posições dentro do espaço rastreado e; as instalações zootécnicas aplicadas a suínos, que devido às suas dimensões fazem com que a suinocultura seja o tipo de criação mais apropriada para a implementação deste sistema, devido ao espaço utilizado pelos animais monitorados.

3.1. Instalações típicas em suinocultura

Segundo SIG (2001) as aplicações *Bluetooth* atingem distâncias próximas a 10 metros, podendo se estender até 15 metros. Nesse cenário, as instalações zootécnicas para suínos são perfeitamente compatíveis com a proposta do presente trabalho.

Segundo Sartor; Souza e Tinôco (2006), a área por baia para cada porca deve ser de $2,5\text{m}^2$ e sendo recomendado de 4 a 6 fêmeas por baia, com comprimento e largura próximos de 4m, totalizando uma área de 16m^2 .

Dando embasamento ao descrito pelo autor anterior, em União Européia (2009), as porcas e as marrãs (denominação dada às porcas adultas, antes da primeira parição) devem ser instaladas em espaços de $1,64\text{m}^2$ e $2,25\text{m}^2$, respectivamente.

Além disso, em Amaral et al. (2006) relata-se que a granja de suíno, de forma semelhante aos outros trabalhos citados, deve propiciar baias com uma área de 2m^2 para leitoas, sendo recomendado a acomodação de 6 a 10 leitoas por baia. Para porcas é recomendada uma área de 3m^2 , por animal, sendo acomodados de 5 a 10 animais por área.

Dessa forma, considerando a distância ativa de uma rede *Bluetooth*, os suínos constituem o tipo de criação mais interessante, por serem minimizadas as possibilidades de não funcionamento ou falhas de recepção de sinal dentro do espaço monitorado, o que causaria um comportamento indesejado para sua implementação.

3.2. Rede *Bluetooth*

A rede *Bluetooth* é uma tecnologia de comunicação que forma um link de rádio de curto alcance, cujo objetivo principal de criação consistiu em substituir os cabos para a troca de dados, entre dispositivos eletrônicos (KANSAL, 2002). Tem como características principais o baixo consumo de potência, a baixa taxa de alcance e de transmissão (BILLO, 2003).

Embora inicialmente possa parecer estranha a utilização de uma rede com tantas limitações, é importante ressaltar que nem todas as aplicações necessitam de altas taxas de velocidade ou grande alcance como, por exemplo, a sincronização de uma agenda de contatos entre um telefone celular e um computador ou o uso de um fone de ouvido sem fio.

Ao encontro dessas utilizações vem o baixo consumo de potência, característico desse tipo de rede, o que torna a tecnologia *Bluetooth* uma alternativa interessante para diversas aplicações.

3.2.1. Histórico

De acordo com Tanenbaum (2002), a rede *Bluetooth* foi desenvolvida em 1994, pela empresa L. M. Ericsson, com o intuito de conectar sua linha de telefones a outros dispositivos, como os PDAs.

O nome do projeto foi dado em homenagem ao Rei Harold Blaatand “*Bluetooth*” II, que reinou no período de 940-981 d.C. e, unificou a Dinamarca e a Noruega (KANSAL, 2002).

Em fevereiro de 1998, a Ericsson juntamente com a IBM, Intel, Nokia e Toshiba formaram o SIG (*Special Interest Group*), um consórcio com a intenção de desenvolver um padrão de rede sem fio aberto (TANEMBAUM, 2002). Segundo o mesmo autor, em julho 1999 o SIG publicou as especificações do protocolo *Bluetooth* 1.0.

Atualmente, o SIG é composto por outras empresas como Motorola, Apple, LG e a rede *Bluetooth* encontra-se na versão 4.1, cujas especificações foram lançadas em dezembro de 2013 (www.bluetooth.com).

3.2.2. Topologias

De acordo com Billo (2003), é possível estabelecer conexões *Bluetooth* do tipo ponto-a-ponto (*ad-hoc*), como do tipo ponto-multiponto. Esta conexão ponto-multiponto é conhecida como *piconet* e é caracterizada pela conexão entre um dispositivo mestre com até outros 7 dispositivos escravos, dentro da distância ativa da conexão (TANEMBAUM, 2002).

Segundo Billo (2003), o dispositivo mestre é o responsável por controlar o acesso dos dispositivos escravos, além de determinar as características das transmissões.

Quando um dispositivo presente em uma *piconet* pertence à outra *piconet*, tem-se a formação de uma *scatternet*. Tais topologias podem ser vistas na Figura 1.

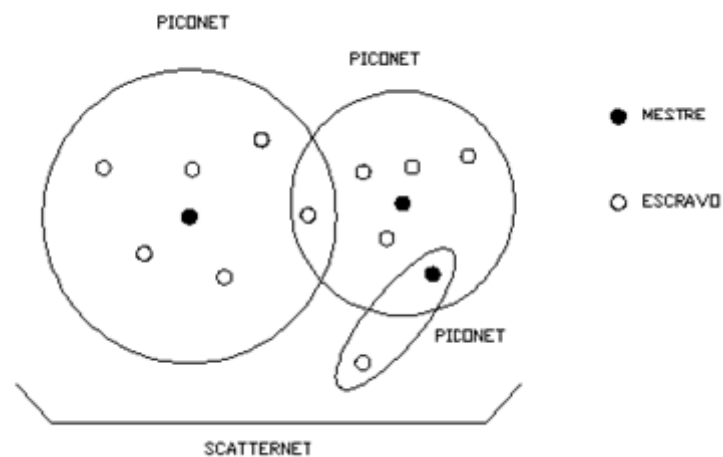


Figura 1 - Diagrama esquemático de uma piconet e scatternet (modificada). Fonte: Billo (2003).

Conforme a figura, é possível verificar que um dispositivo escravo pode pertencer a duas *piconets* ao mesmo tempo. Além disso, existe a possibilidade de um dispositivo ser escravo em uma *piconet* e mestre em outra. Numa *piconet* nunca haverá dois mestres, e não é possível a comunicação entre dois nós (pontos) escravos.

3.2.3. Pilha de Protocolos

Com o intuito de facilitar o entendimento dos protocolos com os quais a rede *Bluetooth* funciona, estes são dispostos na forma de camadas. Na literatura podem ser encontradas diversas versões esquemáticas da pilha de protocolos *Bluetooth*.

Essas versões não são distintas entre si de modo global. Enquanto Tanenbaum (2002) apresenta uma ilustração complexa, Billo (2003) e Kansal (2002) apresentam versões sintéticas, que são mais interessantes para o escopo desse trabalho. Na Figura 2, adaptada de Billo (2003), tem-se a representação esquemática da pilha de protocolos *Bluetooth*.

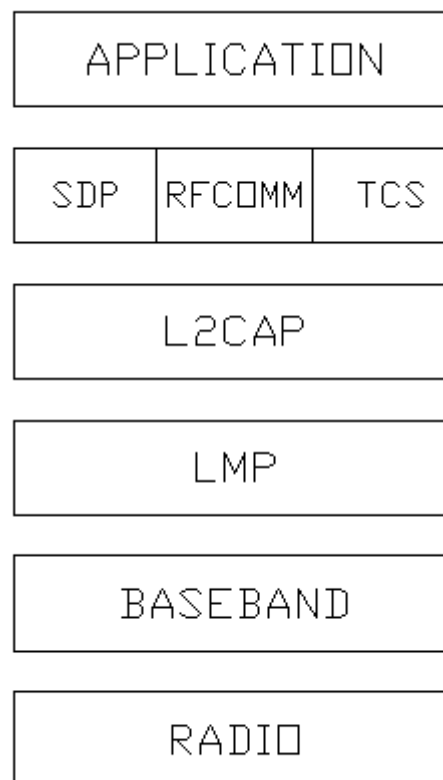


Figura 2 - Pilha de Protocolos *Bluetooth* (adaptado). Fonte: BILLO (2003).

A camada de Rádio é a camada responsável pela movimentação dos bits entre os pontos mestre e escravo (TANENBAUM, 2002). Segundo SIG (2001), a rede *Bluetooth* opera na frequência de 2,4000GHz, conhecida como banda ISM (*Industrial Scientific Medicine*). A faixa completa de frequência é de 2,4000GHz a 2,4835 GHz. Essa banda de frequência está dividida em 79

canais nomeados de 0 a 78, com largura de banda de 1,0MHz, por canal. Além disso, existe uma banda guarda inferior de 2,0MHz e uma banda guarda superior de 3,5MHz.

Durante a transmissão, o dispositivo mestre alterna continuamente o canal utilizado. Tal técnica é conhecida como *Frequency Hopping Spread Spectrum* (FHSS), que conforme é visto em Torres (2001), trata-se da alternância de tempos em tempos do canal utilizado. Com isso, só pode ter acesso à informação os dispositivos que souberem qual é a sequência de saltos de canais de frequência. Dessa forma, é diminuída a interceptação dos dados e a interferência entre duas transmissões que ocorram simultaneamente (TORRES, 2001).

Em SIG (2001) observa-se que o sinal *Bluetooth* é modulado pelo método *Gaussian Frequency Shift Keying* (GFSK). Em Forouzan (2007), a modulação *Frequency Shift Keying* (FSK) é apresentada como um sistema de modulação em que a frequência da portadora varia para representar um dado. Em outras palavras, a frequência do sinal modulado é a mesma para uma informação 0 e muda de valor, caso o a informação seja 1.

A Camada *Baseband* é responsável por controlar o rádio (KANSAL, 2002). Essa camada é responsável por providenciar a sequência de saltos da técnica FHSS, além de transformar o fluxo de bits em quadros, que serão transmitidos em multiplexação por divisão de tempo ou TDM (*Time Division Multiplexing*). É o tipo de multiplexação em que vários canais são transmitidos pelo mesmo meio a cada tempo (STALLINGS, 1997).

Segundo Kansal (2002) é possível estabelecer dois tipos de *links*: *Synchronous Connection Oriented* (SCO), que é uma transmissão síncrona

usada para dados de voz e; *Asynchronous Connection Less (ACL)*, usada para transmissão de dados.

O *Link Manager Protocol*, conforme Kansal (2002) é o protocolo responsável por três funções básicas: auxiliar o dispositivo mestre de uma *piconet* a gerenciar o compartilhamento de canal; configuração do *link*, permitindo a inclusão ou exclusão de dispositivos escravos em um *piconet* e; função de segurança, através da autenticação dos dispositivos a serem ligados.

O *Logical Link Control and Adaptation Protocol (L2CAP)* é o protocolo responsável por organizar os pacotes de dados recebidos das camadas superiores em quadros, para a transmissão (TANEMBAUM, 2002).

Segundo o mesmo autor, este protocolo é quem determina a qual protocolo superior um pacote deve ser entregue, como, por exemplo, RFCOMM. Em terceiro lugar é o protocolo responsável por gerenciar a qualidade de serviço. Se o L2CAP está presente no módulo *Bluetooth*, então a camada *Host Controller* é desnecessária.

A Camada de Aplicação consiste nas próprias aplicações que utilizam o link *Bluetooth* (KANSAL, 2002), que para operarem podem necessitar de outros protocolos, como o RFCOMM (que emula uma comunicação serial) ou o *Service Discovery Protocol (SDP)*.

3.2.4. Estabelecimento da Comunicação

Segundo Billo (2003) e Kansal (2002), um dispositivo *Bluetooth* permanece por padrão em **Standby**, um modo de consumo de potência extremamente baixo.

Quando colocado no estado de ***Inquiry***, o dispositivo envia um pacote denominado *inquiry packet*, com o intuito de encontrar outros dispositivos na sua abrangência. Quando um dispositivo permite ser localizado por outros dispositivos, configuração chamada de *inquiry enable*, o mesmo coloca-se esporadicamente no estado de ***Inquiry Scan***.

Nesse estado, se o dispositivo receber um *inquiry packet* ao ser procurado, tal dispositivo enviará o sinal *inquiry response*, fornecendo seu endereço físico e permitindo que o dispositivo que realiza a busca tome conhecimento da sua presença.

3.3. Redes Neurais Artificiais (RNA)

Silva; Spatti e Flauzino (2010) definem redes neurais artificiais como sendo modelos computacionais que buscam operar de modo análogo ao sistema nervoso dos seres vivos. Trata-se de unidades de processamento, também denominadas neurônios artificiais, que possuem a faculdade de adquirir e manter um determinado conhecimento, sobre um determinado problema ou evento.

Haykin (2001) define uma rede neural artificial como um processador constituído de unidades de processamento simples, que se assemelham ao cérebro humano, por adquirir conhecimento através de um processo de aprendizagem (treinamento) e por armazenar as informações entre suas conexões, através dos pesos sinápticos entre as conexões neuronais artificiais.

Um grande benefício das redes neurais artificiais é a sua capacidade de generalização, ou seja, uma RNA é capaz de fornecer saídas coerentes, para

valores de entrada que não constituíam a experiência adquirida pela rede ao ser treinada (HAYKIN, 2001).

As principais características das redes neurais artificiais apontadas por Silva; Spatti e Flauzino (2010) e Haykin (2001) são:

- Necessidade de pouco conhecimento estatístico: uma rede neural é capaz de extrair as relações existentes entre entradas e saídas. Muitas vezes seu desempenho é superior a métodos estatísticos;

- Adaptação por experiência e capacidade de aprendizado: a aquisição do conhecimento se dá a partir da apresentação de diversos exemplos, denominados amostras. Com o uso de um método de treinamento, uma RNA pode extrair qualquer relação existente entre as variáveis da aplicação;

- Generalização: uma vez treinada, uma rede neural artificial pode encontrar soluções para situações desconhecidas;

- Tolerância a falhas: como há um grande número de interconexões, uma RNA é capaz de operar mesmo quando existem falhas em algum de seus neurônios.

3.3.1. O neurônio biológico

O neurônio é definido em Silva; Spatti e Flauzino (2010) como célula elementar do sistema nervoso. Basicamente, possui várias entradas (dendritos) e uma única saída (axônio). Os elementos constituintes dos neurônios estão ilustrados na Figura 3.

As entradas de um neurônio estão conectadas quimicamente aos axônios de outras células nervosas, graças às conexões sinápticas (KOVÁČZ, 2006), ou seja, a função do neurônio consiste em conduzir impulsos elétricos provenientes de reações físico-químicas.

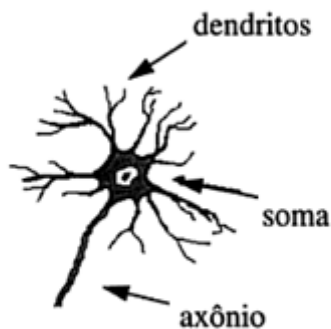


Figura 3 - Neurônio Real (modificada). Fonte: Kovács (2006).

A região das sinapses contém neurotransmissores, substâncias responsáveis por transferir o estímulo recebido ao dendrito do neurônio, levando a uma alteração de potencial elétrico (conhecido por limiar de ativação), além de serem os responsáveis pela ponderação do sinal advindo do axônio.

Segundo Crossman (2011), os neurotransmissores são armazenados em vesículas da terminação pré-sináptica. Assim sendo, dependendo do neurotransmissor liberado na sinapse, esta poderá ser excitatória ou inibitória. Caso ocorra a primeira, será gerado um impulso enviado ao axônio; caso contrário, o impulso será contido (KOVÁ CZ, 2006). Os neurotransmissores mais importantes no Sistema Nervoso Central são o ácido glutâmico e o ácido gama-aminobutírico, que atuam como transmissores excitatórios e inibitórios, respectivamente (CROSSMAN, 2011).

3.3.2. O neurônio artificial

O neurônio artificial pode ser definido como a unidade de processamento fundamental de uma RNA (HAYKIN, 2001). O modelo mais

simples foi concebido e inspirado nas características de uma rede neural biológica, por McCulloch e Pitts, em 1943 (SILVA; SPATTI; FLAUZINO, 2010).

Esse modelo é ilustrado na Figura 4.

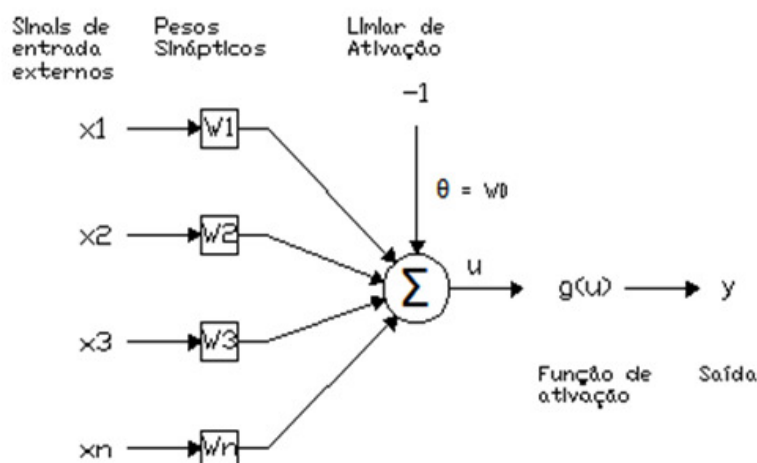


Figura 4 - Modelo de Neurônio artificial de McCulloch e Pitts em 1943. Fonte: Silva; Spatti e Flauzino (2010).

Comparado a um neurônio biológico, os dendritos estão à esquerda. Nessa região, observa-se a chegada de impulsos advindos externamente, denominado camada de entrada, representados pelo vetor $[X_1, X_2, X_3, \dots, X_n]^T$. A função dos neurotransmissores é exercida pela matriz de pesos sinápticos, representada pelo vetor $[W_1, W_2, W_3, \dots, W_n]^T$.

O corpo celular do neurônio biológico é representado pelo combinador linear. Nele ocorre a união de todos os sinais externos, após a ponderação por seus respectivos pesos sinápticos e do limiar de ativação. A subtração entre o valor obtido pelo combinador linear e o limiar de ativação gera o potencial de ativação, representado pela letra "u" na Figura 4.

O potencial de ativação é em seguida modificado pela função de ativação (g), cuja função é de limitar o valor do sinal dentro de um intervalo

aceitável. Devido à sua característica, essa função é denominada em Haykin (2001) também como função restritiva. Após a aplicação da função de ativação é gerada o sinal de saída (y). O comportamento de um neurônio artificial pode ser representado pelas Eq. (1) e (2):

$$u = \sum_{i=1}^n W_i \cdot X_i - \theta \quad (1)$$

$$y = g(u) \quad (2)$$

em que u é o Potencial de Ativação, W_i o peso sináptico referente à i -ésima entrada, X_i é o valor da i -ésima entrada, θ é o limiar de ativação, y é a saída do neurônio e $g(u)$ a função de ativação aplicada ao potencial de ativação gerado.

Para fins de simplificação algorítmica, o limiar de ativação pode ser incluído à matriz de pesos sinápticos e, dessa forma, pode-se reescrever a Eq. (1) anterior, de acordo com a Eq. (3):

$$u = \sum_{i=0}^n W_i \cdot X_i \quad (3)$$

em que u é o Potencial de Ativação, W_i é o peso sináptico referente à i -ésima entrada, X_i é o valor da i -ésima entrada, sendo que $i=0$ refere-se ao limiar de ativação do neurônio.

3.3.3. Função de ativação

Como visto anteriormente, a função de ativação tem a função de limitar a saída do neurônio artificial dentro de um intervalo aceitável. Para redes

neurais artificiais são utilizadas funções parcialmente diferenciáveis, como por exemplo, a função degrau e função bipolar e funções totalmente diferenciáveis, como a função logística e a função tangente hiperbólica.

Em virtude do tipo de rede necessária, considerou-se mais adequada a função de ativação logística. Essa função é dada pela relação Eq. (4)

$$g(u) = \frac{1}{1 + e^{-u}} \quad (4)$$

onde u é o potencial de ativação. A função está ilustrada na Figura 5

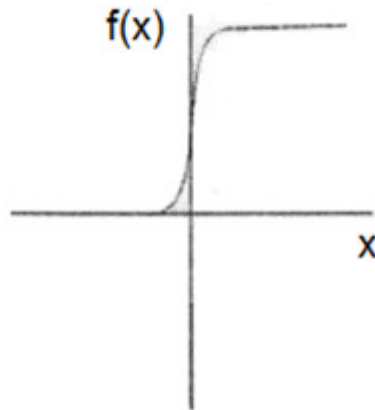


Figura 5 - Função Logística (modificada). Fonte: Braga; Carvalho e Ludermin (2000).

3.3.4. Arquiteturas, topologias, tipos e processos de treinamento

Entende-se por arquitetura de rede a maneira como os neurônios estão arranjados (SILVA; SPATTI; FLAUZINO 2010; HAYKIN 2001). Segundo os mesmos autores, as principais arquiteturas de redes neurais artificiais são:

- **Feedforward (redes adiante) de camada simples:** caracterizadas pelo fluxo de informações seguir em uma única direção. Os tipos de redes neurais mais conhecidos implementados nessa arquitetura são: o *Perceptron* e

o *Adaline*, cujos processos de treinamento são respectivamente, a Regra de Hebb e Regra Delta.

Dentro de uma arquitetura podem ser obtidas diferentes topologias. Na Figura 6 é ilustrada uma rede *feedforward*, de camada única, cuja topologia é caracterizada por uma camada de entrada com n entradas (sinapses) e camada neural de saída com n_1 neurônios.

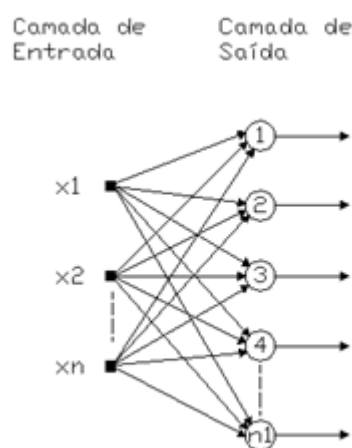


Figura 6 - Arquitetura *feedforward* de camada única (modificada). Fonte: Silva; Spatti e Flauzino (2010).

- ***Feedforward* (redes adiante) de camadas múltiplas:** de grande interesse para este trabalho, essa arquitetura de redes neurais caracteriza-se assim, como a arquitetura anterior, pelo fluxo de informações ser unidirecional. No entanto, esta arquitetura de rede caracteriza-se por possuir uma ou mais camadas intermediárias, também denominadas camadas escondidas. As arquiteturas mais conhecidas são a *Perceptron* Multicamadas (*Perceptron Multilayer*) e as redes de base radial (*radial basis function* – RBF), cujos respectivos treinamentos são dados pelas Regras Delta Generalizada e Delta Competitiva. Entre as diversas topologias que podem ser construídas para essa arquitetura, a Figura 7 representa uma rede com camada de entrada com

n sinais, 2 camadas intermediárias, sendo a primeira com n_1 neurônios e a segunda com 2 neurônios e camada de saída, com 1 neurônio.

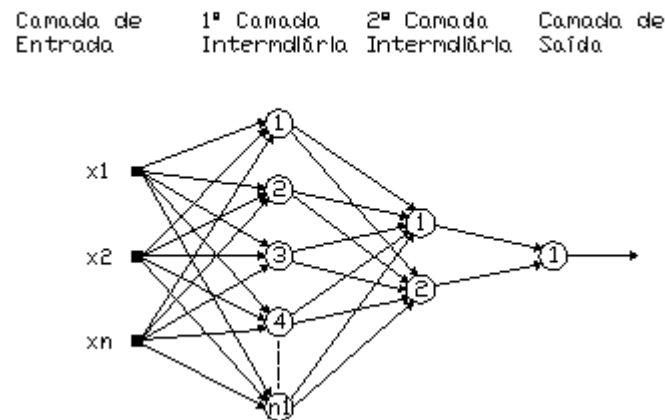


Figura 7 - Rede *feedforward* de múltiplas camadas (modificada). Fonte: Silva; Spatti e Flauzino (2010).

- **Redes recorrentes:** são caracterizadas pela presença de realimentação, ou seja, as saídas dos neurônios são utilizadas como entradas para outros neurônios das camadas anteriores. Os exemplos mais conhecidos de redes recorrentes são: a Rede de Hopfield e a rede *Perceptron* multicamadas com realimentação. Seus processos de treinamento são respectivamente, o método da minimização de energia e a Regra Delta generalizada. A Figura 8 ilustra uma rede com realimentação, numa topologia formada por 4 neurônios.

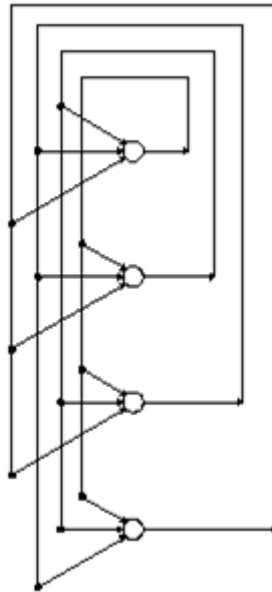


Figura 8 - Rede recorrente (modificada). Fonte: Silva; Spatti e Flauzino (2010).

3.3.5. A Rede Perceptron Multicamadas

As redes do tipo Perceptron Multicamadas (PMC) são caracterizadas por possuírem ao menos uma camada intermediária de neurônios (SILVA; SPATTI; FLAUZINO, 2010). Esse tipo de rede possui um poder computacional muito maior do que as redes de uma única camada, pois são capazes de tratar os dados que não são linearmente separáveis.

Devido à inexistência de conhecimento sobre algoritmos de treinamentos para redes com camadas intermediárias, as pesquisas sobre RNA ficaram estacionadas na década de 70 (BRAGA; CARVALHO; LUDERMIN, 2000).

Com o advento das redes PMC foram encontradas novas aplicações para redes neurais, o que despertou novamente o interesse em pesquisa e desenvolvimento nessa área. No caso da PMC, entre as principais aplicações, enumeradas por Silva; Spatti e Flauzino (2010) estão: a aproximação universal

de funções, reconhecimento de padrões, identificação e controle de processos, previsão de séries temporais e otimização de sistemas.

A rede PMC é classificada como arquitetura *feedforward*, ou seja, em seu processamento as informações seguem um único sentido. Seu processo de treinamento, conhecido como algoritmo de retropropagação de erro (*Backpropagation*), publicado em 1986 por Rumelhart (SILVA; SPATTI; FLAUZINO, 2010), cuja essência de implementação é o gradiente descendente do erro. Devido a isso, sua função de ativação obrigatoriamente deve ser contínua e diferenciável (BRAGA; CARVALHO; LUDERMIN, 2000).

A topologia que melhor se ajusta ao problema em geral é definida empiricamente (BRAGA; CARVALHO; LUDERMIN, 2000), uma vez que, segundo o mesmo autor, o número adequado de neurônios, depende do número de amostras de treinamento, quantidade de ruído nas amostras, complexidade da função a ser aprendida e distribuição estatística dos dados.

A Figura 9 ilustra uma Rede PMC, que possui camada de entrada com n sinais, camada intermediária com n_1 neurônios e camada de saída com n_2 neurônios, muito semelhante à implementação usada nesse trabalho.

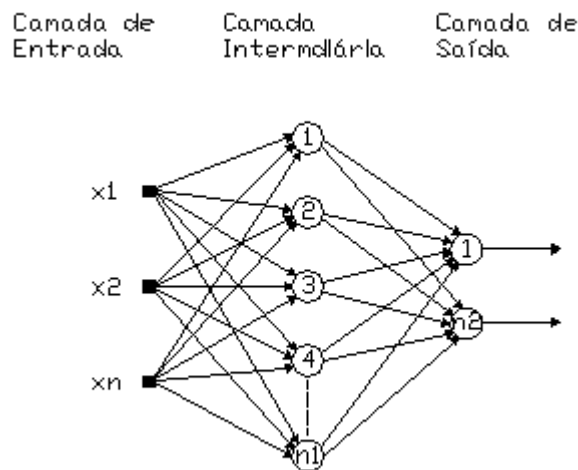


Figura 9 - Rede PMC com uma camada intermediária.

Os n dados de entrada são recebidos pelos neurônios da camada intermediária. Em cada neurônio ocorrerão os processos que matematicamente foram descritos pela Eq. (1) e (2).

No caso da rede PMC, como existem vários neurônios, torna-se mais interessante agrupar os valores dos sinais presentes na rede em forma de vetores, em que cada linha representa um neurônio de determinada camada ou um sinal de entrada (para o caso da camada de entrada).

É importante salientar que ao número de entradas de cada camada deve ser acrescido o sinal -1 , que multiplica o limiar de ativação e lhe confere a característica de subtração, como visto na Eq. (1). Por isso, os vetores de sinais sempre terão o número -1 na primeira linha, que será denominada nos algoritmos como linha "0".

No caso dos pesos sinápticos e limiar de ativação, como existem vários pesos por neurônio (um por sinal de entrada) e vários neurônios por camada, seus valores são apresentados na forma de matrizes, em que cada linha representa um neurônio e cada coluna expressa o multiplicador: a primeira

coluna contém o limiar de ativação (chamada de coluna “0”) e as demais contêm os pesos sinápticos de cada entrada.

Através da Figura 9 observa-se que ocorrerão as sinapses dos sinais da camada de entrada com a primeira camada escondida. Aplicando as Equações (3) e (2) em cada neurônio j de cada camada, obtém-se a formação dos vetores de potenciais de ativação, conforme as Equações (5) e (6):

$$I_j^{(1)} = \sum_{i=0}^n W_{j,i}^{(1)} \cdot X_i^{(1)} = W_{j,0}^{(1)} \cdot X_0 + \dots + W_{j,n}^{(1)} \cdot X_n \quad (5)$$

$$I_j^{(2)} = \sum_{i=0}^{n1} W_{j,i}^{(2)} \cdot Y_i^{(1)} = W_{j,0}^{(2)} \cdot Y_0^{(1)} + \dots + W_{j,n1}^{(2)} \cdot Y_{n1}^{(1)} \quad (6)$$

em que $I_j^{(1)}$ é o potencial de ativação do j -ésimo neurônio da primeira camada escondida, $W_{j,i}^{(1)}$ é peso sináptico da i -ésima entrada aplicada pelo j -ésimo neurônio da primeira camada escondida, X_i é o sinal da i -ésima entrada, $I_j^{(2)}$ é o potencial de ativação do j -ésimo neurônio da camada de saída, $W_{j,i}^{(2)}$ é peso sináptico da i -ésima entrada aplicada pelo j -ésimo neurônio da camada de saída e Y_i é o sinal de saída dos neurônio j da camada anterior, acrescido do sinal -1 que irá multiplicar o limiar de ativação.

Como descrito anteriormente, em cada neurônio é aplicada a função de ativação no potencial de ativação, gerado após as sinapses. Dessa forma, a saída de cada neurônio j , em cada camada neural é dada pelas Equações (7) e (8):

$$Y_j^{(1)} = g(I_j^{(1)}) \quad (7)$$

$$Y_j^{(2)} = g(I_j^{(2)}) \quad (8)$$

em que $Y_j^{(1)}$ é a saída do neurônio j da camada intermediária, $I_j^{(1)}$ é o potencial de ativação do mesmo neurônio, $Y_j^{(2)}$ é a saída do neurônio j da camada de saída, $I_j^{(2)}$ é o potencial de ativação do mesmo neurônio.

Os $Y_j^{(1)}$ gerados, que constituem a saída dos neurônios da primeira camada intermediária, são os sinais de entrada da próxima camada neural e, para isso deverá ser acrescentado o valor -1, que será multiplicado pelo limiar de ativação dos neurônios da camada seguinte. Os $Y_j^{(2)}$ constituem a saída da rede neural.

O treinamento da rede PMC, segundo Silva; Spatti e Flauzino (2010) é realizado pelo algoritmo de retropropagação do erro, que consiste na minimização do erro de aproximação entre os resultados fornecidos pela rede e as saídas desejadas.

Para cada amostra de entradas fornecida à rede, saídas são geradas, e serão comparadas às saídas desejadas.

O algoritmo *Backpropagation* visa realizar a minimização do erro, fazendo a correção nos valores dos pesos sinápticos e limiar de ativação dos neurônios da camada de saída. Em seguida é realizada a atualização da matriz de pesos sinápticos e limiar de ativação dos neurônios das camadas anteriores, justificando, assim, o nome de retropropagação do erro.

Dada uma amostra z -ésima de entradas apresentada a rede, o erro para determinado neurônio é dado pela subtração do valor obtido (Y_j) ao valor desejado (d_j). O valor instantâneo da energia do erro para o referido neurônio (e_j) é dado pela Eq. (9).

$$e_j = \frac{1}{2} (d_j - Y_j)^2 \quad (9)$$

Dessa forma, o erro quadrático, que representa o desempenho local da rede para a amostra é dado pelo somatório da energia instantânea do erro em cada neurônio, dada pela Eq. (10)

$$E(z) = \frac{1}{2} \sum_{j=1}^{n2} (d_j - Y_j)^2 \quad (10)$$

sendo $E(z)$ o somatório dos diferentes e_j calculados pela equação anterior, d_j os valores desejados para a saída do neurônio j e Y_j a saída fornecida pelo mesmo neurônio.

Supondo que existam p amostras de entrada e estando aplicadas, a rede PMC fornecerá um erro quadrático médio dado pela média de todos os erros quadráticos calculados para cada amostra, como pode ser observado na Eq. (11).

$$E_{méd} = \frac{1}{p} \sum_{z=1}^p E(z) \quad (11)$$

À aplicação completa de todas as amostras de entradas para treinamento, dá-se o nome de época. O algoritmo da retropropagação de erro ajustará o valor das matrizes de pesos sinápticos e limiar de ativação de todos os neurônios, até que o erro quadrático médio, entre duas épocas sucessivas, seja menor que o valor de precisão especificado para o treinamento.

Para o ajuste das matrizes de pesos sinápticos e limiar de ativação busca-se a minimização do erro entre a saída obtida e o valor desejado. Dessa forma, aplicando o operador gradiente, tem-se a Eq. (12)

$$\nabla E^{(2)} = \frac{\partial E}{\partial W_{j,i}^{(2)}} = \frac{\partial E}{\partial Y_j^{(2)}} \frac{\partial Y_j^{(2)}}{\partial I_j^{(2)}} \frac{\partial I_j^{(2)}}{\partial W_{j,i}^{(2)}} \quad (12)$$

onde $\frac{\partial E}{\partial W_{j,i}^{(2)}}$ é a derivada parcial do erro em relação à matriz de pesos sinápticos

da camada de saída, $\frac{\partial E}{\partial Y_j^{(2)}}$ é a derivada parcial do erro em relação a saída do

neurônio j da camada de saída, $\frac{\partial Y_j^{(2)}}{\partial I_j^{(2)}}$ é a derivada parcial da saída do neurônio j

da camada de saída em relação ao potencial de ativação do mesmo neurônio e

$\frac{\partial I_j^{(2)}}{\partial W_{j,i}^{(2)}}$ é a derivada parcial do potencial de ativação do j-ésimo neurônio da

camada de saída em relação aos pesos sinápticos do mesmo neurônio.

Se forem analisadas as equações (6), (8) e (10) tem-se as Equações (13), (14) e (15):

$$\frac{\partial I_j^{(2)}}{\partial W_{j,i}^{(2)}} = Y_i^{(1)} \quad (13)$$

$$\frac{\partial Y_j^{(2)}}{\partial I_j^{(2)}} = g'(I_j^{(2)}) \quad (14)$$

$$\frac{\partial E}{\partial Y_j^{(2)}} = -(d_j - Y_j^{(2)}) \quad (15)$$

em que $\frac{\partial I_j^{(2)}}{\partial W_{j,i}^{(2)}}$ é a derivada parcial do potencial de ativação do neurônio j da camada de saída em relação aos pesos sinápticos do mesmo neurônio, cujo valor corresponde a entrada de dados da camada neural de saída, $Y_i^{(1)}$; $\frac{\partial Y_j^{(2)}}{\partial I_j^{(2)}}$ é a derivada parcial da saída dos j -ésimo neurônio da camada de saída em relação ao potencial de ativação do mesmo neurônio, cujo valor é a derivada de primeira ordem da função de ativação aplicada ao potencial de ativação do neurônio da camada de saída citado e; $\frac{\partial E}{\partial Y_j^{(2)}}$ é a derivada parcial do erro em relação a saída do j -ésimo neurônio da camada de saída, dada pela subtração entre o valor desejado para a saída do neurônio j e a saída fornecida pelo mesmo neurônio. O sinal negativo surge nessa equação em função da aplicação da regra da cadeia.

Substituindo as Eq. (13), (14) e (15) na aplicação do operador gradiente dado pela Eq. (12), tem-se a Eq. (16):

$$\frac{\partial E}{\partial W_{j,i}^{(2)}} = -(d_j - Y_j^{(2)}) \cdot g'(I_j^{(2)}) \cdot Y_i^{(1)} \quad (16)$$

O treinamento dá-se ao alterar os valores dos pesos sinápticos e limiares de ativação a cada amostra apresentada, até que se alcance o menor erro possível.

Dessa forma, aplica-se um termo denominado fator de aprendizagem, que indica o tamanho do passo de variação dos valores dos pesos sinápticos e dos limiares de ativação.

A variação dos valores das matrizes é dada, portanto, pela aplicação do fator de aprendizagem no sentido contrário ao do gradiente da função erro, onde obtém-se a Eq. (17).

$$\Delta W_{j,i}^{(2)} = -\eta \cdot \frac{\partial E}{\partial W_{j,i}^{(2)}} = \eta \cdot \delta_j^{(2)} \cdot Y_i^{(1)} \quad (17)$$

onde o termo $\Delta W_{j,i}^{(2)}$ representa a alteração do valor do i-ésimo peso sináptico do j-ésimo neurônio da camada neural de saída, η é a taxa de aprendizagem do treinamento e $\delta_j^{(2)}$ é o gradiente local do j-ésimo neurônio da camada de saída, dado pela Eq. (18).

$$\delta_j^{(2)} = (d_j - Y_j^{(2)}) \cdot g'(I_j^{(2)}) \quad (18)$$

Em relação à Eq. (17), pode-se notar que o sinal negativo da orientação contrária ao gradiente anulou-se com o sinal negativo proveniente do cálculo do gradiente do erro.

Sabendo-se calcular a variação dos valores das matrizes de pesos sinápticos e limiares de ativação, pode-se reescrever os valores das relações acima, em função do valor anterior e do valor atual das referidas matrizes e, dessa forma, tem-se a equação que realiza a atualização da matriz de pesos sinápticos e limiar de ativação dos neurônios da camada de saída, dada pela Eq. (19).

$$W_{j,i}^{(2)}(atual) = W_{j,i}^{(2)}(anterior) + \eta \cdot \delta_j^{(2)} \cdot Y_i^{(1)} \quad (19)$$

Para a camada intermediária, Haykin (2001) aponta a dificuldade do fato de não existir as respostas desejadas para os neurônios da camada intermediária. Por isso, o ajuste da matriz de pesos sinápticos e limiar de ativação dos neurônios da camada intermediária são feitos em função de estimativas dos erros produzidos pelos neurônios da camada da frente, cuja matriz de pesos sinápticos e limiar de ativação foi previamente alterada.

Dessa forma, deseja-se ajustar a matriz de pesos sinápticos e limiar de ativação dos neurônios da primeira camada intermediária, através da minimização do erro da saída da rede, frente ao erro produzido após os ajustes da camada de saída. Assim, a aplicação do operador gradiente para os neurônios da camada intermediária pode ser calculado através da Eq.(20).

$$\nabla E^{(1)} = \frac{\partial E}{\partial W_{j,i}^{(1)}} = \frac{\partial E}{\partial Y_j^{(1)}} \frac{\partial Y_j^{(1)}}{\partial I_j^{(1)}} \frac{\partial I_j^{(1)}}{\partial W_{j,i}^{(1)}} \quad (20)$$

onde $\frac{\partial E}{\partial W_{j,i}^{(1)}}$ é a derivada parcial do erro em relação à matriz de pesos sinápticos

da camada intermediária, $\frac{\partial E}{\partial Y_j^{(1)}}$ é a derivada parcial do erro em relação a saída

do neurônio j da camada intermediária, $\frac{\partial Y_j^{(1)}}{\partial I_j^{(1)}}$ é a derivada parcial da saída dos

neurônios j da camada intermediária em relação ao potencial de ativação do j-

ésimo neurônio da mesma camada e $\frac{\partial I_j^{(1)}}{\partial W_{j,i}^{(1)}}$ é a derivada parcial do potencial de

ativação do j-ésimo neurônio da camada intermediária em relação aos pesos sinápticos do neurônio da desta mesma camada.

De modo análogo ao processo de ajuste da matriz de pesos sinápticos e limiar de ativação dos neurônios da camada de saída, analisando as Eq. (5) e (7), tem-se as Equações (21), (22) e (23).

$$\frac{\partial I_j^{(1)}}{\partial W_{j,i}^{(1)}} = X_i \quad (21)$$

$$\frac{\partial Y_j^{(1)}}{\partial I_j^{(1)}} = g'(I_j^{(1)}) \quad (22)$$

$$\frac{\partial E}{\partial Y_j^{(1)}} = \sum_{z=1}^{n_2} \frac{\partial E}{\partial I_z^{(2)}} \cdot \frac{\partial I_z^{(2)}}{\partial Y_j^{(1)}} = \sum_{z=1}^{n_2} \frac{\partial E}{\partial I_z^{(2)}} \cdot \frac{\partial (\sum_{z=1}^{n_2} W_{z,j}^{(2)} \cdot Y_j^{(1)})}{\partial Y_j^{(1)}} \quad (23)$$

Na Eq. (21) observa-se que a derivada parcial do potencial de ativação do j-ésimo neurônio em relação aos pesos sinápticos da camada intermediária

$\frac{\partial I_j^{(1)}}{\partial W_{j,i}^{(1)}}$ equivale à matriz de entrada X_i ; na Eq. (22), verifica-se que a derivada

parcial da saída do neurônio j da camada intermediária em relação ao potencial

de ativação do mesmo neurônio consiste na derivada de primeira ordem da

função de ativação aplicada ao potencial de ativação do referido neurônio e; na

Eq. (23) é possível verificar que a derivada parcial do erro em relação à saída

do j-ésimo neurônio da camada intermediária é dado pelo somatório, para

todos os neurônios da camada de saída, da derivada parcial do erro em

relação ao potencial de ativação do z-ésimo neurônio, da camada de saída,

multiplicada pela derivada parcial do potencial de ativação do mesmo neurônio em relação à saída do j-ésimo neurônio, da camada intermediária. O termo $I_z^{(2)}$ pode ser substituído pela multiplicação da matriz de pesos sinápticos dos neurônios da camada de saída pela saída dos neurônios da camada intermediária, chegando assim ao último membro apresentado.

A Eq. (23) pode ser simplificada e reescrita como apresentado na Eq. (24).

$$\frac{\partial E}{\partial Y_j^{(1)}} = \sum_{z=1}^{n2} \frac{\partial E}{\partial I_z^{(2)}} \cdot W_{z,j}^{(2)} \quad (24)$$

A derivada parcial do erro em relação ao potencial de ativação do z-ésimo neurônio da camada de saída, $\partial E / \partial I_z^{(2)}$, pode ser substituído pelo resultado da multiplicação das equações (14) e (15), que resulta em na Eq. (25).

$$\frac{\partial E}{\partial Y_j^{(1)}} = - \sum_{z=1}^{n2} \delta_z^{(2)} \cdot W_{z,j}^{(2)} \quad (25)$$

onde é possível observar que a derivada parcial do erro em relação à saída do j-ésimo neurônio da camada intermediária é igual ao valor negativo do somatório, para todos os neurônios da camada de saída, da multiplicação entre o gradiente local do neurônio z da camada de saída pelos pesos sinápticos do mesmo neurônio.

Substituindo as Eq. (21), (22) e (25) na eq. (20), obtém-se a Eq. (26).

$$\frac{\partial E}{\partial W_{j,i}^{(1)}} = - \left(\sum_{z=1}^{n2} \delta_j^{(2)} \cdot W_{z,j}^{(2)} \right) \cdot g' \left(I_j^{(1)} \right) \cdot X_i \quad (26)$$

Com isso, tem-se que o ajuste na matriz de pesos sinápticos e limiar de ativação dos neurônios da camada intermediária, deve ser dado pela aplicação do fator de aprendizagem ao gradiente da função de erro, conforme as Equações (27) e (28):

$$\Delta W_{j,i}^{(1)} = -\eta \cdot \frac{\partial E}{\partial W_{j,i}^{(1)}} = -\eta \cdot \delta_j^{(1)} \cdot X_i \quad (27)$$

onde:

$$\delta_j^{(1)} = \left(\sum_{z=1}^{n2} \delta_j^{(2)} \cdot W_{z,j}^{(2)} \right) \cdot g' \left(I_j^{(1)} \right) \quad (28)$$

De forma análoga ao treinamento da camada neural de saída, alguns termos foram reunidos na variável $\delta_j^{(1)}$ representando o gradiente local do j-ésimo neurônio da camada intermediária. Em forma iterativa, tem-se, portanto, a Eq. (29)

$$W_{j,i}^{(1)}(\text{atual}) = W_{j,i}^{(1)}(\text{anterior}) + \eta \cdot \delta_j^{(1)} \cdot X_i \quad (29)$$

Dessa forma, é possível realizar o treinamento da rede neural artificial, a ponto de se encontrar valores de pesos sinápticos e limiares de ativação que sejam os mais coerentes, a fim de que dado um conjunto de valores de entrada

leve a valores de saída dentro das características extraídas do conjunto de amostras.

4. MATERIAL E MÉTODOS

Para a realização do trabalho, o mesmo foi dividido em duas etapas, sendo a primeira atividade realizada em campo e a segunda em laboratório.

A primeira etapa consistiu na tomada de dados de potência de sinal de transmissão *Bluetooth*, através da colocação dos pontos fixos na área determinada, bem como o referenciamento do mesmo, em relação ao eixo cartesiano.

A segunda etapa foi realizada para a construção e montagem da base de dados, para serem processadas pela RNA. Os pontos adquiridos pela tecnologia *Bluetooth* foram armazenadas numa base de dados.

Esses dados foram preparados e passaram a compor a base de dados para o treinamento, validação e testes da RNA PMC escolhida, com correção do erro baseado no algoritmo *Backpropagation*, onde a mesma disponibiliza como saída duas informações, ambas relacionadas às coordenadas X e Y, para localização dos suínos monitorados dentro da área do experimento por meio dos dados coletados na primeira etapa do trabalho.

4.1. Tomada de dados

Para a realização da etapa experimental de aquisição de dados foi necessária a obtenção de uma área que reproduzisse uma granja de suínos. Segundo Amaral et al. (2006), uma granja deve possuir baias que forneçam um espaço de 2m² por leitoas, em número de seis a dez por área, independente do tamanho da baia, e 3m² por porcas, em número de cinco a dez animais por área. Dessa forma, demarcou-se uma área na Faculdade de Zootecnia e Engenharia de Alimentos (FZEA-USP) em forma quadrada, de piso em

concreto com 4 metros de lado, totalizando 16m². Com isso, é possível alocar 8 leitoas ou 5 porcas.



Figura 10 - Área para obtenção dos dados de treinamento

Dentro dessa área, ilustrada na Figura 10 foram coletados dados para que a rede neural aprendesse a encontrar um animal em seu interior, ou seja, a rede recebeu os sinais coletados pela emissão de potência do dispositivo móvel (suínos) referente a três antenas fixas em uma área demarcada e referenciada, com coordenadas X e Y e o seu processo de treinamento consistiu em fazer com que a rede fornecesse as coordenadas X e Y da localização do dispositivo móvel. A região foi dividida em pontos distantes de 25 cm entre si, formando uma matriz ou malha de 17 linhas e 17 colunas, totalizando 289 pontos distintos. Essas são as posições ensinadas à rede PMC.

Posições diferentes dessas, desde que dentro da área demarcada, devem ser localizadas pela rede neural através de sua característica de generalização (SILVA; SPATTI; FLAUZINO, 2010). Por questão organizacional, foi selecionado um vértice para ser usado como referencial, como em um

sistema de eixo cartesiano, sendo denominada na base de dados de coordenadas X e Y (0,0).

Para a caracterização dos pontos, foram instalados três dispositivos detentores da tecnologia *Bluetooth*, chamados de dispositivo 1, 2 e 3 (ou estações fixas), fixados nas posições ou coordenadas (100, 100), (200,300) e (300,100) centímetros, respectivamente, conforme apresentado na Figura 11.

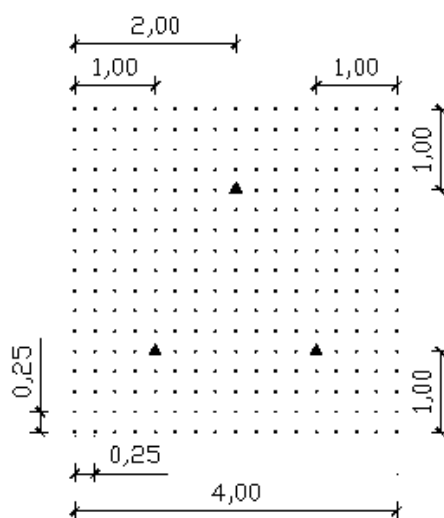


Figura 11 - Área utilizada para a realização do experimento e coleta de dados, com cotas em metros.

Os três dispositivos possuem tecnologia *Bluetooth* versão 4.0 e possuem o software *Bluetooth SPP PRO* (software livre sobre a plataforma Android, V 4.1), que é capaz de enviar um sinal de *inquiry packet* para os dispositivos *Bluetooth* à sua volta e ao receber o sinal *inquiry response* do dispositivo móvel, expressa a potência dessa comunicação.

A potência do sinal de resposta, conhecida nos meios de telecomunicações como *Received Signal strenght indication* (RSSI), é o parâmetro de referência para a caracterização de cada ponto.

Um quarto dispositivo, detentor de tecnologia *Bluetooth* versão 2.1, foi instalado em cada ponto da matriz de pontos construída. Em cada local, identificado como posição (0,0), (25,0), (50,0) centímetros e assim sucessivamente, foi medido o RSSI (sinal) entre este dispositivo volante e os três dispositivos fixos.

Em cada uma das posições em que o quarto dispositivo estava instalado foi medido por vinte vezes o RSSI entre este e os três equipamentos fixos. Na Figura 12 pode-se visualizar o software do dispositivo 1, fazendo a busca e identificação do dispositivo móvel.

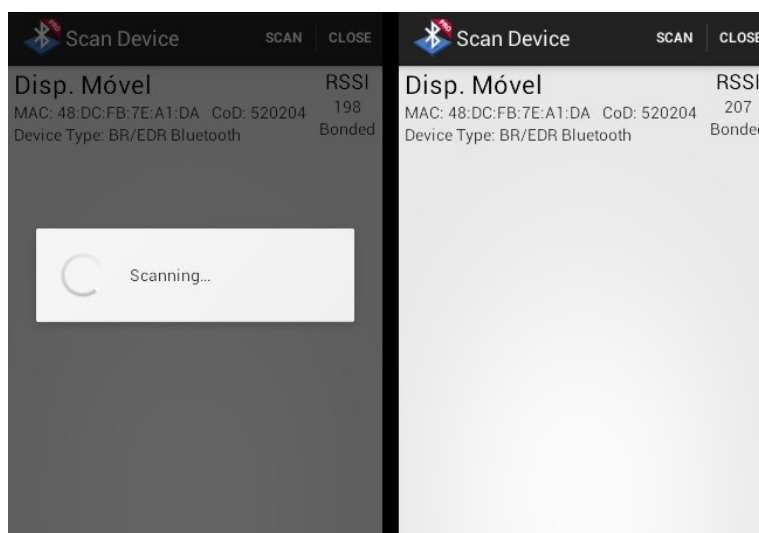


Figura 12 - Screenshot da tela do sistema coletor de dados

O *software* retorna um número adimensional que exprime a potência do sinal de *Bluetooth*. Os dados coletados foram organizados em forma de planilha onde cada ponto medido correspondia a uma linha e para cada linha existiam 3 colunas, uma para cada dispositivo *Bluetooth* fixo (estações fixas).

Cada uma dessas 3 colunas foi subdivididas em 20 subcolunas nas quais foram dispostos o valor do RSSI. No total foram coletados dados em 286 pontos, uma vez que nos pontos (100,100), (200,300) e (300,100) centímetros

o dispositivo móvel não foi instalado, visto que nestes pontos estavam os dispositivos fixos 1, 2 e 3, respectivamente.

O sistema de posicionamento é semelhante ao conhecido método da triangulação, em que dado um ponto num plano e três pontos de referência conhecidos, através da distância entre o ponto de teste e os três pontos conhecidos, é possível encontrar a posição do dispositivo móvel.

Para este trabalho, a obtenção da distância é substituída pela potência do sinal *Bluetooth* e a interpretação dessa potência em distância é realizada pela rede neural artificial. A Figura 13 ilustra a proposta implementada.

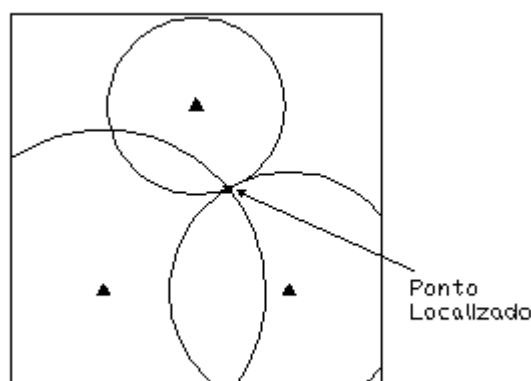


Figura 13 - Localização pelo método da triangulação

Terminadas as medições nos 286 pontos, estava composta a planilha com dados para o treinamento da rede. Para a etapa de teste e validação da rede neural artificial, foram escolhidos 8 pontos arbitrários dentro do espaço de 16m^2 e, novamente foi medido 20 vezes o valor do RSSI entre o dispositivo móvel e os três dispositivos fixos.

De modo análogo, esses dados foram dispostos em planilha em que cada linha representa um ponto de teste e para cada uma das linhas existem 3 colunas divididas em 20 subcolunas. Dessa forma, estava completa a aquisição

de dados, existindo um conjunto de dados para o treinamento e um conjunto de dados para teste da Rede Neural Artificial.

4.2. Processamento de dados

A segunda etapa consistiu na confecção de uma rede neural e de uma base de dados, para treinamento e operação da referida rede.

4.2.1. Obtenção da rede Neural

A situação pode ser entendida como a de uma aproximação funcional, uma vez que o valor do RSSI decai de modo não linear, em função do aumento da distância entre o dispositivo móvel e um dispositivo fixo, ou seja, o valor RSSI é uma função da distância entre os dispositivos.

Por isso, embora existam diferentes tipos de Redes Neurais Artificiais, o tipo que melhor se aplica a este caso é a do *Perceptron* Multicamadas, PMC, ou *Multilayer Perceptron* (MLP), pois segundo o Teorema da Aproximação funcional, um PMC treinado a partir do método de aprendizagem de retropropagação do erro (*backpropagation*) constitui uma ferramenta de grande aplicabilidade para mapeamentos de sistemas não lineares do tipo entrada e saída (HAYKIN, 2001).

Quanto à topologia da rede, ficou evidente que esta deveria possuir camada de entrada com três sinais distintos, que se referem ao valor médio de RSSI fornecido pelos dispositivos fixos 1, 2 e 3, respectivamente e camada de saída dotada de dois neurônios, sendo um responsável por fornecer a primeira coordenada da localização do dispositivo móvel (X) e o outro responsável pela segunda coordenada (Y).

Quanto ao número de camadas intermediárias, segundo Silva; Spatti e Flauzino (2010), o Teorema da Aproximação funcional estabelece que uma camada neural intermediária é suficiente para extrair as características de qualquer função. No entanto, a quantidade de neurônios na camada intermediária que melhor poderá extrair as características da função em questão é obtida de modo empírico.

Dessa forma, foram confeccionados um algoritmo de treinamento para Rede Neural Artificial do tipo *Perceptron* Multicamadas (PMC), com método de aprendizagem de Retropropagação (*Backpropagation*) e um algoritmo de Operação para a Rede PMC que estão reproduzidos nos apêndices A e B, respectivamente. Como o número de neurônios da camada intermediária não era conhecido, ambos os algoritmos foram confeccionados de forma a permitirem que se escolha o número de neurônios na referida camada, para que fosse feito o treinamento e a operação, permitindo, assim, a comparação dos seus desempenhos.

Nesse contexto, foram realizados treinamento e operação em redes com 6, 9, 10, 15 e 20 neurônios na camada intermediária (escondida), estando fixos para todas as redes uma taxa de aprendizagem de 0,01 e precisão de 1×10^{-6} . Após a operação das redes treinadas observou-se qual topologia apresentou melhores resultados. Encontrada a melhor topologia, novos treinamentos da rede PMC foram realizados, utilizando como taxas de aprendizado os valores 0,1, 0,05, 0,005 e 0,001 com precisão fixa em 1×10^{-6} , de forma a encontrar com qual taxa de aprendizagem a rede forneceu melhores respostas.

Em seguida, utilizando a taxa de aprendizagem de melhor desempenho, foram realizados novos treinamentos e operações com diferentes valores de

precisão: 1×10^{-2} , 1×10^{-4} , 1×10^{-8} e 1×10^{-10} . Dessa forma, foi possível encontrar qual conjunto de parâmetros de configuração da rede PMC seria capaz de fornecer resultados mais satisfatórios.

4.2.2. Criação da base de dados

Como visto anteriormente, para cada ponto em que o dispositivo móvel foi acionado foram coletados vinte medidas de potência (RSSI), entre este e os três dispositivos fixos.

O número de repetições justifica-se porque o valor do RSSI não é exatamente o mesmo para a mesma distância. Como a tecnologia *Bluetooth* é um sinal de rádio, é importante lembrar que ondas eletromagnéticas ao se propagarem na atmosfera sofrem absorção de energia por parte de vapor d'água e moléculas de oxigênio (MIGUENS, 2001).

Desta forma, a simples dinâmica atmosférica, como ventos ou mudança na umidade relativa do ar, são capazes de provocarem medidas de RSSI diferentes, para a mesma distância.

Por isso, o valor de RSSI que caracteriza a distância do dispositivo móvel ao dispositivo fixo foi dado pela média aritmética dos vinte valores RSSI encontrados. No entanto, nos pontos próximos aos vértices, o valor RSSI por muitas vezes não foi encontrado, por um ou mais dispositivos fixos.

Desta maneira, uma vez que a probabilidade de uma busca de RSSI não desse resultado, optou-se por ignorar a busca atribuindo valor zero ao RSSI toda vez que a média entre os dispositivos móvel (suíno) e fixo fosse inferior a 150, ou seja, admite-se que a identificação do ponto móvel (suíno) seria prejudicada, pois para que ocorra a localização do suíno ou de outro animal ou objeto são necessários que as três estações fixas possam identificar o ponto

móvel e retornar sua potência, para que através da interseção entre as médias obtidas possa-se localizar o animal procurado.

No entanto, como visto anteriormente, a função logística usada na Rede PMC é totalmente diferenciável, variando no intervalo de 0 a 1. Por isso, a fim de evitar que a função logística fique em saturação devido aos valores elevados das médias e das coordenadas X e Y, os dados foram normalizados da seguinte maneira: foi atribuído o valor 1 ao maior valor RSSI encontrado (202) e 0 aos casos em que a busca de valor RSSI não forneceu resultado. De modo análogo, para as coordenadas foi atribuído o valor 0 para as coordenadas X ou Y iguais a 0 e 1 para as coordenadas X ou Y iguais a 400 cm. Com isso, todos os valores de RSSI (que variaram de 0 até 202) e todos os valores de coordenadas X e Y (que podem valer de 0 a 400 cm) ficaram compreendidos no intervalo entre 0 e 1.

A Figura 14 sintetiza os passos realizados nesse trabalho, desde a escolha do local, até a obtenção de respostas da provável posição (X,Y) do dispositivo móvel (suíno).

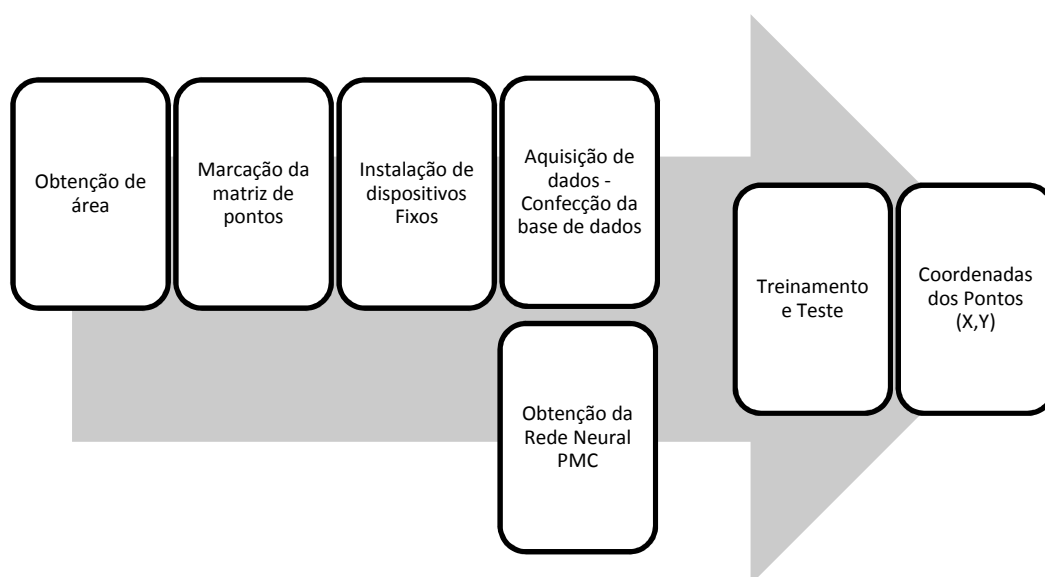


Figura 14 - Descrição e preparo da área para o desenvolvimento do experimento.

5. RESULTADOS E DISCUSSÃO

Para a realização dos testes das arquiteturas de Redes Neurais propostas foi criada uma base de dados, onde estão relacionadas às coordenadas dentro do espaço usado para teste, as médias das vinte medidas feitas entre o dispositivo móvel acoplado ao suíno e os três dispositivos fixos, pontos com tecnologia *Bluetooth*, responsáveis por identificar através de triangulação a posição do animal no espaço monitorado.

Os dados foram normalizados para evitar uma saturação na função de ativação logística da PMC. Os valores obtidos, bem como os valores normalizados utilizados pelo algoritmo de treinamento e operação da rede estão referenciados na Tabela 11 e Tabela 12, dos Anexos C e D, respectivamente.

5.1. Topologia com 6 neurônios na camada intermediária

Ao realizar o treino com 6 neurônios, foram apresentadas ao algoritmo a Matriz 1 e Matriz 2 contendo os pesos sinápticos e limiar de ativação iniciais para a primeira camada intermediária e para a camada de saída.

$$\text{MP1} = \begin{matrix} 0,9880 & 0,8828 & 0,4504 & 0,6732 \\ 0,8641 & 0,9137 & 0,2057 & 0,6643 \\ 0,3889 & 0,5583 & 0,8997 & 0,1228 \\ 0,4547 & 0,5989 & 0,7626 & 0,4073 \\ 0,2467 & 0,1489 & 0,8825 & 0,2753 \\ 0,7844 & 0,8997 & 0,2850 & 0,7167 \end{matrix}$$

Matriz 1 – Pré-treinamento - Camada intermediária (6x4).

$$\text{MP2} = \begin{matrix} 0,2834 & 0,8266 & 0,4979 & 0,8344 & 0,5747 & 0,4564 & 0,8844 \\ 0,8962 & 0,3900 & 0,6948 & 0,6096 & 0,3260 & 0,7138 & 0,7209 \end{matrix}$$

Matriz 2 – Pré-treinamento - Camada de saída (2x7).

O treinamento dessa arquitetura com a configuração de parâmetros citada anteriormente, atingiu precisão $9,9265 \times 10^{-7}$, após 561 épocas de treinamento.

Com isso, a rede atingiu um erro quadrático médio no treinamento de 0,0492, e forneceu novos resultados, ou seja, novas matrizes de pesos sinápticos e limiar de ativação para a camada neural intermediária e camada neural de saída. Os valores obtidos encontram-se na Matriz 3 e

Matriz 4. O gráfico do comportamento do erro relativo médio pelo número de épocas de treinamento pode ser visualizado na Figura 15.

MP1=	1,3991	1,4704	-0,4115	1,1622
	0,9697	1,0051	-0,1257	0,8771
	0,2897	-0,5925	1,5450	-0,7294
	0,4508	0,5802	0,7479	0,4199
	-0,2503	-1,6589	2,3829	-1,1141
	0,8512	0,8822	0,0585	0,8480

Matriz 3 – Pós-treinamento - Camada intermediária (6x4).

MP2=	0,0300	-0,7458	-0,6280	0,3766	-0,2365	1,6012	-0,0749
	0,1036	-1,8078	-1,0144	1,7126	-0,0977	3,2803	-0,8025

Matriz 4 – Pós-treinamento - Camada de saída (2x7).

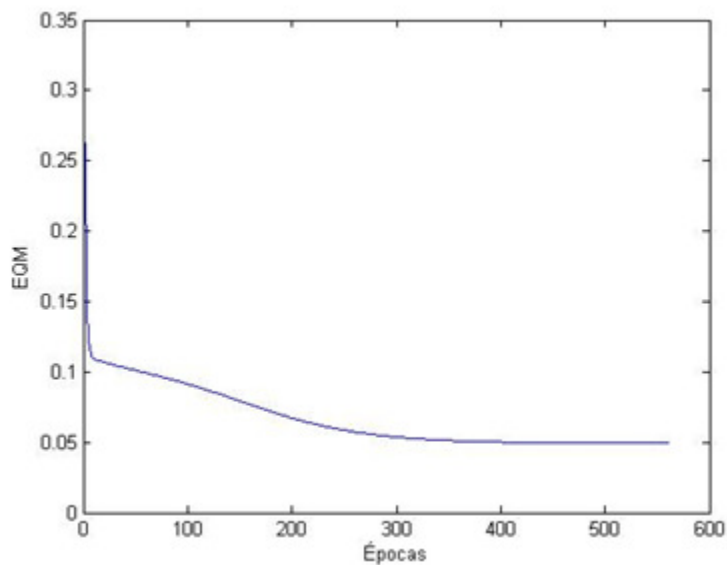


Figura 15 - Evolução do Erro Quadrático Médio por épocas de treinamento para PMC com 6 neurônios na camada intermediária.

Ao serem aplicados os valores de teste a essa rede, com as matrizes de pesos sinápticos e com os limiares de ativação obtidos após o treinamento, foram encontrados os pontos de posicionamento, dados pela Tabela 1:

Tabela 1 - Comparação do posicionamento real com posicionamento experimental em centímetros – PMC com 6 neurônios na camada intermediária.

X (real)	Y (real)	X (exp)	Y (exp)	Erro X	Erro Y
156,00	179,00	186,48	198,60	30,48	19,60
149,00	223,00	188,68	204,20	39,68	18,80
209,00	248,00	195,76	220,84	13,24	27,16
222,00	217,00	186,84	199,44	35,16	17,56
243,00	200,00	184,32	193,44	58,68	6,56
251,00	157,00	181,08	185,48	69,92	28,48
209,00	160,00	183,44	191,00	25,56	31,00
162,00	134,00	178,08	178,68	16,08	44,68
Erro Médio:				36,10	24,23
Erro Relativo Médio em %:				17,81%	14,07%

A Tabela 1 mostra que a implementação de rede com 6 neurônios na camada intermediária fornece resultados razoáveis, pois é possível verificar que em média houve um deslocamento de 36,10 cm, entre as coordenadas X

do ponto real e do ponto fornecido pela rede e de 24,23 cm entre as coordenadas Y, do ponto real e do ponto fornecido pela rede.

5.2. Topologia com 9 neurônios na camada intermediária

Para a implementação com 9 neurônios na camada escondida foram apresentadas para treinamento a Matriz 5 e a Matriz 6.

```
MP1 =  0,4981  0,626  0,0170  0,6147
       0,9009  0,6609  0,1209  0,3624
       0,5747  0,7298  0,8627  0,0495
       0,8452  0,8908  0,4843  0,4896
       0,7386  0,9823  0,8449  0,1925
       0,5860  0,7690  0,2094  0,1231
       0,2467  0,5814  0,5523  0,2055
       0,6664  0,9283  0,6299  0,1465
       0,0835  0,5801  0,0320  0,1891
```

Matriz 5 - Pré-treinamento - Camada intermediária (9x4).

```
MP2 =  0,0427  0,2819  0,6952  0,5358  0,1239  0,8530  0,2703  0,5650  0,4170  0,9479
       0,6352  0,5386  0,4991  0,4452  0,4904  0,8739  0,2085  0,6403  0,2060  0,0821
```

Matriz 6 - Pré-treinamento - Camada de saída (2x10).

É importante ressaltar que a Matriz 5 e a Matriz 6 possuem dimensões diferentes das matrizes vistas para o caso anterior, devido à diferença na quantidade de neurônios na camada intermediária.

Além disso, os valores apresentados não possuem relação alguma com os valores das matrizes anteriores, pelo fato das matrizes iniciais terem sido geradas a partir do comando `rand(linhas, colunas)`, comando aplicado no Matlab, versão R2008a.

Após a realização de 2588 épocas de treinamento, o algoritmo terminou seu processo iterativo ao alcançar a precisão de $9,99 \times 10^{-7}$. Nesse instante, o erro quadrático médio do treinamento alcançou 0,0435. Com isso, a nova

Matriz 7 e Matriz 8, contendo os pesos sinápticos e limiar de ativação para as camadas neurais foram obtidas, cujos valores são:

```

MP1 =  0,4412  0,4000  -0,6360  0,9559
        0,8490  0,5062  -0,2255  0,5405
        0,8518  0,0776  2,3935  -1,3528
        1,1051  1,0416  -1,2403  1,2808
        0,9449  -0,2480  3,5321  -2,0337
        0,9240  1,6477  -2,4131  1,4922
        0,3471  -0,9268  3,8776  -2,3397
        1,2416  1,7974  -2,0689  1,3232
        0,4167  1,1788  -2,3450  1,3765

```

Matriz 7 - Pós-treinamento - Camada intermediária (9x4).

```

Colunas 1 a 7
MP2 = -4,0643  2,4345  0,3954  -2,8811  -0,8632  -2,5519  -2,2102
        -0,1626  -1,2160  -0,6599  2,4780  -2,0476  3,8672  -3,2233

```

```

Colunas 8 a 10
        -0,2213  -3,5503  -0,0317
        4,4202  -2,9479  -2,9963

```

Matriz 8 - Pós-treinamento - Camada de saída (2x10).

Na Figura 16 pode-se observar a diminuição do erro quadrático médio, entre os resultados conhecidos e os resultados fornecidos pela rede, ao longo do seu treinamento.

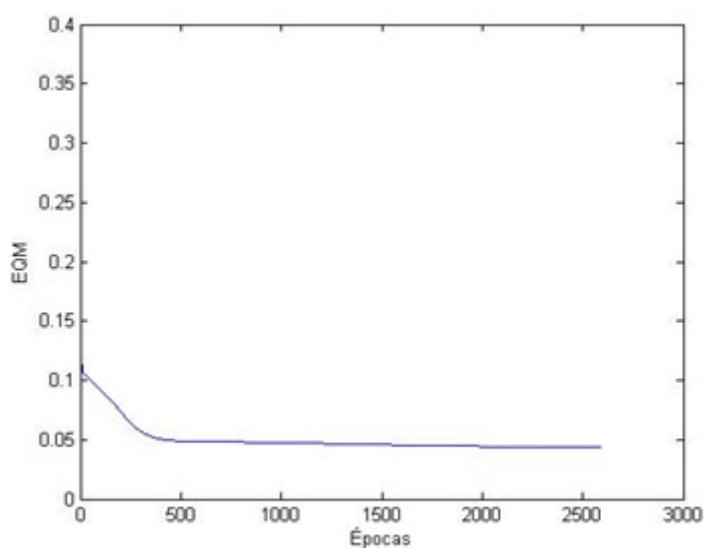


Figura 16 - Evolução do Erro Quadrático Médio por épocas de treinamento para PMC com 9 neurônios na camada intermediária.

Uma vez encontradas as matrizes de pesos sinápticos e limiar de ativação para as camadas neurais após o treinamento, o algoritmo de operação da PMC foi alimentado com os 8 pontos de teste e as posições do dispositivo móvel (suíno) foram calculadas, utilizando as novas matrizes de treinamento. Os resultados do monitoramento podem ser vistos na Tabela 2.

Tabela 2 - Comparação do posicionamento real com posicionamento experimental em centímetros – PMC com 9 neurônios na camada intermediária.

X (real)	Y (real)	X (exp)	Y (exp)	Erro X	Erro Y
156,00	179,00	183,88	214,44	27,88	35,44
149,00	223,00	176,24	257,36	27,24	34,36
209,00	248,00	177,64	310,64	31,36	62,64
222,00	217,00	186,24	217,84	35,76	0,84
243,00	200,00	189,56	188,16	53,44	11,84
251,00	157,00	195,16	143,20	55,84	13,80
209,00	160,00	193,04	164,72	15,96	4,72
162,00	134,00	185,40	132,72	23,40	1,28
Erro Médio:				33,86	20,62
Erro Relativo Médio em %:				16,70%	9,93%

Os resultados mostram que a arquitetura atual apresentou um desempenho um pouco superior em relação à implementação com 6 neurônios na camada intermediária, uma vez que o erro relativo médio para as coordenadas X e Y, foram de 16,70% e 9,93%, respectivamente.

Ao analisar ponto a ponto foi possível verificar que o desempenho de monitoramento, quanto ao eixo X é muito parecido em ambas as arquiteturas. Em contrapartida, observou-se uma melhora substancial em relação ao eixo Y, principalmente nos pontos intermediários do caminho realizado pelo dispositivo móvel (suíno).

Além disso, é importante salientar que o erro máximo encontrado neste monitoramento foi de 55,84 cm para o eixo X (6º ponto) e 62,64 cm para o eixo Y (3º ponto), sendo o erro médio para cada eixo, respectivamente, de 33,86 cm

e 20,62 cm. Esses valores são pequenos se forem levados em consideração as dimensões de um suíno em crescimento ou em terminação e, principalmente, se for considerado o tamanho da baia utilizada no experimento.

5.3. Topologia com 10 neurônios na camada intermediária

Na busca por melhores resultados foi realizado o treinamento com uma arquitetura dotada de 10 neurônios na camada intermediária. Assim, como nos treinamentos observados, a taxa de aprendizagem adotada foi de 0,01 e a precisão requerida para paragem do processo foi de 1×10^{-6} .

O algoritmo de treinamento foi alimentado com a matriz de entradas contendo 286 amostras e através da função rand(linha, coluna) foi fornecida a Matriz 9 e a Matriz 10 contendo os pesos sinápticos e limiar de ativação, para a camada intermediária (10 linhas x 4 colunas) e para a camada de saída (2 linhas x 11 colunas). Os valores contidos nas matrizes são:

MP1 =	0,9516	0,3015	0,0326	0,6448
	0,9203	0,7011	0,5612	0,3763
	0,0527	0,6663	0,8819	0,1909
	0,7379	0,5391	0,6692	0,4283
	0,2691	0,6981	0,1904	0,4820
	0,4228	0,6665	0,3689	0,1206
	0,5479	0,1781	0,4607	0,5895
	0,9427	0,1280	0,9816	0,2262
	0,4177	0,9991	0,1564	0,3846
	0,9831	0,1711	0,8555	0,5830

Matriz 9 - Pré-treinamento - Camada intermediária (10x4).

	Colunas 1 a 7						
MP2 =	0,2518	0,6171	0,8244	0,7302	0,5841	0,9063	0,8178
	0,2904	0,2653	0,9827	0,3439	0,1078	0,8797	0,2607
	Colunas 8 a 11						
	0,5944	0,4253	0,1615	0,4229			
	0,0225	0,3127	0,1788	0,0942			

Matriz 10 - Pré-treinamento - Camada de saída (2x11).

O algoritmo de treinamento realizou sua paragem após 2170 épocas, por ter alcançado precisão de $9,99 \times 10^{-6}$. Com isso, foi obtida nova Matriz 11 e Matriz 12 de pesos sinápticos e limiar de ativação para as camadas neurais, de tal forma que ao serem aplicadas na rede, forneceram erro quadrático médio no treinamento de 0,0449. Os novos valores das matrizes das camadas neurais são:

MP1 =	0,8103	0,4232	-0,9945	1,4531
	0,8175	-0,4837	3,3904	-1,7754
	0,2610	0,3374	1,1879	-0,3646
	1,0142	0,7838	-0,9056	1,2690
	0,3087	-0,2560	1,5835	-0,8453
	0,4969	0,7594	-0,4762	0,6432
	0,6146	0,3404	-0,5170	1,2317
	0,9355	-0,6266	3,6552	-1,7299
	1,1567	2,4771	-3,8210	2,2884
	0,9677	0,2757	0,7593	0,7042

Matriz 11 - Pós-treinamento - Camada intermediária (10x4).

	Colunas 1 a 7						
MP2 =	-1,0460	2,7940	-0,8809	-1,9212	-0,8215	0,5124	-2,6765
	0,0405	-1,7964	3,6941	0,7800	-1,8298	1,6792	-1,0974
	Colunas 8 a 11						
	2,2795	-0,2359	-2,7980	1,4249			
	-1,4032	3,7929	-4,9610	-0,0924			

Matriz 12 - Pós-treinamento - Camada de saída (2x11).

Durante o processo de treinamento foi observada a queda do erro quadrático médio, cuja diminuição ao longo das épocas pode ser vista na Figura 17.

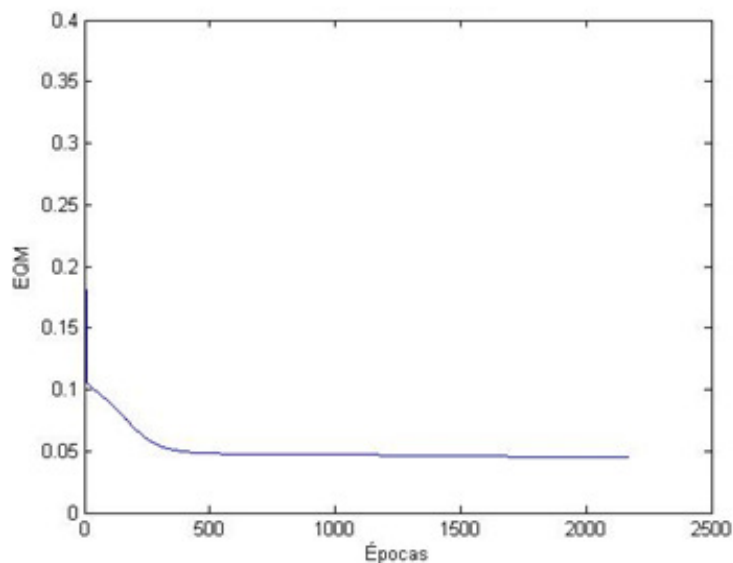


Figura 17 - Evolução do Erro Quadrático Médio por épocas de treinamento para PMC com 10 neurônios na camada intermediária.

Com a rede treinada e com as novas matrizes de pesos sinápticos e limiar de ativação das camadas neurais intermediária e de saída, o algoritmo de operação foi alimentado com os dados de entrada de valores de potência, entre o dispositivo móvel (suíno) e os três dispositivos fixos, bem como com os resultados desejados. Com isso, a rede PMC com 10 neurônios na camada intermediária forneceu o rastreamento das posições do dispositivo móvel (suíno), conforme dados mostrados através da Tabela 3.

Tabela 3 - Comparação do posicionamento real com posicionamento experimental em centímetros – PMC com 10 neurônios na camada intermediária.

X (real)	Y (real)	X (exp)	Y (exp)	Erro X	Erro Y
156,00	179,00	188,12	213,96	32,12	34,96
149,00	223,00	189,52	251,24	40,52	28,24
209,00	248,00	197,96	299,88	11,04	51,88
222,00	217,00	193,56	217,28	28,44	0,28
243,00	200,00	194,72	191,56	48,28	8,44
251,00	157,00	193,84	152,12	57,16	4,88
209,00	160,00	191,28	171,04	17,72	11,04
162,00	134,00	182,56	141,60	20,56	7,60
Erro Médio:				31,98	18,42
Erro Relativo Médio em %:				16,21%	9,14%

A análise da tabela revela que a atual arquitetura apresentou um desempenho próximo à anterior, mas com resultados ligeiramente melhores. Ao se comparar as arquiteturas com 10 e 9 neurônios na camada intermediária, percebe-se que a arquitetura com 10 neurônios possui uma taxa de acerto maior que a de 9 neurônios, na maioria das vezes.

Isso é evidenciado pelos resultados médios, cujos valores de erro absoluto foram de 31,98 cm para a coordenada X e 18,42 cm para a coordenada Y. Tal afirmação pode ser reforçada quando se observa o erro relativo médio: 16,21% e 9,14% para as coordenadas X e Y, respectivamente.

O valor máximo de erro encontrado para a coordenada X foi de 57,16 cm (6º ponto), levemente superior aos 55,84 cm da arquitetura analisada previamente. Com relação à coordenada Y, o erro máximo da arquitetura atual foi de 51,88 cm (3º ponto), desempenho substancialmente superior ao da implementação anterior (62,64 cm).

5.4. Topologia com 15 neurônios na camada intermediária

Nesta etapa, buscando-se melhores resultados, implementou-se uma rede PMC com 15 neurônios na camada escondida. Para esse caso, a Matriz 13 e a Matriz 14 foram inseridas, e cujos valores são:

MP1 =	0,5985	0,3251	0,6377	0,0067
	0,4709	0,1056	0,9577	0,6022
	0,6959	0,6110	0,2407	0,3868
	0,6999	0,7788	0,6761	0,9160
	0,6385	0,4235	0,2891	0,0012
	0,0336	0,0908	0,6718	0,4624
	0,0688	0,2665	0,6951	0,4243
	0,3196	0,1537	0,0680	0,4609
	0,5309	0,2810	0,2548	0,7702
	0,6544	0,4401	0,2240	0,3225
	0,4076	0,5271	0,6678	0,7847
	0,8200	0,4574	0,8444	0,4714
	0,7184	0,8754	0,3445	0,0358
	0,9686	0,5181	0,7805	0,1759
	0,5313	0,9436	0,6753	0,7218

Matriz 13 - Pré-treinamento - Camada intermediária (15x4).

	Colunas 1 a 7						
MP2 =	0,4735	0,3411	0,1917	0,2428	0,2691	0,1887	0,0911
	0,1527	0,6074	0,7384	0,9174	0,7655	0,2875	0,5762
	Colunas 8 a 14						
	0,6834	0,4257	0,6476	0,6358	0,2089	0,2362	0,6073
	0,5466	0,6444	0,6790	0,9452	0,7093	0,1194	0,4501
	Colunas 15 e 16						
	0,4587	0,7703					
	0,6619	0,3502					

Matriz 14 - Pré-treinamento - Camada de saída (2x16).

Assim, como nos casos anteriores, a taxa de aprendizado aplicada foi de 0,01 e a precisão requerida de 1×10^{-6} . Após 2436 épocas de treinamento a rede chegou a um erro quadrático médio de 0,0440, com precisão de $9,99 \times 10^{-7}$. Com isso, foram obtidas novas matrizes (Matriz 15 e Matriz 16) de pesos sinápticos e limiar de ativação para as camadas neurais, cujos valores são:

MP1 =	0,7323	-0,2509	2,7719	-1,6031
	0,7604	-0,8142	3,1681	-1,3032
	0,7689	-0,0255	1,2404	-0,5519
	0,8439	0,5774	-0,5761	1,4840
	0,5689	0,6086	-0,2671	0,4572
	0,3556	-0,5321	1,7970	-0,7698
	0,2778	-0,0877	1,0663	-0,1647
	0,3715	-0,4052	1,0606	-0,4365
	0,5094	-0,0359	0,1805	0,6668
	0,6244	-0,3225	2,2922	-1,1919
	0,3132	0,4333	0,4329	0,9805
	1,1250	0,9841	-1,5139	1,7482
	0,8665	1,2695	-1,5345	1,0662
	1,0990	0,0921	2,0972	-0,8787
	1,1683	1,7693	-2,9011	2,2614

Matriz 15 - Pós-treinamento - Camada intermediária (15x4).

Colunas 1 a 7							
MP2 =	-0,4514	-2,7864	1,1571	-1,0880	0,7236	-2,3406	1,2479
	0,9153	2,9205	3,1251	1,1735	-1,7306	-0,8538	1,6518
Colunas 8 a 14							
	0,9017	2,2548	3,4252	-0,5057	0,6544	0,0981	-3,4746
	0,5951	0,9929	-0,4464	2,4355	-0,7126	-2,7260	-2,3393
Colunas 15 e 16							
	-2,7238	0,0591					
	1,9261	-4,1809					

Matriz 16 - Pós-treinamento - Camada de saída (2x16).

Nessa arquitetura, a rede teve o desempenho de seu aprendizado dado pela Figura 18.

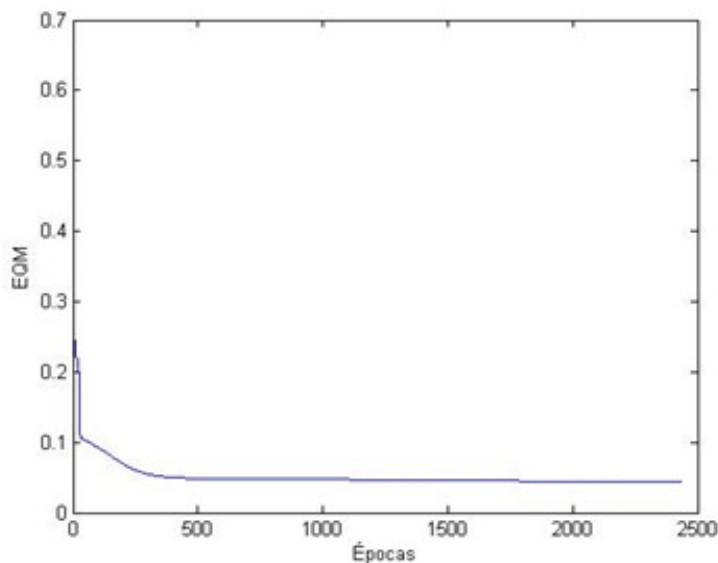


Figura 18 - Evolução do Erro Quadrático Médio por épocas de treinamento para PMC com 15 neurônios na camada intermediária.

Ao utilizar-se o algoritmo de operação para a presente arquitetura foi possível realizar o monitoramento das posições do dispositivo móvel (suíno), cujos resultados estão apresentados na Tabela 4.

Tabela 4 - Comparação do posicionamento real com posicionamento experimental em centímetros – PMC com 15 neurônios na camada intermediária.

X (real)	Y (real)	X (exp)	Y (exp)	Erro X	Erro Y
156,00	179,00	182,56	216,76	26,56	37,76
149,00	223,00	179,52	255,04	30,52	32,04
209,00	248,00	183,88	304,48	25,12	56,48
222,00	217,00	187,64	218,12	34,36	1,12
243,00	200,00	191,12	190,16	51,88	9,84
251,00	157,00	194,32	149,00	56,68	8,00
209,00	160,00	190,24	170,36	18,76	10,36
162,00	134,00	182,96	141,16	20,96	7,16
Erro Médio:				33,11	20,35
Erro Relativo Médio em %:				16,36%	10,07%

A análise dos resultados mostra que a arquitetura com 15 neurônios na camada intermediária, apresenta resultados muito semelhantes aos encontrados na implementação com 9 neurônios, na camada intermediária.

Verificou-se que seu desempenho no rastreamento ponto a ponto é muitas vezes intermediário entre as duas últimas arquiteturas analisadas. Em alguns casos, como a coordenada Y do ponto P2, seu desempenho foi superior ao das anteriores. Em média, o erro absoluto na coordenadas X e Y, respectivamente, foram de 33,11 cm e 20,35 cm, ou seja, seu desempenho absoluto foi intermediário entre a arquitetura de 10 e 9 neurônios.

No entanto, seu desempenho foi tão próximo à arquitetura de 9 neurônios que, quanto ao erro relativo médio, os valores encontrados para as coordenadas X e Y, nessa ordem, foram de 16,36% e 10,07%, em relação a 16,70% e 9,93%, na mesma ordem de coordenadas, para a implementação com 9 neurônios intermediários.

Como se pode observar, a rede apresentou uma evolução na capacidade de rastreamento, à medida que o número de neurônios foi incrementado. No entanto, foi possível verificar que a elevação da camada neural intermediária para 15 neurônio levou a uma queda no desempenho da rede.

5.5. Topologia com 20 neurônios na camada intermediária

Para ratificar se essa queda de desempenho persistiria, implementou-se o algoritmo de treinamento para uma rede PMC com 20 neurônios na camada intermediária. Suas matrizes iniciais (Matriz 17 e Matriz 18) foram:

MP1 =	0,6620	0,5225	0,8611	0,4423
	0,4162	0,9937	0,4849	0,0196
	0,8419	0,2187	0,3935	0,3309
	0,8329	0,1058	0,6714	0,4243
	0,2564	0,1097	0,7413	0,2703
	0,6135	0,0636	0,5201	0,1971
	0,5822	0,4046	0,3477	0,8217
	0,5407	0,4484	0,1500	0,4299
	0,8699	0,3658	0,5861	0,8878
	0,2648	0,7635	0,2621	0,3912
	0,3181	0,6279	0,0445	0,7691
	0,1192	0,7720	0,7549	0,3968
	0,9398	0,9329	0,2428	0,8085
	0,6456	0,9727	0,4424	0,7551
	0,4795	0,1920	0,6878	0,3774
	0,6393	0,1389	0,3592	0,2160
	0,5447	0,6963	0,7363	0,7904
	0,6473	0,0938	0,3947	0,9493
	0,5439	0,5254	0,6834	0,3276
	0,7210	0,5303	0,7040	0,6713

Matriz 17 - Pré-treinamento - Camada intermediária (20x4).

MP2 =	Colunas 1 a 7						
	0,4386	0,7689	0,8620	0,5144	0,5880	0,1999	0,7487
	0,8335	0,1673	0,9899	0,8843	0,1548	0,4070	0,8256
	Colunas 8 a 14						
	0,7900	0,5341	0,1117	0,6787	0,1897	0,1476	0,8507
	0,3185	0,0900	0,1363	0,4952	0,4950	0,0550	0,5606
	Colunas 15 a 21						
	0,9296	0,5828	0,8790	0,0005	0,6126	0,5277	0,8013
	0,6967	0,8154	0,9889	0,8654	0,9900	0,4795	0,2278

Matriz 18 - Pré-treinamento - Camada de saída (2x21).

Após o treinamento, a precisão encontrada foi de $9,98 \times 10^{-7}$, com erro quadrático médio de 0,0461. O processo de aprendizado da rede alcançou esses valores após 1597 épocas de treinamento e forneceu a Matriz 19 e a Matriz 20:

MP1 =	0,6605	0,7108	0,5139	0,7694
	0,5852	0,3658	1,3314	-0,9113
	0,7218	-0,3421	2,0141	-0,7777
	0,8483	0,1487	0,7066	0,4051
	0,4576	-0,2672	1,4568	-0,5230
	0,5039	-0,6310	2,5640	-1,2199
	0,6339	0,4133	-0,3452	1,2084
	0,6265	0,6606	-0,9639	1,1056
	0,9839	0,4986	-0,4239	1,4737
	0,2735	0,7069	-0,0746	0,5755
	0,4156	0,4962	-0,7360	1,0973
	0,2399	0,9118	0,1788	0,7526
	1,1026	0,8978	-0,9293	1,3273
	0,6553	0,8283	0,0622	0,9454
	0,5727	-0,5231	2,2795	-0,9826
	0,4662	-0,6165	2,5271	-1,2444
	0,5629	0,3567	1,0376	0,4002
	0,6603	-0,6884	1,7270	-0,1820
	0,6153	0,3670	0,8739	0,0688
	0,7687	0,6444	0,1060	1,0781

Matriz 19 - Pós-treinamento - Camada intermediária (20x4),

MP2 =	Colunas 1 a 7						
	0,2995	-0,5562	-3,1779	0,4303	0,9386	-0,0660	0,5588
	1,1539	-0,5180	1,4131	2,1187	0,0852	1,2873	2,7883
	Colunas 8 a 14						
	1,7650	0,2590	1,2869	-1,1024	0,4888	-2,0302	0,2268
	-1,1856	-1,6856	-1,4831	-0,6673	-1,3638	-0,8791	-1,7283
	Colunas 15 a 21						
	-0,5265	0,4109	0,6515	-0,5384	3,3472	-0,9670	0,3908
	-0,7571	2,3666	2,8387	0,3981	1,6576	0,3764	-0,9189

Matriz 20 - Pós-treinamento - Camada de saída (2x21),

A diminuição do erro quadrático médio ao longo do treinamento pode ser verificada na Figura 19.

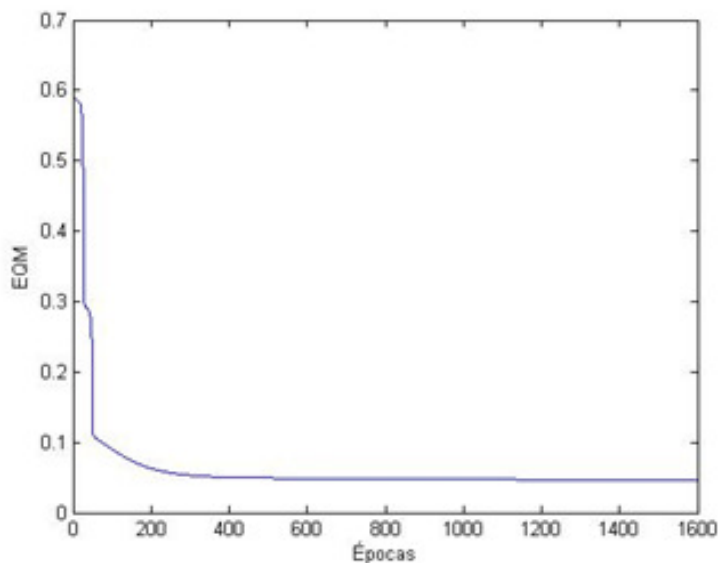


Figura 19 - Evolução do Erro Quadrático Médio por épocas de treinamento para PMC com 20 neurônios na camada intermediária.

Com a rede treinada, o algoritmo de operação foi aplicado, sendo obtidos os dados de monitoramento dados pela Tabela 5.

Tabela 5 - Comparação do posicionamento real com posicionamento experimental em centímetros – PMC com 20 neurônios na camada intermediária.

X (real)	Y (real)	X (exp)	Y (exp)	Erro X	Erro Y
156,00	179,00	180,52	218,88	24,52	39,88
149,00	223,00	181,96	244,68	32,96	21,68
209,00	248,00	189,04	280,28	19,96	32,28
222,00	217,00	187,24	219,04	34,76	2,04
243,00	200,00	190,00	199,92	53,00	0,08
251,00	157,00	190,36	171,68	60,64	14,68
209,00	160,00	185,60	187,04	23,40	27,04
162,00	134,00	178,68	167,24	16,68	33,24
Erro Médio:				33,24	21,37
Erro Relativo Médio em %:				16,31%	12,13%

Os resultados obtidos nessa implementação vem ao encontro da discussão anterior, sobre a queda de desempenho ocorrida após o último incremento no número de neurônios na camada intermediária.

Os valores das médias dos erros absolutos (coordenadas X e Y, nessa ordem) são maiores para a arquitetura atual, sendo 33,24 cm e 21,37 cm contra 33,11 cm e 20,35 cm, da rede com 15 neurônios intermediários.

Quanto ao desempenho no erro relativo médio, verificou-se ligeira melhora no desempenho na localização da coordenada X (16,31% ante 16,36%) e uma piora considerável para a localização no eixo Y (12,13% ante 10,07%). No entanto, mesmo com a queda de desempenho, os valores das médias de erros absolutos da arquitetura atual são melhores que os encontrados na rede com 9 neurônios intermediários, exceto quanto ao erro relativo médio da coordenada Y.

5.6. Fechamento dos resultados apresentados

Após a realização de treinamentos e operações com diversas topologias os resultados obtidos foram sintetizados na Tabela 6.

Tabela 6 - Comparação de desempenho de treinamentos e operação para as 5 arquiteturas de rede PMC utilizadas.

	Épocas	EQM	Precisão	Erro absoluto X (cm)	Erro absoluto Y (cm)	Erro relativo X	Erro relativo Y
6 neurônios na camada intermediária	561	0,0492	9,93E-7	36,10	24,23	17,81%	14,07%
9 neurônios na camada intermediária	2588	0,0435	9,99E-7	33,86	20,62	16,70%	9,93%
10 neurônios na camada intermediária	2170	0,0449	9,99E-7	31,98	18,42	16,21%	9,14%
15 neurônios na camada intermediária	2436	0,0440	9,99E-7	33,11	20,35	16,36%	10,07%
20 neurônios na camada intermediária	1597	0,0461	9,98E-7	33,24	21,37	16,31%	12,13%

Uma vez que a proposta do trabalho é rastrear as diferentes localizações do dispositivo móvel (suíno), elaborou-se a Tabela 7, em que é possível observar o deslocamento real do animal monitorado, saindo do ponto inicial P1 e deslocando-se até o ponto final P8.

Tabela 7 - Comparação da distância real e distâncias estimadas em centímetros para cada arquitetura de rede treinada.

	P1-P2	P2-P3	P3-P4	P4-P5	P5-P6	P6-P7	P7-P8	Distância total
Deslocamento real	44,56	65,00	33,60	27,00	43,72	42,12	53,72	309,76
Deslocamento experimental 6 neurônios na camada intermediária	6,00	18,08	23,16	6,52	8,56	5,96	13,40	81,76
Deslocamento experimental 9 neurônios na camada intermediária	43,60	53,28	93,20	29,88	45,28	21,60	32,88	319,68
Deslocamento experimental 10 neurônios na camada intermediária	37,32	49,36	82,72	25,72	39,48	19,08	30,72	284,36
Deslocamento experimental 15 neurônios na camada intermediária	38,36	49,64	86,44	28,16	41,28	21,76	30,12	295,80
Deslocamento experimental 20 neurônios na camada intermediária	25,80	36,32	61,28	19,32	28,24	16,08	20,96	208,04

É possível observar que as arquiteturas com 6 e 20 neurônios na camada intermediária são as que apresentaram maiores desvios nos cálculos das distâncias percorridas. Considerando que essas implementações são as que apresentaram maiores erros de monitoramento, torna-se natural entender porque os cálculos de distância realizados pelas mesmas tiveram desempenho inferior.

Além disso, a Tabela 7 mostra que a arquitetura dotada de 9 neurônios foi a que chegou mais próxima das distâncias reais percorridas. No entanto, é

importante lembrar de que essa implementação foi inferior às arquiteturas com 10 e 15 neurônios na camada intermediária, quando o erro da localização foi calculado.

Esse desempenho superior ocorreu por conta de sua característica de rastreamento ter fornecido pontos que, embora apresentassem maior erro em relação ao ponto real, eram equidistantes ao próximo ponto. Em outras palavras, a arquitetura com 9 neurônios encontrou pontos com desvios maiores, mas cuja distância em relação ao ponto anterior era semelhante, em relação à distância real. Por isso, o destaque de desempenho aqui observado não constitui fator para uma classificação superior em desempenho de rastreamento, em relação às duas últimas implementações citadas.

Diante dessas análises, tornou-se evidente que a rede neural que melhor extraiu as características dos dados de treinamento apresentado e que conseguiu realizar a generalização com maior fidelidade foi a Perceptron Multicamadas, com 10 neurônios na camada neural intermediária, que pode ser vista na Figura 20.

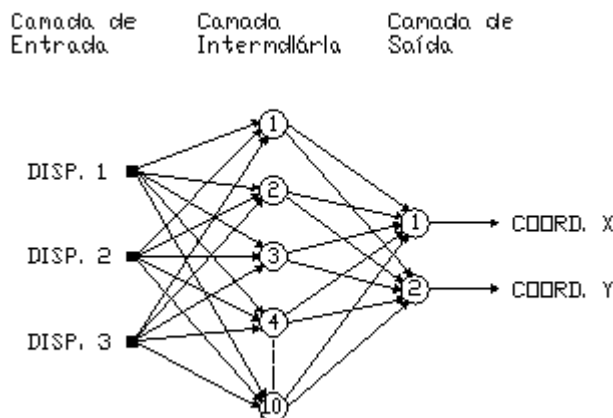


Figura 20 - Topologia de Rede com 10 neurônios na camada intermediária.

Encontrada a melhor rede, torna-se interessante observar como seria seu desempenho se fossem alteradas algumas características da rede, como taxas de aprendizado e precisão requerida para parada do algoritmo de treinamento (*Backpropagation*).

5.7. Otimização dos parâmetros de treinamento

Dessa forma, foram realizados novos treinamentos e operações, mantendo fixa a precisão em 1×10^{-6} e alterando para cada treino, o valor da taxa de aprendizagem. Os resultados desses treinamentos podem ser observados na Tabela 8.

Tabela 8 - Comparação de desempenho de treinamento e operação para as 5 taxas de aprendizado utilizadas na rede PMC com 10 neurônios intermediários.

Aprendizado	Épocas	EQM	Precisão	Erro absoluto		Erro relativo	
				X (cm)	Y (cm)	X	Y
0,1	272	0,0809	9,42E-6	41,17	114,96	18,42%	65,83%
0,05	170	0,0707	5,92E-6	40,72	88,10	18,53%	51,40%
0,01	2170	0,0449	9,99E-6	31,98	18,415	16,21%	9,14%
0,005	1287	0,0470	9,98E-6	35,93	23,95	17,57%	13,07%
0,001	4972	0,0471	9,99E-6	36,30	27,30	17,70%	13,76%

O primeiro treinamento realizado com 10 neurônios intermediários, no momento em que se buscava a melhor topologia para o monitoramento e rastreamento do dispositivo móvel (suíno) foi realizado com taxa de aprendizado 0,01. Nessa configuração, quando a taxa de aprendizado aumenta, o algoritmo atinge mais rapidamente a precisão 1×10^{-6} requerida. O fato inverso ocorre quando a aprendizagem é diminuída. Nesse caso, o algoritmo atualiza as matrizes de pesos sinápticos e limiar de ativação a passos menores e, devido a isso, o número de épocas, e conseqüentemente, o custo computacional é maior.

Em ambas as alterações observou-se uma queda de desempenho, o que leva a conclusão de que a melhor taxa de aprendizagem é 0,01.

Encontrada a melhor taxa de aprendizagem, foram realizados novos treinamentos e operações alterando-se a precisão requerida para paragem do algoritmo de treinamento. Os resultados dos treinamentos e das operações podem ser observados na Tabela 9.

Tabela 9 - Comparação de desempenho de treinamento e operação para as 5 precisões de aprendizado utilizadas na rede PMC com 10 neurônios intermediários.

Precisão	Épocas	EQM	Precisão	Erro absoluto X (cm)	Erro absoluto Y (cm)	Erro relativo X	Erro relativo Y
1×10^{-2}	12	0,1102	6,80E-3	33,17	58,14	17,80%	35,63%
1×10^{-4}	305	0,0576	9,90E-5	36,03	32,21	17,87%	19,66%
1×10^{-6}	2170	0,0449	9,99E-6	31,98	18,415	16,21%	9,14%
1×10^{-8}	7828	0,0389	9,83E-9	29,96	31,03	14,95%	15,37%
1×10^{-10}	10898	0,0393	8,87E-11	34,61	31,15	16,88%	15,28%

A análise da tabela mostra que a diminuição da precisão, em relação à precisão inicial de 1×10^{-6} , leva a treinamentos com menores erros quadráticos médios a custo de uma elevação substancial no número de épocas de treinamento, ou seja, de tempo de processamento (maior necessidade

computacional). Quando a precisão é aumentada ocorre o efeito inverso, a rede alcança a precisão requerida de modo mais rápido à custa de um erro quadrático médio maior. No entanto, no aspecto operacional, os novos parâmetros não foram capazes de fornecer melhores respostas para o rastreamento do dispositivo móvel.

Essa afirmação ganha força quando são observados os dados da Tabela 10. Como se pode observar, nenhuma das alterações realizadas nos parâmetros de taxa de aprendizagem e precisão foram capazes de fornecerem resultados de qualidade superior à primeira implementação com 10 neurônios na camada intermediária.

Tabela 10 - Comparação da distância real e distâncias estimadas em centímetros para diferentes taxas de aprendizado e precisões de rede PMC com 10 neurônios intermediários.

		P1-P2	P2-P3	P3-P4	P4-P5	P5-P6	P6-P7	P7-P8	Distância total
Deslocamento real		44,56	65,00	33,60	27,00	43,72	42,12	53,72	309,76
Aprend, Precisão									
0,01	1×10^{-6}	37,32	49,36	82,72	25,72	39,48	19,08	30,72	284,36
0,1	1×10^{-6}	7,56	11,04	16,48	4,84	8,12	3,88	8,20	60,08
0,05	1×10^{-6}	4,52	11,20	14,08	3,52	5,52	2,76	8,40	50,00
0,005	1×10^{-6}	6,2	18,72	24,04	6,84	8,8	6,32	13,72	84,60
0,001	1×10^{-6}	2,76	13,96	16,52	4,96	5,56	5,12	11,36	60,24
0,01	1×10^{-2}	1,72	0,64	1,04	0,40	0,64	1,16	0,60	6,24
0,01	1×10^{-4}	0,88	7,28	7,52	2,04	2,32	2,12	6,60	28,80
0,01	1×10^{-8}	60,08	60,28	106,76	35,56	61,60	22,16	47,12	393,56
0,01	1×10^{-10}	63,00	59,84	104,40	32,56	63,12	17,16	46,00	386,08

Dessa forma, a rede *Perceptron* Multicamadas com 10 neurônios na camada intermediária, utilizando taxa de aprendizagem de 0,01 e precisão de 1×10^{-6} mostrou-se a mais interessante, quanto aos aspectos de treinamento e de operação para o rastreamento do dispositivo móvel (suíno).

Em Wang et al (2010), observam-se um estudo semelhante através da medição de RSSI Bluetooth em uma área retangular de 6m por 8m, cuja distância foi estimada através do método dos mínimos quadrados, aplicados aos valores de potência obtidos de quatro dispositivos fixos (um em cada vértice da área).

Ao testar a localização de um ponto instalado na coordenada $X=3m$ e $Y=4m$, o método desenvolvido retornou a posição $X=2,78m$ e $Y=4,21m$. Comparativamente com a proposta desse trabalho, a melhor implementação de rede neural realizada, com 10 neurônios na camada intermediária obteve, em média, desvios de 31,98cm para a coordenada X e 18,42cm para a coordenada Y, ao passo que Wang et al. (2010) obteve desvios de 22,0cm para a coordenada X e 21,0cm para a coordenada Y, o que revela desempenhos bastante semelhantes.

Por outro lado, é importante salientar que a tomada de dados para treinamento da rede neural foi realizado pelo espaçamento dos pontos em 25cm, o que já possibilitou resultados com alta exatidão. Um treinamento realizado com maior número de pontos obtidos através do espaçamento menor entre os pontos de treinamento, possivelmente permitiria a obtenção de dados substancialmente melhores.

6. CONCLUSÃO

Quanto aos objetivos pretendidos, os resultados mostraram que a rede neural do tipo *Perceptron* Multicamadas apresentou-se como um algoritmo eficiente na interpretação dos dados obtidos através dos sinais disponibilizados pela tecnologia RSSI da transmissão *Bluetooth*.

Portanto, a utilização da tecnologia *Bluetooth* associada a redes neurais artificiais, consiste num sistema eficiente para monitoramento e rastreamento suíno em seu espaço de manejo (baia), possibilitando alimentar o sistema com os dados de RSSI *Bluetooth*, sendo devolvida pela rede PMC o posicionamento animal nas coordenadas X e Y.

7. SUGESTÕES PARA FUTUROS TRABALHOS

Levando-se em conta os ruídos presentes nas medidas, que são inerentes a todo processo de amostragem de dados realizado, principalmente quando existem sinais, cuja resposta apresenta naturalmente pequenas oscilações, como é o caso das transmissões de ondas, fica evidenciada a eficiência da aplicação das Redes Neurais Artificiais *Perceptron* Multicamadas (PMC), para o monitoramento e o rastreamento da posição dos animais em seu espaço de manejo (baia).

Os resultados obtidos foram relevantes, com um erro relativo médio de 16,21% para as coordenadas de eixo X e 9,14%, para as coordenadas de eixo Y. Junto a esses resultados obtidos, é importante salientar que o espaço experimental utilizado foi dividido em pontos distantes 25 cm em coordenadas X e Y, permitindo a geração de uma base de dados com 286 amostras, que é uma quantidade bastante razoável para o treinamento da rede neural utilizada. No entanto, a quantidade de informação relativa ao comportamento da potência de RSSI da tecnologia *Bluetooth* não aproveitada nesse intervalo de espaço é considerável.

Por isso, a realização de amostragens com pontos mais próximos entre si constitui uma implementação interessante para trabalhos futuros que poderão fornecer resultados ainda mais satisfatórios.

Ao longo do desenvolvimento desse sistema de monitoramento e rastreamento ficou evidenciado, que não se trata de uma tecnologia absoluta, ou engessada. Com isso, as diversas possibilidades de instalação de pontos fixos em diferentes locais e com diferentes topologias de rede, com mais

camadas intermediárias, também apresentam potenciais para fornecerem resultados satisfatórios.

Junte-se a isso, o fato de existir um protocolo de comunicação entre o animal, representado pelo dispositivo móvel, e as estações fixas. Trata-se de um cenário que permite a utilização de sensores para transmissão de dados do animal monitorado, como temperatura, por exemplo, que permitirá futuramente o estudo da movimentação animal, correlacionada ao seu comportamento, ao seu ambiente e à qualidade e produtividade animal, constituindo assim um grande horizonte de pesquisas a serem realizadas.

REFERÊNCIAS BIBLIOGRÁFICAS

BILLO, E. A. **Uma pilha de protocolos Bluetooth adaptável à aplicação**. 2003. 82 f. Monografia (Bacharelado em Ciência da Computação) – Universidade Federal de Santa Catarina, Florianópolis, 2003.

BLUETOOTH SPECIAL INTEREST GROUP, **Specification of the Bluetooth System**, 2001. Disponível em <www.bluetooth.com>. Acesso em: 05 jan. 2014.

BLUETOOTH SPECIAL INTEREST GROUP, Board of Directors. Disponível em: <<http://www.bluetooth.com/Pages/Board-of-Directors.aspx>>.. Acesso em: 05 jan. 2014.

BRAGA, Antônio de Pádua; CARVALHO, André Ponce de Leon F. de; LUDERMIN, Teresa Bernarda, **Redes Neurais Artificiais: Teoria e Aplicações**, Rio de Janeiro: LTC, 2000.

CROSSMAN, A. R.; NEARY, D. **Neuroanatomia Ilustrada**, 4. Ed. Rio de Janeiro: Elsevier, 2011.

AMARAL, Armando Lopes do et al., **Boas Práticas de Produção de Suínos**, Empresa Brasileira de Pesquisa Agropecuária (EMBRAPA), Concórdia, 2006.

FOROUZAN, Behrouz A. **Data Communications and Networking**. 4. Ed. New York: McGraw Hill, 2007.

HAYKIN, Simon, **Redes Neurais: princípios e prática**. 2. Ed. Porto Alegre: Bookman, 2001.

KANSAL, Aman. **Bluetooth Primer**. Los Angeles: Red-M. 2002.

KOVÁČZ, Z. L. **Redes neurais artificiais: Fundamentos e aplicações**. 4. Ed. São Paulo: Editora Livraria da Física, 2006.

MIGUENS, ALTINEU PIRES, Navegação: a ciência e a arte, Vol III, Capítulo 34. Disponível em: <https://www.mar.mil.br/dhn/dhn/quadros/livro_tres.html>. Acesso: 05 dez. 2013.

SARTOR, V.; SOUZA. F. C.; TINÔCO, I. F. F. . **INFORMAÇÕES BÁSICAS PARA PROJETOS DE COSNTRUÇÕES RURAIS** - UNIDADE 2 - SUINOS. 2006. (Desenvolvimento de material didático ou instrucional - APOSTILA)

SILVA, I. N.; SPATTI, D. H.; FLAUZINO, R. A. **Redes neurais artificiais**: para engenharia e ciências aplicadas. São Paulo: Artliber, 2010.

SEVEGNANI, Kelly B. et al. Zootecnia de Precisão: análise de imagens no estudo do comportamento de frangos de corte em estresse térmico. **Revista Brasileira de Engenharia Agrícola e Ambiental**, Campina Grande, v. 9, n. 1, pag. 115-9, 2005.

STALLINGS, W. **Data and computer communications**. 5. Ed. New Jersey: Prentice Hall, 1997.

TANEMBAUM, A. S. **Computer Networks**. 4. Ed. Amsterdam: Prentice Hall, 2002.

TORRES, G. **Redes de computadores**: Curso completo. Rio de Janeiro: Axcel Books, 2001.

UNIÃO EUROPÉIA, **COMUNICAÇÃO DA COMISSÃO AO CONSELHO E AO PARLAMENTO EUROPEU**. Relativa à proteção dos suínos em sistema de criação intensiva, tendo em conta particularmente o bem-estar das porcas criadas em vários graus de confinamento e em grupo. Disponível em: <http://ec.europa.eu/food/fs/aw/aw_legislation/pigs/comm_pt.pdf>. Acesso: 28 set. 2013.

UNITED NATIONS, STATISTICS DIVISION, 2012,. Disponível em:

<<http://unstats.un.org/unsd/demographic/products/socind/default.htm>> Acesso:

05 jan. 2014.

WANG, Y. et al. "Bluetooth Indoor Positioning using RSSI and Least Square Estimation", **IEEE ICFCC**, Shanghai, China, 2010.

APÊNDICE A — Algoritmo de Treinamento da Rede Perceptron Multicamadas

```

%Programa de Treinamento MLP - Regra Delta Generalizada
%Entrada de dados para treinamento
NX=input('Entre com o número de entradas: ')
N1=input('Entre com o número de neurônios da primeira camada escondida: ')
N2=input('Entre com o número de neurônios da camada de saída: ')
MX= input('Entre com a matriz de treinamento: ')
MD= input('Entre com os resultados desejados: ')
MP1= input('Entre com a matriz de pesos iniciais (10x4): ')
MP2= input('Entre com a matriz de pesos iniciais (2x11): ')
P= input('Entre com o números de amostras apresentado: ')
AP= input('Entre com a taxa de aprendizado: ')
PR= input('Entre com a precisão desejada: ')
%Inicialização de variáveis
erro=1;
epoca=0;
EQM=0;
EQM2=0;
while erro>=PR
    EQM=EQM2;

    %Fase forward
    for am=1:P
        I1=MP1*MX(:,am);
        Y1=0;
        for a=1:N1
            Y1(a,1)=1/(1+exp(-I1(a,1)));
        end
        Y1=[-1;Y1];
        I2=MP2*Y1;
        for b=1:N2
            Y2(b,1)=1/(1+exp(-I2(b,1)));
        end

        %Fase backward
        for a=1:N2
            S2(a,1)=(MD(a,am)-Y2(a,1))*(1/(1+exp(-I2(a,1))))*(1-(1/(1+exp(-I2(a,1)))));
        end
        MP2=MP2+AP*S2*Y1';
        for a=1:N1
            b=a+1;
            for c=1:N2
                SOMA1=S2(c,1)*MP2(c,b);
            end
            S1(a,1)=SOMA1*(1/(1+exp(-I1(a,1))))*(1-(1/(1+exp(-I1(a,1)))));
        end
        for i=1:N1

```

```

    MP1(i,:)=MP1(i,:)+AP*S1(i,1)*MX(:,am)';
    end
end
epoca=epoca+1;
EQM2=0;
for am=1:P
    I1=MP1*MX(:,am);
    Y1=0;
    for a=1:N1
        Y1(a,1)=1/(1+exp(-I1(a,1)));
    end
    Y1=[-1;Y1];
    I2=MP2*Y1;
    for b=1:N2
        Y2(b,1)=1/(1+exp(-I2(b,1)));
    end
    SOMA2=0;
    for c=1:N2
        SOMA2=SOMA2+MD(c,am)-Y2(c,1);
    end
    EQM2=EQM2+0.5*SOMA2^2;
end
EQM2=EQM2/P;
erro=abs(EQM2-EQM);
EQMPLOT(epoca)=EQM2;
k(epoca)=epoca;
end

disp('A Matriz de Pesos Final é: ')
MP1
MP2
disp('O número de épocas é: ')
epoca
disp('A precisão alcançada foi: ')
erro
disp('O erro quadrático médio encontrado foi: ')
EQM2
plot(k, EQMPLOT)
title('Erro Quadrático Médio')
xlabel('Épocas')
ylabel('EQM')

```

APÊNDICE B — Algoritmo de Operação da Rede Perceptron Multicamadas

```

%Programa MLP - Operação
%Inicialização de Variáveis
erro=[0;0];
errorelmed=[0;0];
dy=0
dy2=0
disty1=0
disty2=0

%Entrada de dados para operação
X_AM=input('Entre com o número de amostras a ser apresentado: ')
X_OP=input('Entre com a Matriz de entradas a ser classificada: ')
Y_OP=input('Entre com a Matriz de saídas conhecidas para as entradas dadas: ')
NX=input('Entre com o número de entradas: ')
N1=input('Entre com o número de neurônios da primeira camada escondida: ')
N2=input('Entre com o número de neurônios da camada de saída: ')
PESOS_OP1=input('Entre com a Matriz de Pesos MP1 obtida após treinamento: ')
PESOS_OP2=input('Entre com a Matriz de Pesos MP2 obtida após treinamento: ')
('Aplicando a função de ativação')
for am=1:X_AM
    I1OP=PESOS_OP1*X_OP(:,am);
    Y1OP=0;
    for a=1:N1
        Y1OP(a,1)=1/(1+exp(-I1OP(a,1)));
    end
    Y1OP=[-1;Y1OP];
    I2OP=PESOS_OP2*Y1OP;
    for b=1:N2
        Y2OP(b,am)=1/(1+exp(-I2OP(b,1)));
        erro(b,1)=erro(b,1)+abs(Y2OP(b,am)-Y_OP(b,am));
    end
end

%Respostas da operação
disp('Vetor com os resultados')
Y2OP
disp('Erro relativo médio: ')
errorelmed=erro*100/X_AM

%Cálculo das Distâncias percorridas / Distância total
for i=1:X_AM-1
    dy(1,i)=((Y_OP(1,i+1)-Y_OP(1,i))^2 + (Y_OP(2,i+1)-Y_OP(2,i))^2)^0.5;
    disty1=disty1+dy(1,i);
    dy2(1,i)=((Y2OP(1,i+1)-Y2OP(1,i))^2 + (Y2OP(2,i+1)-Y2OP(2,i))^2)^0.5;

```

```
    disty2=disty2+dy2(1,i);
end
disp('Distâncias parciais percorridas real')
dy
disp('Distâncias parciais percorridas experimental')
dy2
disp('Distância total percorrida real')
disty1
disp('Distância total percorrida experimental')
disty2

plot(Y2OP(1,:),Y2OP(2,:), '*')
hold
plot(Y_OP(1,:),Y_OP(2,:), 'o')
```

**APÊNDICE C — Tabela de dados normalizados para Treinamento da Rede
Perceptron Multicamadas**

Tabela 11 - Dados para treinamento da rede PMC

Amostra	Sinal de Entrada para o Limiar de Ativação	Entradas			Saídas	
		Disp, 1	Disp, 2	Disp, 3	Coord, X	Coord, Y
1	-1,0000	0,8218	0,7985	0,7960	0,0000	0,0000
2	-1,0000	0,8574	0,7931	0,8005	0,0625	0,0000
3	-1,0000	0,8485	0,8005	0,8119	0,1250	0,0000
4	-1,0000	0,8470	0,7891	0,8153	0,1875	0,0000
5	-1,0000	0,8436	0,7916	0,8074	0,2500	0,0000
6	-1,0000	0,8327	0,7950	0,8153	0,3125	0,0000
7	-1,0000	0,8327	0,7886	0,8198	0,3750	0,0000
8	-1,0000	0,8485	0,7995	0,8218	0,4375	0,0000
9	-1,0000	0,8312	0,7936	0,8223	0,5000	0,0000
10	-1,0000	0,8282	0,8010	0,8297	0,5625	0,0000
11	-1,0000	0,8208	0,7931	0,8307	0,6250	0,0000
12	-1,0000	0,8153	0,7896	0,8322	0,6875	0,0000
13	-1,0000	0,8149	0,7876	0,8356	0,7500	0,0000
14	-1,0000	0,8089	0,7866	0,8327	0,8125	0,0000
15	-1,0000	0,8079	0,7847	0,8277	0,8750	0,0000
16	-1,0000	0,8069	0,7842	0,8257	0,9375	0,0000
17	-1,0000	0,8040	0,7822	0,8243	1,0000	0,0000
18	-1,0000	0,8713	0,7871	0,7896	0,0000	0,0625
19	-1,0000	0,8738	0,7886	0,8054	0,0625	0,0625
20	-1,0000	0,8752	0,7891	0,8139	0,1250	0,0625
21	-1,0000	0,8757	0,7906	0,8178	0,1875	0,0625
22	-1,0000	0,8797	0,7911	0,8213	0,2500	0,0625
23	-1,0000	0,8767	0,7985	0,8223	0,3125	0,0625
24	-1,0000	0,8762	0,8000	0,8238	0,3750	0,0625
25	-1,0000	0,8743	0,8005	0,8361	0,4375	0,0625
26	-1,0000	0,8649	0,8015	0,8436	0,5000	0,0625
27	-1,0000	0,8520	0,7990	0,8475	0,5625	0,0625
28	-1,0000	0,8411	0,7975	0,8525	0,6250	0,0625
29	-1,0000	0,8287	0,7960	0,8574	0,6875	0,0625
30	-1,0000	0,8243	0,7901	0,8614	0,7500	0,0625
31	-1,0000	0,8203	0,7896	0,8569	0,8125	0,0625
32	-1,0000	0,8069	0,7881	0,8559	0,8750	0,0625
33	-1,0000	0,7995	0,7861	0,8550	0,9375	0,0625
34	-1,0000	0,7985	0,7856	0,8545	1,0000	0,0625
35	-1,0000	0,8757	0,7906	0,8064	0,0000	0,1250
36	-1,0000	0,8767	0,7926	0,8074	0,0625	0,1250
37	-1,0000	0,8807	0,7946	0,8099	0,1250	0,1250

Tabela 11 - Dados para treinamento da rede PMC (continuação)

Amostra	Sinal de Entrada para o Limiar de Ativação	Entradas			Saídas	
		Disp, 1	Disp, 2	Disp, 3	Coord, X	Coord, Y
38	-1,0000	0,8970	0,7960	0,8109	0,1875	0,1250
39	-1,0000	0,9104	0,7975	0,8153	0,2500	0,1250
40	-1,0000	0,8970	0,8000	0,8193	0,3125	0,1250
41	-1,0000	0,8901	0,8025	0,8277	0,3750	0,1250
42	-1,0000	0,8950	0,7899	0,8215	0,4375	0,1250
43	-1,0000	0,8872	0,7926	0,8326	0,5000	0,1250
44	-1,0000	0,8835	0,7923	0,8684	0,5625	0,1250
45	-1,0000	0,8570	0,7931	0,8684	0,6250	0,1250
46	-1,0000	0,8408	0,7916	0,8768	0,6875	0,1250
47	-1,0000	0,8199	0,7973	0,8788	0,7500	0,1250
48	-1,0000	0,8046	0,7950	0,8645	0,8125	0,1250
49	-1,0000	0,7991	0,7964	0,9027	0,8750	0,1250
50	-1,0000	0,8017	0,7923	0,8856	0,9375	0,1250
51	-1,0000	0,7913	0,7891	0,8757	1,0000	0,1250
52	-1,0000	0,8758	0,7934	0,8025	0,0000	0,1875
53	-1,0000	0,8520	0,7930	0,7929	0,0625	0,1875
54	-1,0000	0,9117	0,7925	0,8020	0,1250	0,1875
55	-1,0000	0,9428	0,7961	0,8138	0,1875	0,1875
56	-1,0000	0,9307	0,8027	0,8124	0,2500	0,1875
57	-1,0000	0,9477	0,7889	0,8331	0,3125	0,1875
58	-1,0000	0,9301	0,7907	0,8311	0,3750	0,1875
59	-1,0000	0,8850	0,7937	0,8422	0,4375	0,1875
60	-1,0000	0,8707	0,7919	0,8725	0,5000	0,1875
61	-1,0000	0,8614	0,7997	0,8710	0,5625	0,1875
62	-1,0000	0,8344	0,7988	0,8925	0,6250	0,1875
63	-1,0000	0,8174	0,7876	0,8955	0,6875	0,1875
64	-1,0000	0,8200	0,7957	0,9295	0,7500	0,1875
65	-1,0000	0,8069	0,8000	0,9440	0,8125	0,1875
66	-1,0000	0,7994	0,7939	0,9045	0,8750	0,1875
67	-1,0000	0,8061	0,7988	0,8718	0,9375	0,1875
68	-1,0000	0,7895	0,7975	0,8364	1,0000	0,1875
69	-1,0000	0,8614	0,8027	0,7984	0,0000	0,2500
70	-1,0000	0,8927	0,8008	0,7961	0,0625	0,2500
71	-1,0000	0,9243	0,7932	0,8116	0,1250	0,2500
72	-1,0000	0,9788	0,7975	0,8053	0,1875	0,2500
73	-1,0000	0,9524	0,7955	0,8082	0,3125	0,2500
74	-1,0000	0,9039	0,7928	0,8507	0,3750	0,2500
75	-1,0000	0,8671	0,7968	0,8361	0,4375	0,2500
76	-1,0000	0,8687	0,7977	0,8552	0,5000	0,2500
77	-1,0000	0,8601	0,7980	0,9022	0,5625	0,2500
78	-1,0000	0,8283	0,7949	0,9141	0,6250	0,2500

Tabela 11 - Dados para treinamento da rede PMC (continuação)

Amostra	Sinal de Entrada para o Limiar de Ativação	Entradas			Saídas	
		Disp, 1	Disp, 2	Disp, 3	Coord, X	Coord, Y
79	-1,0000	0,8128	0,8047	0,9656	0,6875	0,2500
80	-1,0000	0,8043	0,7950	0,9304	0,8125	0,2500
81	-1,0000	0,8094	0,8006	0,8965	0,8750	0,2500
82	-1,0000	0,8062	0,7943	0,8683	0,9375	0,2500
83	-1,0000	0,7928	0,8007	0,8369	1,0000	0,2500
84	-1,0000	0,8743	0,7946	0,7941	0,0000	0,3125
85	-1,0000	0,8636	0,8099	0,7958	0,0625	0,3125
86	-1,0000	0,8958	0,7978	0,8032	0,1250	0,3125
87	-1,0000	0,9475	0,7955	0,7935	0,1875	0,3125
88	-1,0000	0,9703	0,7983	0,7983	0,2500	0,3125
89	-1,0000	0,9431	0,8131	0,8423	0,3125	0,3125
90	-1,0000	0,9181	0,8158	0,8106	0,3750	0,3125
91	-1,0000	0,8686	0,8032	0,8203	0,4375	0,3125
92	-1,0000	0,8314	0,8010	0,8413	0,5000	0,3125
93	-1,0000	0,8176	0,7941	0,8699	0,5625	0,3125
94	-1,0000	0,8183	0,7943	0,9074	0,6250	0,3125
95	-1,0000	0,8171	0,7918	0,9512	0,6875	0,3125
96	-1,0000	0,8097	0,7981	0,9443	0,7500	0,3125
97	-1,0000	0,7988	0,7933	0,9052	0,8125	0,3125
98	-1,0000	0,8002	0,7930	0,8408	0,8750	0,3125
99	-1,0000	0,8020	0,7931	0,8446	0,9375	0,3125
100	-1,0000	0,7988	0,7958	0,8321	1,0000	0,3125
101	-1,0000	0,8611	0,7933	0,7941	0,0000	0,3750
102	-1,0000	0,8985	0,8059	0,7896	0,0625	0,3750
103	-1,0000	0,8851	0,8121	0,7947	0,1250	0,3750
104	-1,0000	0,9238	0,8025	0,8112	0,1875	0,3750
105	-1,0000	0,9136	0,8102	0,8101	0,2500	0,3750
106	-1,0000	0,8866	0,8067	0,8034	0,3125	0,3750
107	-1,0000	0,8772	0,7921	0,8300	0,3750	0,3750
108	-1,0000	0,8448	0,7921	0,8443	0,4375	0,3750
109	-1,0000	0,8128	0,8046	0,8522	0,5000	0,3750
110	-1,0000	0,8000	0,8131	0,8670	0,5625	0,3750
111	-1,0000	0,8088	0,7963	0,8948	0,6250	0,3750
112	-1,0000	0,8106	0,8135	0,9230	0,6875	0,3750
113	-1,0000	0,7950	0,8087	0,8978	0,7500	0,3750
114	-1,0000	0,7966	0,8119	0,8990	0,8125	0,3750
115	-1,0000	0,7990	0,8020	0,8798	0,8750	0,3750
116	-1,0000	0,7957	0,7993	0,8337	0,9375	0,3750
117	-1,0000	0,7936	0,8033	0,8260	1,0000	0,3750
118	-1,0000	0,8494	0,8013	0,7913	0,0000	0,4375
119	-1,0000	0,8725	0,7988	0,7970	0,0625	0,4375
120	-1,0000	0,8767	0,8090	0,7943	0,1250	0,4375

Tabela 11 - Dados para treinamento da rede PMC (continuação)

Amostra	Sinal de Entrada para o Limiar de Ativação	Entradas			Saídas	
		Disp, 1	Disp, 2	Disp, 3	Coord, X	Coord, Y
121	-1,0000	0,9059	0,8211	0,7970	0,1875	0,4375
122	-1,0000	0,8802	0,8121	0,8173	0,2500	0,4375
123	-1,0000	0,8567	0,8134	0,8215	0,3125	0,4375
124	-1,0000	0,8413	0,8011	0,8230	0,3750	0,4375
125	-1,0000	0,8213	0,8072	0,8354	0,4375	0,4375
126	-1,0000	0,8163	0,8080	0,8334	0,5000	0,4375
127	-1,0000	0,7985	0,8194	0,8720	0,5625	0,4375
128	-1,0000	0,8035	0,8036	0,8691	0,6250	0,4375
129	-1,0000	0,8045	0,8164	0,8950	0,6875	0,4375
130	-1,0000	0,7983	0,8184	0,8696	0,7500	0,4375
131	-1,0000	0,8037	0,8135	0,8705	0,8125	0,4375
132	-1,0000	0,8037	0,8171	0,8446	0,8750	0,4375
133	-1,0000	0,7946	0,7973	0,8240	0,9375	0,4375
134	-1,0000	0,7973	0,8081	0,8203	1,0000	0,4375
135	-1,0000	0,8510	0,8069	0,7973	0,0000	0,5000
136	-1,0000	0,8547	0,8134	0,7936	0,0625	0,5000
137	-1,0000	0,8485	0,8160	0,8099	0,1250	0,5000
138	-1,0000	0,8743	0,8203	0,8171	0,1875	0,5000
139	-1,0000	0,8676	0,8246	0,8139	0,2500	0,5000
140	-1,0000	0,8569	0,8315	0,8074	0,3125	0,5000
141	-1,0000	0,8389	0,8226	0,8324	0,3750	0,5000
142	-1,0000	0,8235	0,8104	0,8054	0,4375	0,5000
143	-1,0000	0,8243	0,8161	0,8235	0,5000	0,5000
144	-1,0000	0,8017	0,8304	0,8626	0,5625	0,5000
145	-1,0000	0,8106	0,8244	0,8537	0,6250	0,5000
146	-1,0000	0,7980	0,8401	0,8738	0,6875	0,5000
147	-1,0000	0,8027	0,8184	0,8520	0,7500	0,5000
148	-1,0000	0,8015	0,8281	0,8550	0,8125	0,5000
149	-1,0000	0,7983	0,8092	0,8257	0,8750	0,5000
150	-1,0000	0,7985	0,8085	0,8488	0,9375	0,5000
151	-1,0000	0,8053	0,7993	0,8104	1,0000	0,5000
152	-1,0000	0,8317	0,8176	0,8002	0,0000	0,5625
153	-1,0000	0,8443	0,8248	0,8002	0,0625	0,5625
154	-1,0000	0,8483	0,8333	0,8030	0,1250	0,5625
155	-1,0000	0,8496	0,8296	0,7983	0,1875	0,5625
156	-1,0000	0,8394	0,8397	0,8089	0,2500	0,5625
157	-1,0000	0,8300	0,8496	0,8062	0,3125	0,5625
158	-1,0000	0,8418	0,8530	0,8158	0,3750	0,5625
159	-1,0000	0,8208	0,8532	0,8153	0,4375	0,5625
160	-1,0000	0,8157	0,8460	0,8454	0,5000	0,5625
161	-1,0000	0,8124	0,8327	0,8567	0,5625	0,5625
162	-1,0000	0,7968	0,8564	0,8564	0,6250	0,5625

Tabela 11 - Dados para treinamento da rede PMC (continuação)

Amostra	Sinal de Entrada para o Limiar de Ativação	Entradas			Saídas	
		Disp, 1	Disp, 2	Disp, 3	Coord, X	Coord, Y
163	-1,0000	0,7990	0,8470	0,8416	0,6875	0,5625
164	-1,0000	0,8037	0,8454	0,8636	0,7500	0,5625
165	-1,0000	0,7923	0,8048	0,8384	0,8125	0,5625
166	-1,0000	0,7946	0,8147	0,8381	0,8750	0,5625
167	-1,0000	0,7903	0,8034	0,8148	0,9375	0,5625
168	-1,0000	0,7926	0,8003	0,8116	1,0000	0,5625
169	-1,0000	0,8389	0,8076	0,7995	0,0000	0,6250
170	-1,0000	0,8240	0,8069	0,7926	0,0625	0,6250
171	-1,0000	0,8381	0,8245	0,8048	0,1250	0,6250
172	-1,0000	0,8437	0,8290	0,7951	0,1875	0,6250
173	-1,0000	0,8357	0,8760	0,8126	0,2500	0,6250
174	-1,0000	0,8279	0,8854	0,8188	0,3125	0,6250
175	-1,0000	0,8178	0,8809	0,8252	0,3750	0,6250
176	-1,0000	0,8158	0,8725	0,8211	0,4375	0,6250
177	-1,0000	0,8093	0,8522	0,8352	0,5000	0,6250
178	-1,0000	0,8082	0,8836	0,8416	0,5625	0,6250
179	-1,0000	0,7954	0,8799	0,8485	0,6250	0,6250
180	-1,0000	0,7916	0,8515	0,8395	0,6875	0,6250
181	-1,0000	0,7948	0,8300	0,8289	0,7500	0,6250
182	-1,0000	0,7943	0,8188	0,8198	0,8125	0,6250
183	-1,0000	0,7939	0,8064	0,8199	0,8750	0,6250
184	-1,0000	0,0000	0,8000	0,8087	0,9375	0,6250
185	-1,0000	0,0000	0,7973	0,8036	1,0000	0,6250
186	-1,0000	0,8225	0,8037	0,7923	0,0000	0,6875
187	-1,0000	0,8208	0,8052	0,7840	0,0625	0,6875
188	-1,0000	0,8309	0,8262	0,7890	0,1250	0,6875
189	-1,0000	0,8205	0,8433	0,7915	0,1875	0,6875
190	-1,0000	0,8240	0,8547	0,7985	0,2500	0,6875
191	-1,0000	0,8196	0,8723	0,8017	0,3125	0,6875
192	-1,0000	0,8106	0,8965	0,7990	0,3750	0,6875
193	-1,0000	0,8047	0,9543	0,8176	0,4375	0,6875
194	-1,0000	0,8205	0,9705	0,8258	0,5000	0,6875
195	-1,0000	0,8035	0,9637	0,8357	0,5625	0,6875
196	-1,0000	0,7983	0,9091	0,8307	0,6250	0,6875
197	-1,0000	0,7938	0,8783	0,8277	0,6875	0,6875
198	-1,0000	0,7950	0,8186	0,8188	0,7500	0,6875
199	-1,0000	0,0000	0,8226	0,8161	0,8125	0,6875
200	-1,0000	0,0000	0,8146	0,8153	0,8750	0,6875
201	-1,0000	0,0000	0,7936	0,8059	0,9375	0,6875
202	-1,0000	0,0000	0,7933	0,7965	1,0000	0,6875
203	-1,0000	0,8050	0,8097	0,7970	0,0000	0,7500
204	-1,0000	0,8183	0,8029	0,7908	0,0625	0,7500

Tabela 11 - Dados para treinamento da rede PMC (continuação)

Amostra	Sinal de Entrada para o Limiar de Ativação	Entradas			Saídas	
		Disp, 1	Disp, 2	Disp, 3	Coord, X	Coord, Y
205	-1,0000	0,8239	0,8151	0,7907	0,1250	0,7500
206	-1,0000	0,8205	0,8488	0,7902	0,1875	0,7500
207	-1,0000	0,8196	0,8498	0,7968	0,2500	0,7500
208	-1,0000	0,8151	0,8831	0,8116	0,3125	0,7500
209	-1,0000	0,8027	0,9074	0,8069	0,3750	0,7500
210	-1,0000	0,8034	0,9641	0,7955	0,4375	0,7500
211	-1,0000	0,7933	0,9343	0,8024	0,5625	0,7500
212	-1,0000	0,8007	0,8936	0,8227	0,6250	0,7500
213	-1,0000	0,8010	0,8386	0,8191	0,6875	0,7500
214	-1,0000	0,0000	0,8272	0,8069	0,7500	0,7500
215	-1,0000	0,0000	0,8071	0,8155	0,8125	0,7500
216	-1,0000	0,0000	0,7994	0,7989	0,8750	0,7500
217	-1,0000	0,0000	0,7944	0,7925	0,9375	0,7500
218	-1,0000	0,0000	0,7944	0,7930	1,0000	0,7500
219	-1,0000	0,8077	0,8112	0,7961	0,0000	0,8125
220	-1,0000	0,8069	0,8050	0,7881	0,0625	0,8125
221	-1,0000	0,8057	0,8329	0,7904	0,1250	0,8125
222	-1,0000	0,8025	0,8149	0,7952	0,1875	0,8125
223	-1,0000	0,8106	0,8536	0,7901	0,2500	0,8125
224	-1,0000	0,8015	0,8673	0,8015	0,3125	0,8125
225	-1,0000	0,7949	0,8802	0,8109	0,3750	0,8125
226	-1,0000	0,7921	0,9373	0,8042	0,4375	0,8125
227	-1,0000	0,8128	0,9668	0,8176	0,5000	0,8125
228	-1,0000	0,8045	0,9081	0,8069	0,5625	0,8125
229	-1,0000	0,0000	0,8508	0,8185	0,6250	0,8125
230	-1,0000	0,0000	0,8542	0,8065	0,6875	0,8125
231	-1,0000	0,0000	0,8261	0,8029	0,7500	0,8125
232	-1,0000	0,0000	0,8315	0,7944	0,8125	0,8125
233	-1,0000	0,0000	0,8249	0,7960	0,8750	0,8125
234	-1,0000	0,0000	0,8040	0,7869	0,9375	0,8125
235	-1,0000	0,0000	0,8047	0,7866	1,0000	0,8125
236	-1,0000	0,8017	0,8188	0,0000	0,0000	0,8750
237	-1,0000	0,8092	0,8187	0,0000	0,0625	0,8750
238	-1,0000	0,8168	0,8337	0,0000	0,1250	0,8750
239	-1,0000	0,8057	0,8348	0,7832	0,1875	0,8750
240	-1,0000	0,8124	0,8345	0,7619	0,2500	0,8750
241	-1,0000	0,7983	0,8740	0,7926	0,3125	0,8750
242	-1,0000	0,7990	0,8791	0,7968	0,3750	0,8750
243	-1,0000	0,7998	0,9191	0,0000	0,4375	0,8750
244	-1,0000	0,8035	0,8963	0,8114	0,5000	0,8750
245	-1,0000	0,8042	0,8839	0,7983	0,5625	0,8750
246	-1,0000	0,0000	0,8720	0,7984	0,6250	0,8750

Tabela 11 - Dados para treinamento da rede PMC (conclusão)

Amostra	Sinal de Entrada para o Limiar de Ativação	Entradas			Saídas	
		Disp, 1	Disp, 2	Disp, 3	Coord, X	Coord, Y
247	-1,0000	0,0000	0,8215	0,7986	0,6875	0,8750
248	-1,0000	0,0000	0,8112	0,8067	0,7500	0,8750
249	-1,0000	0,0000	0,8192	0,7970	0,8125	0,8750
250	-1,0000	0,0000	0,8039	0,7968	0,8750	0,8750
251	-1,0000	0,0000	0,8065	0,7928	0,9375	0,8750
252	-1,0000	0,0000	0,7966	0,7961	1,0000	0,8750
253	-1,0000	0,7958	0,8008	0,0000	0,0000	0,9375
254	-1,0000	0,7978	0,8030	0,0000	0,0625	0,9375
255	-1,0000	0,7958	0,8267	0,0000	0,1250	0,9375
256	-1,0000	0,7990	0,8379	0,0000	0,1875	0,9375
257	-1,0000	0,7950	0,8500	0,7941	0,2500	0,9375
258	-1,0000	0,7963	0,8545	0,7926	0,3125	0,9375
259	-1,0000	0,7931	0,8821	0,7993	0,3750	0,9375
260	-1,0000	0,7958	0,8975	0,7940	0,4375	0,9375
261	-1,0000	0,7995	0,8859	0,7959	0,5000	0,9375
262	-1,0000	0,0000	0,8572	0,7985	0,5625	0,9375
263	-1,0000	0,0000	0,8822	0,7912	0,6250	0,9375
264	-1,0000	0,0000	0,8552	0,7966	0,6875	0,9375
265	-1,0000	0,0000	0,8300	0,7970	0,7500	0,9375
266	-1,0000	0,0000	0,8238	0,7907	0,8125	0,9375
267	-1,0000	0,0000	0,8171	0,7894	0,8750	0,9375
268	-1,0000	0,0000	0,8075	0,0000	0,9375	0,9375
269	-1,0000	0,0000	0,7886	0,0000	1,0000	0,9375
270	-1,0000	0,7975	0,8277	0,0000	0,0000	1,0000
271	-1,0000	0,7941	0,8359	0,7905	0,0625	1,0000
272	-1,0000	0,7995	0,8547	0,7876	0,1250	1,0000
273	-1,0000	0,7987	0,8502	0,7898	0,1875	1,0000
274	-1,0000	0,7906	0,8475	0,7892	0,2500	1,0000
275	-1,0000	0,7946	0,8612	0,7844	0,3125	1,0000
276	-1,0000	0,7975	0,8564	0,7925	0,3750	1,0000
277	-1,0000	0,0000	0,8800	0,7923	0,4375	1,0000
278	-1,0000	0,0000	0,8822	0,7893	0,5000	1,0000
279	-1,0000	0,0000	0,8560	0,7937	0,5625	1,0000
280	-1,0000	0,0000	0,8359	0,7921	0,6250	1,0000
281	-1,0000	0,0000	0,8446	0,7836	0,6875	1,0000
282	-1,0000	0,0000	0,8275	0,7894	0,7500	1,0000
283	-1,0000	0,0000	0,8017	0,7889	0,8125	1,0000
284	-1,0000	0,0000	0,8011	0,0000	0,8750	1,0000
285	-1,0000	0,0000	0,7948	0,0000	0,9375	1,0000
286	-1,0000	0,0000	0,7931	0,0000	1,0000	1,0000

APÊNDICE D — Tabela de dados normalizados para Operação da Rede Perceptron Multicamadas

Tabela 12 - Dados para teste e operação da rede PMC

Amostra	Sinal de Entrada para o Limiar de Excitação	Entradas			Saídas	
		Disp, 1	Disp, 2	Disp, 3	Coord, X	Coord, Y
1	-1,0000	0,8238	0,8193	0,8124	0,3900	0,4475
2	-1,0000	0,8391	0,8569	0,8163	0,3725	0,5575
3	-1,0000	0,8124	0,8946	0,8208	0,5225	0,6200
4	-1,0000	0,8141	0,8441	0,8535	0,5550	0,5425
5	-1,0000	0,8111	0,8428	0,8844	0,6075	0,5000
6	-1,0000	0,8042	0,8203	0,9012	0,6275	0,3925
7	-1,0000	0,8050	0,8074	0,8567	0,5225	0,4000
8	-1,0000	0,8436	0,7938	0,8490	0,4050	0,3350