

Monografia de Graduação

Implementação de um Preditor Neural no Ambiente Industrial Foundation Fieldbus

Alexandre Magno Pimentel Pinheiro Filho

Natal, fevereiro de 2010

Agradecimentos

Agradeço a Deus por me dar a oportunidade de fazer esse curso e nunca permitir que existissem problemas que me fizessem abandoná-lo.

Aos meus pais pelo carinho, apoio, educação, investimento e compreensão, sem os quais não seria possível terminar esse curso sem maiores dificuldades.

Aos professores Jorge e Adrião por esses dois anos de orientação no projeto de pesquisa, permitindo a criação de artigos, participação em congressos e o desenvolvimento deste trabalho.

Aos colegas do LAMP, Daniel, Leonardo, Bruno e Vinícius, por estarem sempre dispostos a interromper o que estavam fazendo para tirar alguma dúvida, bem como aos colegas de projeto Kennedy e Victor, por estarem sempre auxiliando nos trabalhos realizados.

Aos demais colegas do curso, por estarem sempre ajudando na hora dos estudos e dos trabalhos, assim como por sempre proporcionarem momentos de diversão nas horas vagas.

À ANP e ao PRH-14, pelo auxílio financeiro nesses dois últimos anos e por permitir que eu me mantivesse firme no projeto que deu origem a este trabalho.

Resumo

Este trabalho apresenta o estudo e implementação de algoritmos inteligentes para estimação de valores no processo de medição de gás natural, com a finalidade de detectar possíveis falhas nos sensores envolvidos nos processos de transferência de custódia do produto. Na criação destes algoritmos inteligentes, foram testadas algumas arquiteturas de redes neurais artificiais recorrentes, devido a algumas características, tais como aprendizado, adaptação e a capacidade de capturar e lidar com a dinâmica de um sistema. As redes neurais foram criadas para implementar um preditor com a finalidade de reproduzir o comportamento dinâmico da saída do sensor, o qual é comparado com o comportamento real do instrumento, formando assim a base para a detecção e classificação de falhas que possam ocorrer no dispositivo. Finalmente, essa rede recorrente foi embarcada em uma rede de dispositivos inteligentes, chamada rede industrial Foundation Fieldbus, obtendo assim algoritmos inteligentes implementados em campo. Serão apresentados os algoritmos de predição, o modo como as redes neurais foram embarcadas na rede Foundation Fieldbus e os resultados obtidos com os testes.

Abstract

This work presents the study and implementation of intelligent algorithms for estimation of values in the natural gas measurement process, with the intention of detect possible failures in the sensors involved in these product custody transfer process. In the creation of these intelligent algorithms, were tested some recurrent neural networks architectures, due to some characteristics, like learning, adaptation and the capacity of capturing and dealing with a system dynamics. The neural networks were created to implement a predictor with the objective of reproduce the output dynamic behaviour of the sensor, in such a way that the output of the network be compared to the output of the sensor, working as a basis for the detection and classification of failures that may occur in the device. Finnally, this recurrent network was embedded in network of intelligent devices, called Foundation Fieldbus industrial network, obtaining then intelligent algorithms implemented in Field. It will be presented the prediction algorithms, how the neural networks were embedded in the Foundation Fieldbus Network and the results obtained in the tests.

Sumário

Sumário	i
Lista de Figuras	iii
Lista de Tabelas	iv
Lista de Símbolos e Abreviaturas	v
1 Introdução	1
1.1 Motivação	1
1.2 Objetivos	1
1.3 Organização do Texto	2
2 Redes Neurais Artificiais	3
2.1 Introdução	3
2.2 Propriedades das redes neurais	3
2.3 Modelos de um Neurônio	4
2.4 Tipos de Função de Ativação	5
2.5 Arquiteturas de Redes Neurais	7
2.6 Processos de Aprendizagem	9
2.7 Redes neurais dinâmicas recorrentes	10
2.8 Arquiteturas de Redes Neurais Recorrentes	10
3 Redes Foundation Fieldbus	13
3.1 Introdução	13
3.2 Arquitetura do Sistema de Comunicação	14
4 Predição de Séries Temporais	17
4.1 Introdução	17
4.2 Classificação de Séries Temporais	17
4.3 Características das Séries Temporais	17
4.4 Predição de Séries Temporais Utilizando uma Rede Recorrente NARX	18
5 Redes Neurais na Rede Foundation Fieldbus	20
5.1 Construção de um neurônio artificial	20
5.2 Criação de bancos atrasadores	20
5.3 Esquema da rede NARX implementada na rede FF	21
6 Procedimentos Experimentais	23
6.1 Organização da Arquitetura e da Comunicação	23
6.2 Treinamento e Avaliação das Redes Neurais no Matlab	23
6.3 Embarcar a Rede Neural na Rede Industrial	30
7 Conclusão	33

7.1 Considerações Finais.....	33
7.2 Perspectivas Futuras.....	33
Referências Bibliográficas.....	34

Lista de Figuras

Figura 2.1 – Modelo de um neurônio artificial.....	4
Figura 2.2 – Função de limiar.....	5
Figura 2.3 – Função linear por partes.....	6
Figura 2.4 – Função sigmóide com padrão de inclinação variável.....	6
Figura 2.5 – Rede neural não realimentada com uma única camada.....	8
Figura 2.6 – Rede totalmente conectada com arquitetura 4-4-2.....	9
Figura 2.7 – Rede recorrente sem neurônios ocultos.....	9
Figura 2.8 – Arquitetura de uma rede de Elman.....	11
Figura 2.9 – Rede recorrente de Jordan.....	12
Figura 2.10 – Rede Recorrente NARX.....	12
Figura 3.1 – Demonstração da redução no equipamento.....	13
Figura 3.2 – Codificação Manchester Bifase-L.....	14
Figura 3.3 – Preâmbulo e delimitador de início e fim de quadro.....	15
Figura 4.1 – rede NARX no modo paralelo.....	19
Figura 4.2 – rede NARX no modo série-paralelo.....	19
Figura 5.1 – Esquema de um neurônio na rede FF.....	20
Figura 5.2 – Exemplo de teste para ilustrar o comportamento do atraso.....	21
Figura 5.3 – Esquema da rede neural implementada na rede FF.....	22
Figura 6.1 – Predição de 1 passo.....	24
Figura 6.2 – predição de 3 passos.....	25
Figura 6.3 – predição de 5 passos.....	25
Figura 6.4 – Preditor de 1 passo para uma série com a Falha 1.....	26
Figura 6.5 – Preditor de 2 passos para uma série com a Falha 1.....	27
Figura 6.6 – Preditor de 1 passo para uma série com a Falha 2.....	27
Figura 6.7 – Preditor de 2 passos para uma série com a Falha 2.....	28
Figura 6.8 – Preditor de 1 passo para uma série com a Falha 3.....	28
Figura 6.9 – Preditor de 2 passos para uma série com a Falha 3.....	29
Figura 6.10 – Preditor de 1 passo para uma série com a Falha 4.....	29
Figura 6.11 – Preditor de 2 passos para uma série com a Falha 4.....	30
Figura 6.12 – Predição de 1 passo de uma série sem falhas, na rede FF.....	31
Figura 6.13 – Predição de 1 passo de uma série com falhas dos tipos 4 e 1, na rede FF..	31
Figura 6.14 – Predição de 1 passo de uma série com falhas dos tipos 3 e 1, na rede FF..	32

Lista de Tabelas

Tabela 6.1 – Correlações.....	24
-------------------------------	----

Lista de Símbolos e Abreviaturas

AI	Analog Input
EKF	Filtro de Kalman Estendido (Extended Kalman Filter)
FF	Foundation Fieldbus
ISEL	Input Selector
LAS	Link Active Scheduler
NARX	Nonlinear AutoRegressive model with eXogenous input
OPC	OLE for Process Control
OSI	Open Systems Interconnection
SCHAR	Signal Characterizer

Capítulo 1

Introdução

1.1 Motivação

Gás natural é a denominação utilizada para designar os hidrocarbonetos leves encontrados armazenados em formações porosas no subsolo que podem ser encontrados associados ao petróleo. [Rodrigues, 2004]

Desde a etapa de produção até a sua distribuição, o gás natural é constantemente monitorado através de instrumentos de medição, que são utilizados para medir diversas variáveis, tais como vazão, temperatura e pressão.[de Medeiros, 2009]

Uma das formas de distribuição do gás natural é através de gasodutos, que são tubulações com a função de conduzir o gás sob pressão, através de compressores. Como o atrito causa uma perda de energia e queda gradativa da pressão ao longo da tubulação, são necessárias estações de compressão ao longo do percurso para aumentar a pressão e permitir a continuidade do fluxo do produto.

Durante a sua distribuição, o gás natural passa a ser responsabilidade de várias empresas diferentes, já que há a necessidade de transportar o gás natural do produtor ao cliente final. Logo, existe o repasse do produto entre as empresas, por meio de um processo chamado transferência de custódia. Essa transferência de custódia é realizada em entroncamentos de gasodutos ou em citygates, que possuem estações de controle e medição específicas.

Em cada estação, mede-se o volume de gás natural, que é medido através de sua vazão e calculado nas condições de base de pressão e temperatura, 1 atm e 20°C respectivamente.

Os instrumentos de medição são chamados de sensores ou medidores. Um sensor possui a função de transdutor, recebendo como entrada uma grandeza física, como por exemplo: vazão, pressão e temperatura, e disponibiliza como saída um sinal elétrico, com valor proporcional ao da grandeza medida.

É importante que as medições feitas por cada instrumento sejam precisas, já que o custo do gás natural recebido estará de acordo com o volume fornecido. A existência de erros de medição gera cálculos de preços incompatíveis com o volume real do produto, acarretando em perdas econômicas. Como o gás natural é transportado em grandes quantidades, pequenos erros de medição podem levar a prejuízos elevados.

Os erros de medição nos sensores podem ter origem no próprio instrumento, quando ocorre descalibração ou falhas mecânicas e elétricas, ou por influências externas, como quando ocorre o ruído. O ideal é que as medidas dos sensores sejam corrigidas quando o erro ultrapassar um limite aceitável.

As técnicas de detecção de falhas mais convencionais são baseadas em redundância física, que pode gerar um aumento significativo de custos, ou na modelagem matemática do processo em questão, que requer um profundo conhecimento do processo que se está analisando.

1.2 Objetivos

Esse trabalho tem como objetivo realizar a predição do valor de saída de um sensor de temperatura, gerando informação suficiente para que seja possível a detecção e classificação de falhas, sem que seja necessário um conhecimento matemático avançado sobre o comportamento do sistema, nem seja necessária a presença de

redundância física.

Essa predição será realizada por meio da utilização de redes neurais recorrentes, que permitem realizar a predição utilizando apenas valores da série temporal que se deseja simular, sem a necessidade de equações que descrevam o problema.

Essas redes neurais serão distribuídas e embarcadas em uma rede de sensores e atuadores com capacidade de processamento chamada rede Foundation Fieldbus, que pode ser instalada em campo e diminui a necessidade de um elemento supervisor, podendo tomar decisões de forma local, com base nos valores medidos.

1.3 Organização do Texto

Neste documento, o Capítulo 2 apresenta uma fundamentação teórica relacionada a redes neurais com ênfase nas arquiteturas e configurações utilizadas neste trabalho. O Capítulo 3 descreve um pouco do funcionamento e também apresenta uma fundamentação teórica sobre as redes industriais Foundation Fieldbus. No Capítulo 4, são apresentadas algumas informações sobre séries temporais e sobre a técnica de predição utilizada neste trabalho. Já no Capítulo 5, é explicada uma solução sobre como embarcar redes neurais nos sensores da rede industrial Foundation Fieldbus, além de ilustrar um esquema da rede recorrente embarcada neste trabalho. No Capítulo 6 são apresentados os testes realizados e os resultados obtidos são exibidos. Por fim, o Capítulo 7 mostra as conclusões e sugestões para trabalhos futuros com base neste trabalho.

Capítulo 2

Redes Neurais Artificiais

2.1 Introdução

Uma rede neural artificial, ou simplesmente rede neural, é um modelo matemático que visa simular o comportamento do cérebro humano, sendo o seu estudo impulsionado pelo fato de que o cérebro trabalha de forma diferente de um computador digital. O cérebro pode ser visto como um computador extremamente complexo, não-linear e paralelo, tendo a capacidade de organizar os seus neurônios de modo a realizar certas tarefas de maneira muito mais rápida que os melhores computadores atuais, tais como reconhecimento de padrões, percepção e controle motor [Silva, 2005].

Em sua forma mais geral, uma rede neural é um processador paralelo formado por unidades de processamento simples, chamados neurônios, que têm a capacidade de armazenar conhecimento e torná-lo disponível para o uso. Ela se parece com o cérebro nos seguintes aspectos:

- Seu conhecimento é adquirido do ambiente através de um processo de aprendizagem.
- As forças das conexões entre os neurônios, chamadas de pesos sinápticos, são utilizadas para armazenar o conhecimento adquirido.

O processo utilizado no procedimento de aprendizagem da rede é chamado de algoritmo de aprendizagem, que modifica os pesos sinápticos da rede de modo a alcançar um objetivo desejado.

A modificação dos pesos sinápticos gera o método tradicional para o projeto de redes neurais, entretanto, a rede pode ter capacidade de modificar a sua própria estrutura, baseando-se no fato de que neurônios no cérebro humano podem morrer e com isso novas conexões sinápticas podem surgir [Haykin, 2005].

2.2 Propriedades das redes neurais

As redes neurais possuem duas propriedades importantes: primeiro, sua estrutura distribuída paralelamente, e segundo, a sua capacidade de aprender e generalizar. A generalização significa que a rede pode gerar uma saída apropriada para uma entrada que não tenha sido apresentada no processo de aprendizagem. Essas duas características no processamento de informações fazem com que as redes neurais sejam capazes de solucionar problemas complexos de forma vantajosa, quando comparadas com abordagens tradicionais que são atualmente intratáveis. Outras propriedades das redes neurais estão listadas a seguir [Silva, 2005].

1 – Não linearidade. Uma rede neural formada pela interconexão de neurônios não-lineares permite gerar uma relação não-linear entre entradas e saídas.

2 – Mapeamento de entrada e saída. O paradigma de aprendizagem chamado aprendizagem supervisionada envolve a modificação dos pesos sinápticos da rede neural através da apresentação de um conjunto de treinamento formado por entradas e suas respectivas saídas desejadas. O processo de aprendizagem consiste em apresentar um exemplo aleatório do conjunto de treinamento e ajustar os pesos sinápticos de modo a diminuir a diferença entre a saída da rede e a saída desejada do exemplo. Esse processo é repetido várias vezes, até que não exista nenhuma mudança significativa nos valores dos pesos sinápticos. Dessa maneira, a rede gera um mapeamento de entrada-saída a partir dos exemplos apresentados.

3 – Adaptabilidade. Redes neurais têm a capacidade de adaptar os seus pesos sinápticos de acordo com as mudanças no ambiente. Em particular, uma rede treinada para atuar em um dado ambiente, pode ser facilmente retreinada para lidar com pequenas mudanças nas condições do ambiente. Além disso, uma rede neural atuante em um ambiente não estacionário pode ser projetada de modo a modificar os seus pesos sinápticos em tempo real.

4 – Tolerância a falhas. Uma rede neural implementada em *hardware* tem o potencial de ser tolerante a falhas, no sentido de que a sua performance diminui gradativamente sob condições adversas. Por exemplo, se um neurônio ou suas conexões forem danificadas, a recuperação de um padrão é prejudicada em qualidade. Porém, como a rede neural apresenta uma arquitetura onde as informações são distribuídas ao longo da mesma, o dano deve ser extenso antes que a resposta da rede como um todo seja degradada seriamente. Logo, observa-se uma degradação suave do desempenho da rede, ao invés de uma falha seriamente comprometedora [Silva, 2005].

2.3 Modelos de um Neurônio

Um neurônio é uma unidade de processamento de informação fundamental para a operação de redes neurais. O diagrama de blocos da figura 2.1 representa o modelo de um neurônio, que forma a base do projeto de uma rede neural artificial. A seguir serão identificados três elementos básicos do modelo de um neurônio artificial [Silva, 2005].

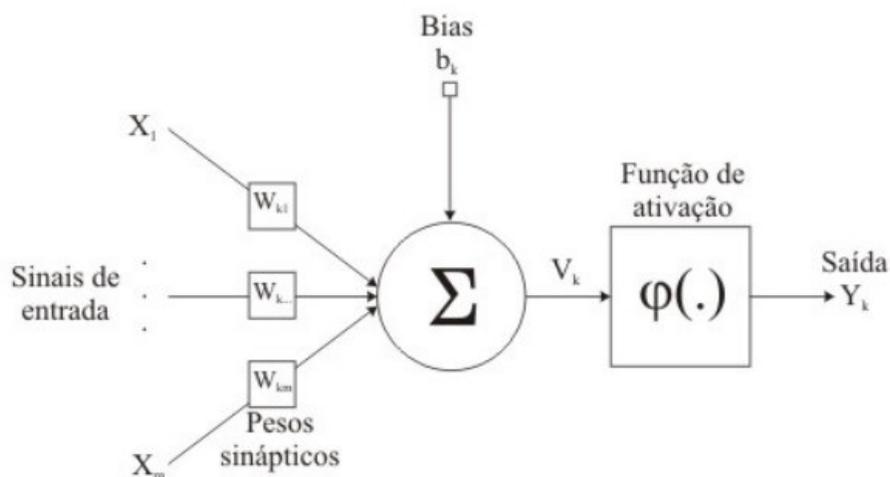


Figura 2.1 – Modelo de um neurônio artificial

1 – Uma série de sinapses. Cada sinapse é caracterizada por um peso ou força de conexão. Especificamente, um sinal x_j na entrada da sinapse j conectada ao neurônio k é multiplicado por um peso w_{kj} . Diferentemente da sinapse de um cérebro, os pesos sinápticos de um neurônio podem variar em uma faixa que pode incluir valores tanto negativos como positivos.

2 – Um combinador linear. Esse combinador serve para somar os sinais de entrada ponderados pelos pesos sinápticos do neurônio.

3 – Uma função de ativação. A função de ativação serve para associar o sinal resultante do combinador linear a um valor de saída, podendo aplicar não-linearidade ou restrição. A função de ativação também tem a tarefa de limitar a amplitude da saída de um neurônio a um valor finito. Geralmente essa limitação fica entre $[0,1]$ ou $[-1,1]$ devido a operações de normalização.

O modelo neuronal da figura 2.1 ainda inclui um bias aplicado externamente, denominado b_k , tendo o efeito de aumentar ou diminuir a entrada na função de ativação, dependendo se o mesmo é positivo ou negativo.

Matematicamente, pode-se descrever um neurônio k através das seguintes equações [Silva, 2005]:

$$u_k = \sum_{j=1}^m w_{kj} x_j$$

$$y_k = \varphi(u_k + b_k)$$

Onde:

- x_1, x_2, \dots, x_m são os valores de entrada;
- $w_{k1}, w_{k2}, \dots, w_{km}$ são os pesos sinápticos do neurônio k ;
- u_k é a saída do combinador linear;
- b_k é o bias;
- $\varphi(\cdot)$ é a função de ativação;
- y_k é o sinal de saída do neurônio.

2.4 Tipos de Função de Ativação

A função de ativação, denotada por $\varphi(v)$, define a saída de um neurônio com base no campo local induzido v . Aqui é possível identificar três tipos básicos de função de ativação [Silva, 2005]:

1 – Função de limiar. Para esse tipo de função de ativação, descrito na figura 2.2, tem-se:

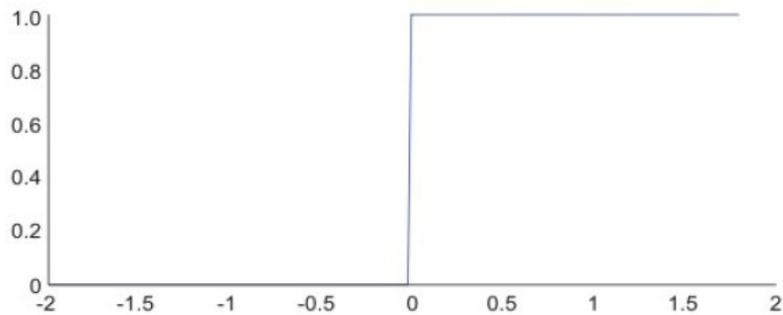


Figura 2.2 – Função de limiar

Correspondentemente, a saída do neurônio k empregando essa função pode ser definida como:

$$\varphi(v_k) = \begin{cases} 1 & \text{se } v_k \geq 0 \\ 0 & \text{se } v_k < 0 \end{cases}$$

Onde v_k é o campo local induzido do neurônio. Esse neurônio é referido na literatura como o modelo de McCulloch-Pitts. Nesse modelo, a saída do neurônio recebe o valor de 1 caso o campo local induzido seja não-negativo, e 0 caso contrário.

2 – Função Linear por Partes. Para a função linear por partes, descrita na figura

2.3, temos:

$$\varphi(v_k) = \begin{cases} 1 & \text{se } v_k \geq \frac{1}{2} \\ v_k & \text{se } \frac{1}{2} > v_k \geq -\frac{1}{2} \\ 0 & \text{se } v_k \leq -\frac{1}{2} \end{cases}$$

Onde é assumido que o fator de operação dentro da região linear é a unidade. Essa forma de função de ativação pode ser vista como uma aproximação de um amplificador não-linear.

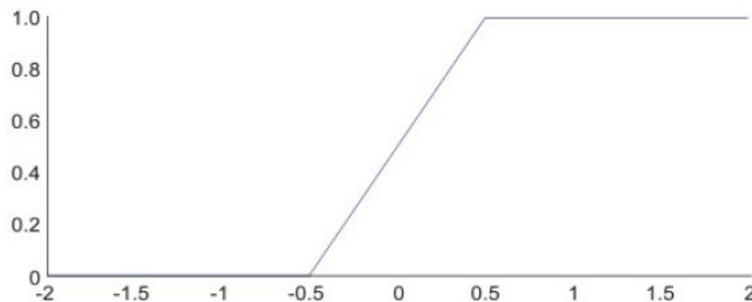


Figura 2.3 – Função linear por partes

As duas situações seguintes podem ser vistas como formas especiais de uma função linear por partes:

- A função se torna um combinador linear se a região linear de operação for mantida dentro da zona de saturação.
- A função linear por partes é reduzida à função de limiar se o fator de amplificação da região é feito infinitamente grande.

3 – Função Sigmóide. A função sigmóide, cujo gráfico tem forma de s, é a função de ativação mais utilizada na construção de redes neurais artificiais. Ela é definida como uma função estritamente crescente que exibe um balanceamento adequado entre o comportamento linear e não-linear. Um exemplo de função sigmóide é a função logística, definida por:

$$\varphi(v_k) = \frac{1}{1 + e^{-av_k}}$$

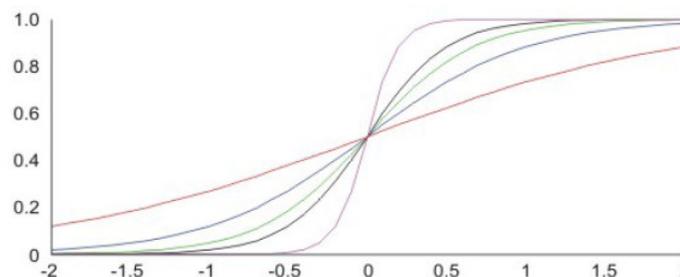


Figura 2.4 – Função sigmóide com padrão de inclinação variável

Onde a é o parâmetro de inclinação da função sigmóide. Variando o parâmetro a , é possível obter funções sigmóides com diferentes inclinações, como ilustrado na figura 2.4. No limite, quando o parâmetro de inclinação tende ao infinito, a função sigmóide se torna uma simples função de limiar. Nota-se também que a função sigmóide é infinitamente diferenciável, enquanto que a função de limiar não é.

As funções de ativação definidas acima variam de 0 a 1, mas algumas vezes é desejável que se tenha uma função de ativação que varie de -1 a 1, onde a função de ativação assume uma forma anti-simétrica em relação à origem, isto é, a função de ativação é uma função ímpar do campo local induzido. Especificamente, a função limiar agora é definida como:

$$\varphi(v_k) = \begin{cases} 1 & \text{se } v_k \geq 0 \\ 0 & \text{se } v_k = 0 \\ -1 & \text{se } v_k < 0 \end{cases}$$

Que é comumente chamada de função sinal. Como uma forma correspondente da função sigmóide, pode-se utilizar a função tangente hiperbólica, definida por:

$$\varphi(v_k) = \frac{e^{v_k} - e^{-v_k}}{e^{v_k} + e^{-v_k}}$$

Permitindo assim que uma função de ativação do tipo sigmóide assumira valores negativos.

2.5 Arquiteturas de Redes Neurais

Em geral, é possível identificar três classes fundamentais de arquiteturas de redes neurais [Silva, 2005].

1 – Redes Alimentadas Adiante com Camada Única. Na forma mais simples de uma rede organizada em camadas, tem-se uma camada de entrada de dados, seguida de uma única camada de processamento, que calcula e fornece as saídas. Na figura 2.5 está ilustrada uma rede neural com camada única para o caso de quatro nós na camada de entrada. Esta rede é chamada rede de camada única, referindo-se apenas à camada de processamento. A nomenclatura não considera a camada de entrada, pois ela não realiza nenhuma computação.

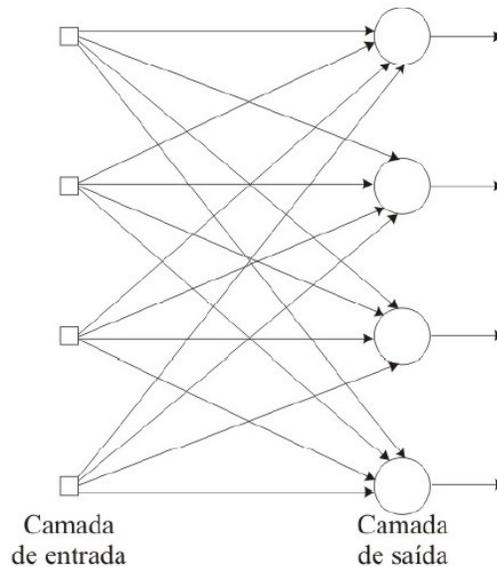


Figura 2.5 – Rede neural não realimentada com uma única camada

2 – Redes Alimentadas Diretamente com Múltiplas Camadas. Essa classe de rede neural, também conhecida como perceptron de múltiplas camadas, caracteriza-se pela presença de uma ou mais camadas ocultas, cujos nós computacionais correspondentes são chamados de neurônios ocultos. A função dos neurônios ocultos é intervir entre a entrada externa e a saída da rede neural de alguma maneira útil. Adicionar uma ou mais camadas ocultas habilita a rede a extrair estatísticas de ordem elevada, desse modo, a rede adquire uma perspectiva global, apesar de sua conectividade local, devido ao conjunto extra de conexões sinápticas e à dimensão extra de interações neurais. A capacidade dos neurônios ocultos de extrair estatísticas de ordem elevada é particularmente valiosa quando o tamanho da camada de entrada é grande.

Nessa arquitetura, os nós da camada de entrada fornecem os sinais de entrada aos nós da primeira camada oculta. Esta, por sua vez, fornece os seus sinais de saída como entrada para a próxima camada, e assim sucessivamente por toda a rede. O conjunto de saídas dos neurônios da última camada forma a resposta global da rede ao padrão de ativação suprido pelos nós na entrada da rede. A figura 2.6 ilustra o esquema de uma rede neural com uma camada oculta. A arquitetura dessa rede pode ser definida como 4-4-2, já que a mesma apresenta 4 neurônios na camada de entrada, 4 neurônios na camada oculta e 2 neurônios na camada de saída.

A rede neural na figura 2.6 é classificada como totalmente conectada, no sentido de que cada nó de cada camada da rede está conectado com cada nó da camada seguinte. De outro modo, se algumas conexões sinápticas estiverem faltando, a rede é classificada como parcialmente conectada.

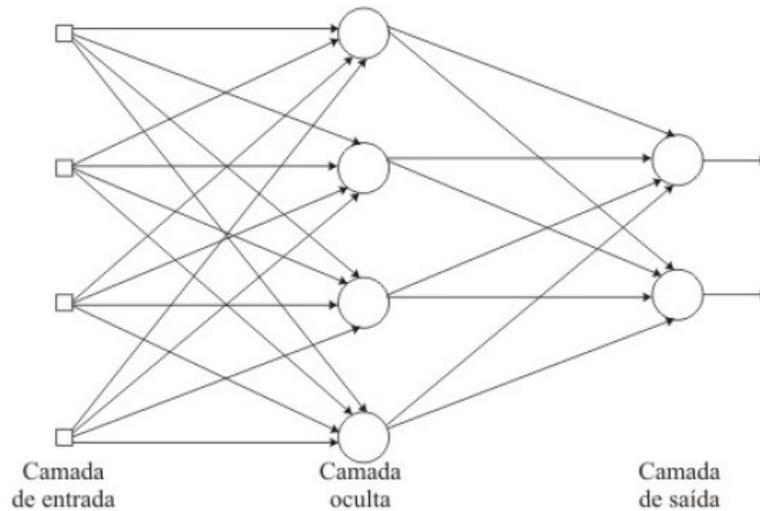


Figura 2.6 – Rede totalmente conectada com arquitetura 4-4-2

3 - Redes Recorrentes. Uma rede neural recorrente diferencia-se da rede mostrada anteriormente no sentido de que ela tem pelo menos um loop de realimentação. Por exemplo, uma rede recorrente pode consistir de uma única camada de neurônios em que cada neurônio alimenta os sinais de entrada de todos os outros neurônios com o seu sinal de saída, como ilustrado na figura 2.7.

A presença de loops de realimentação tem um impacto profundo tanto na capacidade de aprendizagem da rede neural como na sua performance. Além disso, os loops de realimentação envolvem o uso de ramos particulares compostos de elementos de atraso unitário, que resultam em um comportamento de dinâmica não-linear, assumindo que a rede neural contenha unidades não lineares [Silva, 2005].

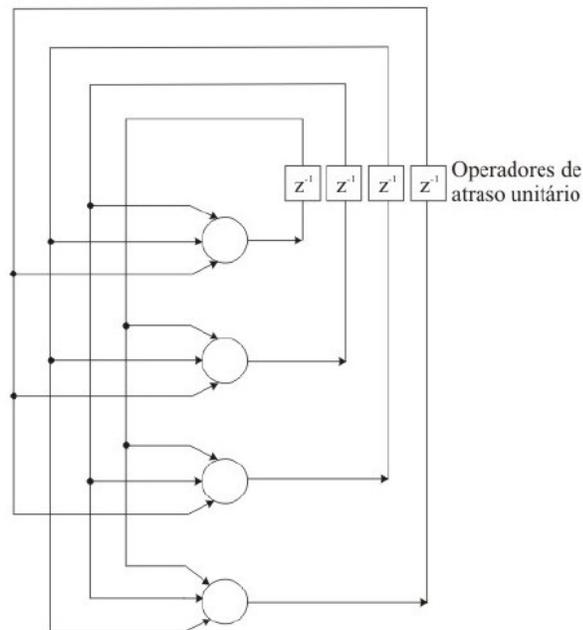


Figura 2.7 – Rede recorrente sem neurônios ocultos

2.6 Processos de Aprendizagem

Com relação ao processo de aprendizagem, as redes neurais se dividem em dois tipos: as supervisionadas e as não supervisionadas. As redes supervisionadas contam com a presença de um professor, que tem conhecimento sobre o ambiente. O conhecimento se encontra na forma de conjuntos de exemplos de entrada-saída e é através deles que a rede consegue aprender. O processo de aprendizagem é realizado através da alteração dos pesos sinápticos, diminuindo assim o erro na saída [Silva, 2005].

Na aprendizagem não supervisionada, não existe um professor para auxiliar o processo, o que significa que não há exemplos rotulados do ambiente a serem aprendidos pela rede.

Dentre os vários algoritmos de treinamento supervisionado, o algoritmo *backpropagation*, ou retropropagação do erro, é o mais utilizado para treinar redes neurais de múltiplas camadas. O procedimento da retropropagação do erro envolve duas fases: a primeira ocorre quando as entradas são apresentadas à rede e propagadas adiante até que seja computado um valor de saída, esse valor é comparado com a saída desejada e então é calculada a diferença entre esses dois valores, ou seja, o erro. Na segunda fase, o erro é propagado na direção contrária da entrada, ou seja, para trás. Esta fase tem como objetivo a alteração dos pesos sinápticos, baseando-se no erro retropropagado. Apenas quando essas duas fases são completadas, podem ser apresentados novos conjuntos de entrada e saída.

2.7 Redes neurais dinâmicas recorrentes

Uma rede neural dinâmica recorrente, ou simplesmente rede recorrente, é uma rede neural com um ou mais laços de realimentação. A realimentação pode ser feita entre neurônios de camadas diferentes, entre neurônios da mesma camada ou de um neurônio para ele mesmo [Júnior, 2006].

Existem dois tipos principais de conexões: conexão de alimentação direta e conexão de realimentação. A conexão de alimentação direta ocorre quando um sinal tem orientação no sentido da entrada para a saída, já a conexão de realimentação ocorre quando o sinal tem sentido contrário, ou seja, da saída para a entrada. Dessa forma, as conexões realimentam uma dada camada com sinais de ativação ou de saída.

Um modo de classificar as redes recorrentes é verificar a extensão das conexões de realimentação existentes, ou seja, se a mesma envolve neurônios de camadas diferentes ou apenas neurônios da mesma camada. Dessa maneira, é possível classificar as redes neurais recorrentes em três grupos:

1 – Conexão Recorrente Local. Neste tipo de conexão, a saída de um neurônio funciona como realimentação apenas para ele mesmo.

2 – Conexão Recorrente Global. Este tipo de conexão acontece entre um neurônio e outro de uma camada anterior, ou seja, o sinal de saída de um neurônio é realimentado para a entrada de outro neurônio em uma camada anterior.

3 – Conexão Recorrente Não-Local. Este é um tipo particular de conexão recorrente global, em que a saída do neurônio está conectada à entrada de outro neurônio localizado na mesma camada.

2.8 Arquiteturas de Redes Neurais Recorrentes

Como visto no tópico anterior, existem diferentes tipos de conexões entre

neurônios em uma rede neural recorrente, fato esse que gera a possibilidade de implementação de várias arquiteturas diferentes para redes neurais recorrentes [Júnior, 2006].

1 – Rede Recorrente de Elman. Esta é uma arquitetura obtida a partir da rede perceptron de múltiplas camadas, através da redefinição da camada de entrada, que passa a ser dividida em duas partes. A primeira corresponde ao vetor de entrada propriamente dito, já a segunda parte, chamada de unidade de contexto, consiste na cópia das saídas dos neurônios da camada oculta no instante anterior. A figura 2.8 ilustra uma rede recorrente de Elman com uma camada oculta.

Se uma rede perceptron de múltiplas camadas tem duas camadas ocultas, é possível escolher de qual camada oculta será feita a realimentação, podendo inclusive serem utilizadas ambas as camadas.

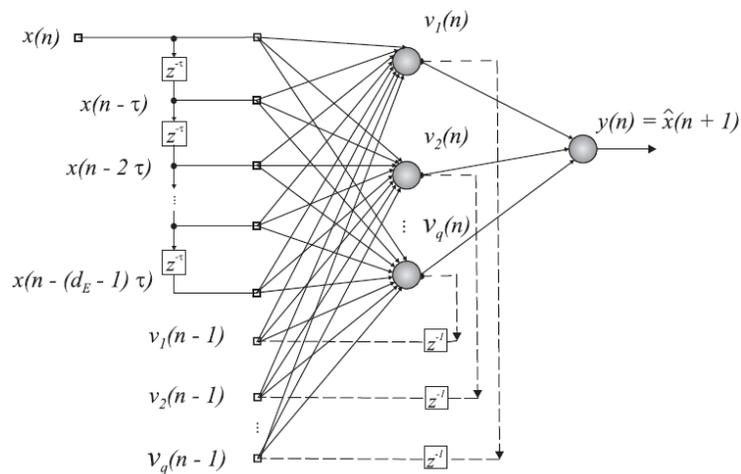


Figura 2.8 – Arquitetura de uma rede de Elman

2 – Rede Recorrente de Jordan. Assim como a rede de Elman, a rede de Jordan também não possui recorrência entre neurônios da mesma camada, sendo por isso classificada como uma rede globalmente recorrente.

Diferentemente da rede de Elman, que utiliza as conexões das camadas ocultas para fazer a realimentação, a rede de Jordan utiliza os sinais de saída da rede neural para realizar essa tarefa. Além disso, essa rede recorrente possui auto-realimentações, nas quais a saída de uma unidade de contexto é realimentada para a sua entrada. A figura 2.9 ilustra uma rede recorrente de Jordan com uma camada escondida.

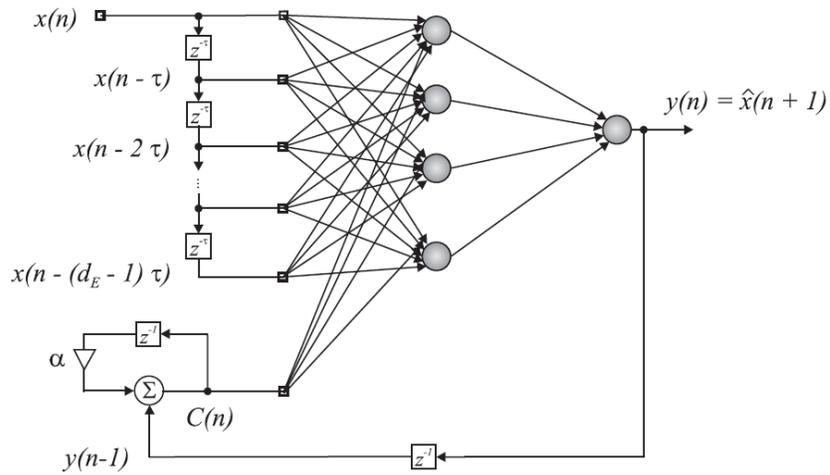


Figura 2.9 – Rede recorrente de Jordan

3 – Rede Dinâmica NARX. Uma importante classe de sistemas não-lineares de tempo discreto, é matematicamente representada pelo modelo NARX (Nonlinear AutoRegressive model with eXogenous inputs), dado por:

$$y(n) = f[y(n-1), \dots, y(n-d_y), u(n), u(n-1), \dots, u(n-d_u+1)]$$

Em que a função $f(\cdot)$ é uma função não-linear, geralmente desconhecida. Quando esta função é aproximada por uma rede perceptron de múltiplas camadas, a topologia resultante é chamada de rede recorrente NARX. A figura 2.10 ilustra uma rede NARX com uma camada oculta. Deve-se notar que a rede NARX possui atrasadores na camada de entrada e um laço de realimentação global.

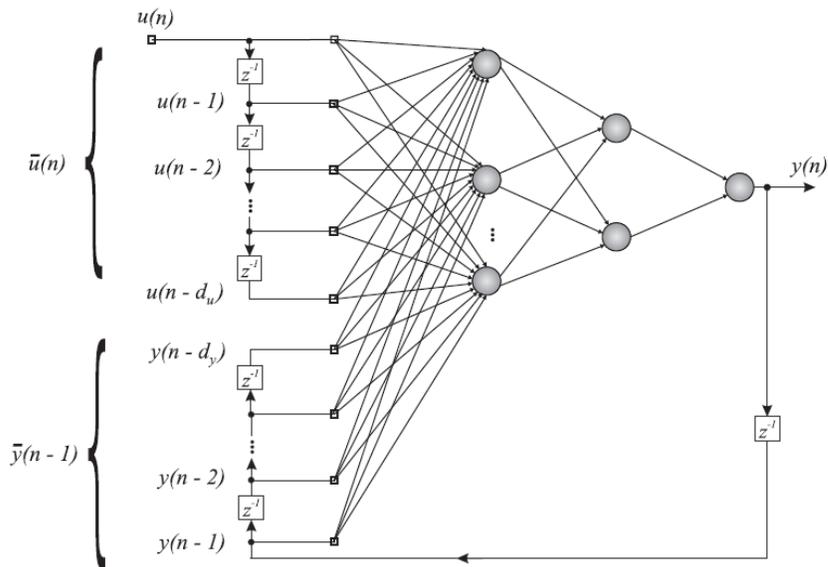


Figura 2.10 – Rede Recorrente NARX

Capítulo 3

Redes Foundation Fieldbus

3.1 Introdução

A Fieldbus Foundation é uma organização sem fins lucrativos, cujo principal propósito é desenvolver e manter um padrão internacional para redes de dispositivos de campo utilizados na automação de processos: o Foundation Fieldbus. O Foundation Fieldbus é um padrão aberto, capaz de possibilitar que dispositivos de diversos fabricantes sejam integrados a um mesmo sistema, caso os mesmos sigam a especificação [Silva, 2005].

O Foundation Fieldbus é um sistema de comunicação digital, serial e bidirecional, que tem a finalidade de interligar instrumentos em campo, tais como sensores e atuadores, utilizando a capacidade de processamento dos mesmos e permitindo a distribuição do controle do processo através da rede. A rede Foundation Fieldbus também pode ser ligada a redes TCP/IP/Ethernet, para que a configuração dos dispositivos possa ser feita remotamente.

Nessa tecnologia, as estratégias de controle não são centralizadas, como acontece em outras redes industriais que fazem uso de circuitos lógicos programáveis, e passam a ser distribuídas entre os dispositivos de campo. Isso se torna possível graças à utilização de blocos funcionais para a configuração dos instrumentos. Outra vantagem é a existência de determinismo nas comunicações entre os dispositivos, o que permite sistemas de controle mais robustos e seguros, sem que haja necessidade do usuário se preocupar com atrasos nas comunicações dos instrumentos e com falhas em malhas de controle geradas por eles.

As redes Foundation Fieldbus também geram a diminuição da quantidade de instrumentos em campo, como fontes de alimentação, controladores lógicos programáveis, e de cabeamento, já que tanto a alimentação como a comunicação é feita pelo mesmo par de fios de maneira serial e digital. A figura 3.1 ilustra um exemplo de diminuição no equipamento [Martins, 2008].

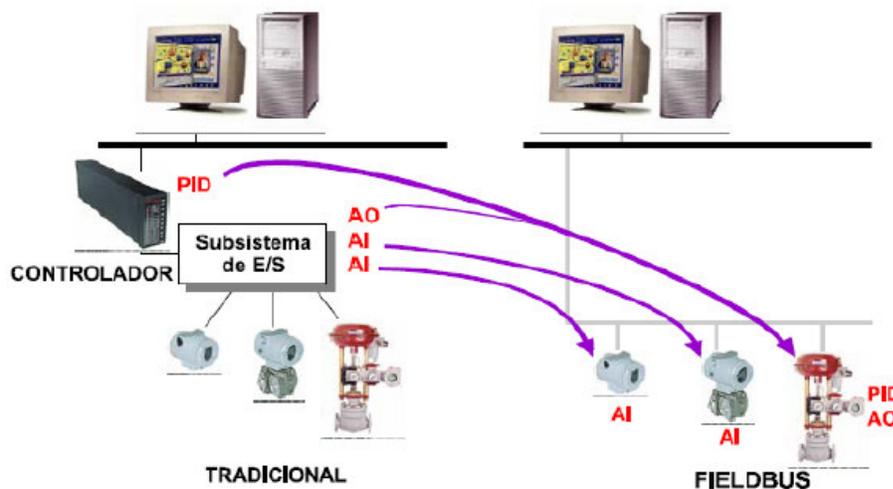


Figura 3.1 – Demonstração da redução no equipamento

Outra característica das redes Foundation Fieldbus é a capacidade de realização

de autotestes para identificar problemas de comunicação ou nos equipamentos sensores e atuadores, gerando alertas para manutenção preventiva ou corretiva.

3.2 Arquitetura do Sistema de Comunicação

A arquitetura do sistema de comunicação entre os instrumentos das redes Foundation Fieldbus (FF) é baseada no modelo OSI da ISO, sendo formado por três camadas: a camada física, a pilha de comunicação e a camada de usuário. Fazendo a relação com o modelo OSI, tem-se que a camada física nas redes FF é equivalente à camada física do modelo OSI. A pilha de comunicação contém a camada de acesso ao meio e a camada de aplicação. A camada de usuário na rede FF é inexistente no modelo OSI [Filho].

As redes FF têm a sua camada física baseada na norma IEC 61158-2. Os sinais da rede são codificados utilizando a codificação Manchester Bifase-L, que é uma comunicação síncrona que envia os sinais de dados combinados com o relógio.

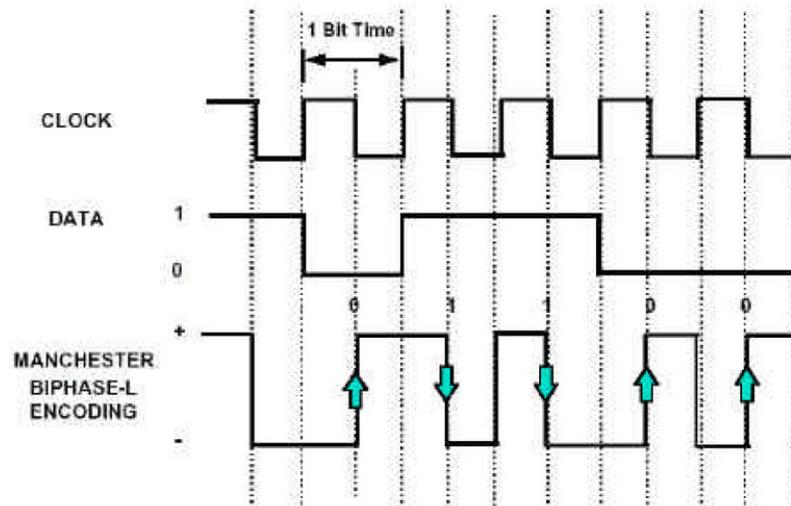


Figura 3.2 – Codificação Manchester Bifase-L

Pode-se ver que o sinal resultante corresponde à função XOR negada dos sinais de clock e dados, em que uma transição positiva do sinal combinado no meio do período correspondente a um bit será considerada 0 e uma transição negativa é lida como 1. Para fazer a sincronização do clock do transmissor com o clock do receptor, utiliza-se um preâmbulo formado por uma sequência de 0s e 1s. Para delimitar os quadros, utiliza-se os sinais N+ e N-, que não apresentam uma transição no meio do tempo correspondente a um bit.

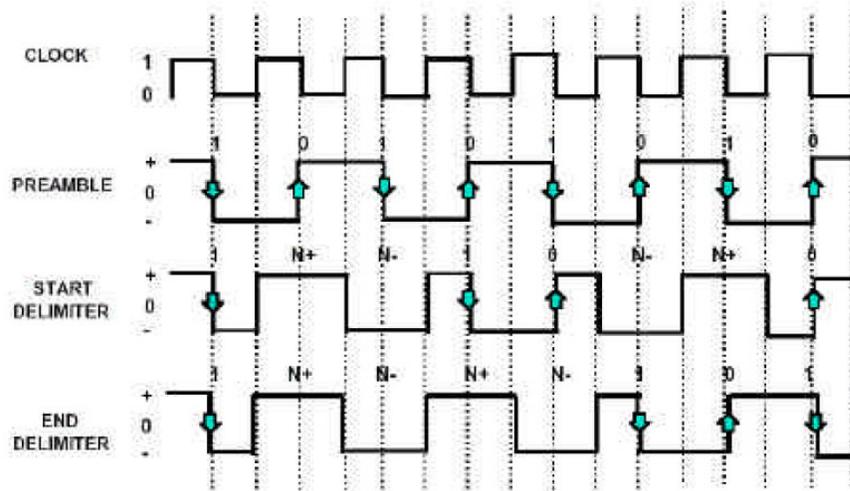


Figura 3.3 – Preâmbulo e delimitador de início e fim de quadro

O nível 2 da pilha de comunicação, a camada de link de dados, é o responsável pelo gerenciamento da comunicação da rede. Esta função é realizada através do agendador de link ativo (link active scheduler, ou LAS), que tem a tarefa de ser o árbitro do barramento, possuindo todos os recursos necessários para gerenciar as comunicações na rede. Uma mesma rede pode ter vários árbitros de barramento, havendo assim um sistema de redundância, desse modo, caso um árbitro tenha algum problema ou seja retirado da rede, um outro agendador de link ativo pode tomar o seu lugar e dar continuidade ao funcionamento da rede.

Existem dois tipos de comunicação nas redes industriais FF, a síncrona ou agendada e a assíncrona ou não agendada. No início do funcionamento da rede, o LAS monta uma lista com todos os instrumentos pertencentes à mesma e o tempo em que esses instrumentos devem enviar seus dados, essa operação garante o determinismo da rede, permitindo que o usuário saiba quando um dado será atualizado na malha de controle.

Já a comunicação assíncrona pode ocorrer entre as transmissões agendadas, no momento em que geralmente algum dispositivo deseja enviar uma mensagem no tempo de processamento de outro instrumento. Nesse tempo, o LAS permite que esse dispositivo utilize o barramento para enviar dados até que o tempo máximo reservado para o mesmo termine.

A camada de aplicação utiliza o conceito de blocos para realizar todas as suas tarefas. Existem três tipos de blocos: blocos de recursos, blocos de transdutores e blocos de função, onde os blocos de recursos têm a função de descrever as características gerais do dispositivo, os blocos transdutores são responsáveis pela interface com os sensores e atuadores de campo, bem como receber informações de calibração e configuração dos instrumentos. Já os blocos de função determinam o comportamento do sistema. As entradas e saídas de diferentes blocos de função podem ser interligadas para configurar uma estratégia de controle.

Os principais blocos funcionais utilizados neste e em outros trabalhos foram os seguintes [Martins, 2008]:

- Bloco Aritmético – Esse bloco tem a função de realizar cálculos sobre os sinais de entrada, podendo retornar resultados como a média, a soma ou a divisão entre os sinais de entrada, entre outras funções. Cada valor de entrada tem associada à mesma um valor de ganho (multiplicado ao valor de

- entrada) e um valor de bias (que é somado à entrada), que podem ser ajustados para efetuar correções ou modificações necessárias para o cálculo.
- Bloco Entrada Analógica (AI) – A principal função deste bloco é disponibilizar um valor digital para outros blocos a partir de um valor analógico. Para isso, o bloco AI recebe dados de um transdutor e faz a conversão do valor para a escala correta, de acordo com parâmetros especificados pelo usuário.
 - Bloco Caracterizador de Sinais (SCHAR) – Esse bloco tem a capacidade de implementar uma função genérica, na qual a saída relativa a uma dada entrada é baseada na interpolação não-linear de um conjunto de 20 pares de valores x e y.
 - Bloco Seletor de Entradas (ISEL) – Esse bloco tem a função de receber quatro entradas e retornar uma saída baseando-se em um critério de escolha, como a maior entrada, a menor ou a média entre elas. Existe ainda a opção de retornar o índice relativo à entrada escolhida, ao invés da entrada propriamente dita.

Capítulo 4

Predição de Séries Temporais

4.1 Introdução

Uma série temporal é um conjunto de observações feitas sequencialmente ao longo do tempo. A principal característica desse tipo de informação é que os dados são dependentes dos valores vizinhos. É importante analisar e modelar essa dependência, levando em conta também que a ordem em que aparecem os dados é crucial para o estudo do sistema. Deve-se notar que o tempo pode ser substituído por outras variáveis, tais como distância, profundidade, etc [Ehlers, 2007].

O estudo de séries temporais é útil em várias áreas do conhecimento, como por exemplo na economia (valores diários de ações, taxa mensal de desemprego), na medicina (eletrocardiograma), meteorologia (temperatura diária, velocidade do vento), ou como visto nesse trabalho, no estudo e avaliação de processos (medição da temperatura de gás natural), entre outras áreas.

4.2 Classificação de Séries Temporais

Uma série temporal é classificada como contínua quando suas observações são feitas continuamente no tempo. Definindo um conjunto $T = \{t : t_1 < t < t_2\}$, a série temporal será definida como $\{X(t) : t \in T\}$. Já uma série temporal discreta ocorre quando as observações são feitas em instantes específicos, normalmente em intervalos de tempo iguais. Dado um conjunto $T = \{t_1, t_2, \dots, t_n\}$, a série temporal será definida como $\{X(t) : t \in T\}$ [Ehlers, 2007].

É importante notar que a variável observada X pode assumir tanto valores discretos como contínuos. Em muitas situações, X pode ser discreta por definição, mas pode ser tratada como contínua se o número de valores observados não for muito pequeno. Por outro lado, séries temporais contínuas podem ser discretizadas, ou seja, seus valores só são registrados em certos intervalos de tempo.

Uma série temporal também pode conter mais de uma variável se k variáveis são observadas a cada instante de tempo. Neste caso, várias séries correlacionadas devem ser analisadas conjuntamente, ou seja, em cada instante de tempo, tem-se um vetor de observações de tamanho igual ao número de variáveis observadas.

4.3 Características das Séries Temporais

Séries temporais possuem algumas características particulares [Ehlers, 2007], tais como:

- Necessidade de técnicas específicas para serem analisadas;
- É indispensável considerar a ordem temporal das observações;
- Fatores como presença de tendências e variação sazonal ou cíclica podem ser difíceis de tratar ou estimar;
- É mais difícil de lidar com observações perdidas e informações discrepantes devido à sua natureza sequencial.

O estudo de séries temporais pode ter vários objetivos dependendo da aplicação, tais como:

- Descrever propriedades da série, tais como o padrão de tendência, existência

de variação sazonal ou cíclica, alterações estruturais, etc;

- Explicar a variação de outra série;
- Predizer valores futuros da série, afim de tomar alguma decisão, como por exemplo, comprar e vender ações ou aumentar o estoque de produtos a serem vendidos, ou como nesse trabalho, comparar o valor previsto com o valor de saída de um sensor, com o objetivo de detectar falhas no sistema.

4.4 Predição de Séries Temporais Utilizando uma Rede Recorrente NARX

Existem várias formas de realizar a predição de valores futuros em séries temporais, nesse trabalho foi utilizada uma rede NARX para realizar essa tarefa.

Dada uma série temporal, $\{x(n)\}$, $n=1:N$, define-se o regressor de entrada $U(n)$ da rede NARX como [Júnior, 2006]:

$$U(n) = [u(n), u(n-1), \dots, u(n-du + 1)],$$

$$U(n) = [x(n), x(n-\tau), \dots, x(n-(dE-1)\tau)],$$

Onde o regressor $U(n)$ é composto de dE valores observados da série a cada τ instantes de tempo. Para levar em conta a informação dinâmica fornecida pelo laço de realimentação, a rede NARX pode ser treinada nos modos paralelo e série-paralelo.

No modo paralelo, o regressor do sinal de saída $Y(n)$ é definido como:

$$Y(n-1) = [y(n-1), y(n-2), \dots, y(n-dy)],$$

Onde $Y(n)$ compreende dy saídas anteriores da rede neural. É importante notar que, para uma rede treinada adequadamente, a saída da rede no instante n é uma estimativa do valor futuro da série, ou seja, $y(n) = \hat{x}(n+1)$.

Já no modo série-paralelo (ou não-recorrente), o regressor de saída é definido da seguinte maneira:

$$Y(n-1) = [x(n), x(n-1), \dots, x(n-dy + 1)].$$

Sendo então construído com amostras reais da série temporal de interesse. Qualquer que seja o modo de uso da rede NARX, ambas as arquiteturas implementam o seguinte mapeamento entrada-saída:

$$Y(n) = \hat{x}(n+1) = \hat{f}[Y(n-1), U(n)],$$

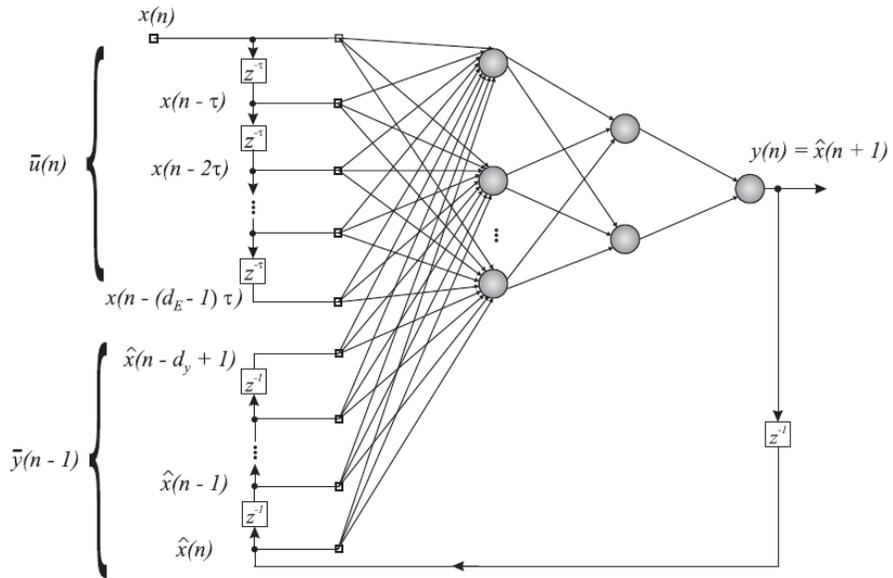


Figura 4.1 – rede NARX no modo paralelo

Em que a função não-linear $\tilde{f}(\cdot)$ pode ser realizada pela rede perceptron de múltiplas camadas padrão e treinada pelo algoritmo *backpropagation* simples.

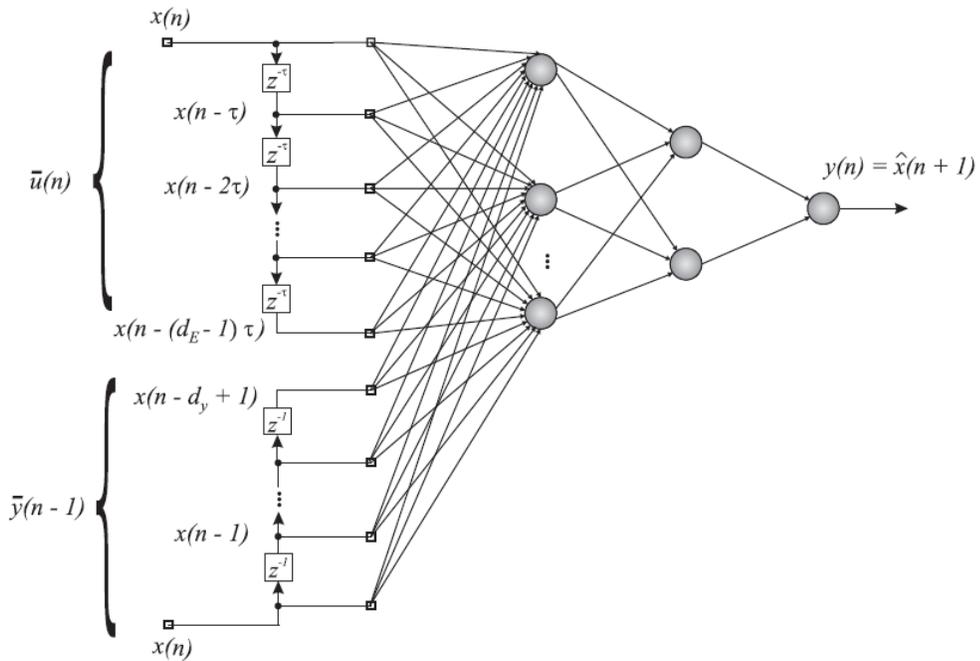


Figura 4.2 – rede NARX no modo série-paralelo

Capítulo 5

Redes Neurais na Rede Foundation Fieldbus

5.1 Construção de um neurônio artificial

Para a construção de uma rede neural, são necessários vários neurônios artificiais interligados, formando a arquitetura desejada, que nesse trabalho é uma rede neural recorrente NARX. Como não existe um bloco funcional padrão com a finalidade de implementar um neurônio no ambiente Foundation Fieldbus, foi necessário desenvolver a rede neural utilizando apenas blocos funcionais padrões existentes [Silva, 2005].

Para implementar a rede neural no ambiente FF, foram utilizados dois tipos de blocos funcionais: blocos aritméticos e blocos caracterizadores de sinal (SCHAR), necessários para implementar a função tangente sigmóide. Desse modo, o bloco SCHAR foi configurado com 20 pares de pontos x e y com o objetivo de mapear a tangente sigmóide no intervalo requerido de modo bastante aproximado. Para melhor sintonizar os blocos caracterizadores, foi utilizada uma técnica de inteligência artificial chamada algoritmos genéticos.

Para implementar o neurônio artificial, foram utilizados blocos aritméticos para aplicar os pesos e bias aos valores de entrada, definindo o parâmetro "algorithm type" como somador tradicional, que multiplica cada entrada por um peso e faz a soma desses produtos com um valor de bias. As saídas dos blocos aritméticos são então ligados às entradas dos blocos SCHAR para calcular a função de ativação. O esquema de um neurônio está mostrado na figura 5.1

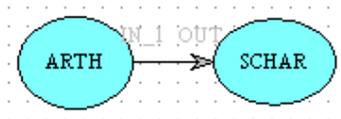


Figura 5.1 – Esquema de um neurônio na rede FF

5.2 Criação de bancos atrasadores

Por meio de alguns testes e do estudo do funcionamento da rede Foundation Fieldbus, foi observado que é possível gerar atrasadores alocando blocos funcionais em instrumentos diferentes, de modo que cada vez que um dado é transmitido de um instrumento para outro, esse dado chega um instante de tempo atrasado em relação a um dado transmitido entre blocos alocados em um mesmo instrumento. Um exemplo de teste será descrito a seguir.

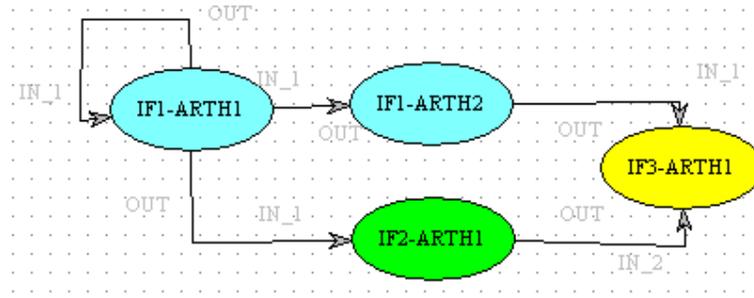


Figura 5.2 – Exemplo de teste para ilustrar o comportamento do atraso

No teste realizado, o bloco IF1-ARTH1, alocado no instrumento IF1, tem valor inicial de saída igual a zero, tendo esse valor incrementado com o passar do tempo, por meio de realimentação, ligando o valor de saída ao valor de entrada, e aplicando um ganho e um bias iguais a 1. Esse valor de saída é repassado para os blocos IF1-ARTH2 (alocado no instrumento IF1) e IF2-ARTH1 (alocado no instrumento IF2). Feito isso, um outro bloco funcional, o IF3-ARTH1 (alocado no instrumento IF3), recebe os valores de saída dos blocos IF1-ARTH2 e IF2-ARTH1 e subtrai o valor do primeiro pelo do segundo. Como o resultado dessa subtração é sempre igual a 1, conclui-se que a saída do bloco IF1-ARTH1 chega ao bloco IF2-ARTH1 um instante de tempo depois do bloco IF1-ARTH2. Esse teste foi repetido gerando-se vários atrasadores, com dados sendo repassados a diferentes blocos funcionais, gerando atrasos cada vez maiores, sempre obtendo o resultado esperado.

De posse do resultado dos testes, pode-se perceber que é possível projetar os bancos atrasadores de modo a obter os atrasos desejados, sendo necessário apenas aumentar o número de blocos funcionais, alocando-os em diferentes instrumentos, de acordo com a necessidade.

5.3 Implementação da rede recorrente

Tendo visto como implementar um neurônio artificial e os bancos atrasadores, foi possível implementar a rede recorrente NARX, com 2 atrasos na entrada e 3 na saída, além de 1 camada oculta com 3 neurônios e função de ativação tangente sigmóide, e 1 neurônio na camada de saída. De posse desses dados, pode-se observar que a mesma é equivalente a uma rede perceptron de múltiplas camadas com arquitetura 6-3-1, sendo os 6 neurônios de entrada referentes à entrada e aos atrasos previamente definidos. Deve-se considerar também que como a rede está distribuída por vários instrumentos diferentes, a informação demora um certo tempo para ser recebida na entrada, processada e disponibilizada na saída, a rede deve ser treinada levando em conta esse atraso adicional. Fazendo um teste similar ao teste para verificar os atrasos em instrumentos diferentes, modificando os blocos funcionais de modo a simplesmente passar o valor recebido adiante, foi constatado que a saída referente a uma entrada disponibilizada no tempo t só vai ser retornada no instante $t+2$, com isso, foi treinada a rede NARX com as entradas x , $x-1$ e $x-2$, e as saídas $y-3$, $y-4$ e $y-5$.

Um esquema da rede neural embarcada na rede Foundation Fieldbus será mostrado na figura 5.3.

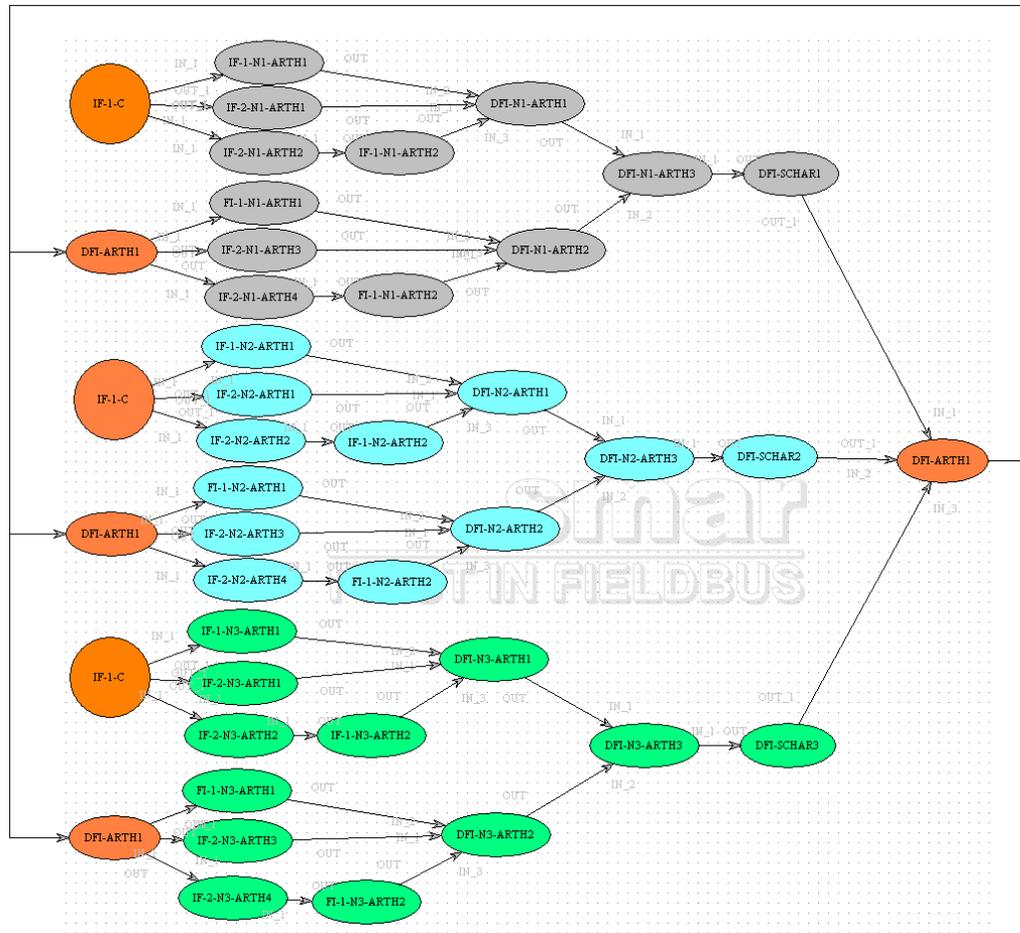


Figura 5.3 – Esquema da rede NARX implementada na rede FF

Na qual os blocos cor de laranja possuem os valores de entrada e saída da rede, os blocos cinza implementam o primeiro neurônio, os blocos azuis implementam o segundo neurônio e os blocos verdes implementam o terceiro neurônio artificial da rede recorrente.

Pode-se ver que no esquema que o bloco funcional IF-2-N1-ARTH1 é responsável por gerar um atraso na camada de entrada e os blocos funcionais IF-2-N1-ARTH2 e IF-1-N1-ARTH2 em conjunto geram dois atrasos nessa camada.

Capítulo 6

Procedimentos Experimentais

6.1 Organização da Arquitetura e da Comunicação

Para a realização dos experimentos, dispõe-se de um laboratório didático composto por um computador, funcionando como estação de configuração e supervisão, uma rede Ethernet, uma ponte (DFI-302), que faz a interligação entre as redes Ethernet e Foundation Fieldbus e desempenha o papel de mestre da rede FF, e oito dispositivos de campo, onde três deles foram utilizados para a implementação da rede neural recorrente.

Para fazer a comunicação entre o Matlab e o Syscon (programa responsável pela configuração da rede FF), a fim de transmitir os dados da série temporal para os instrumentos e avaliar a resposta dos mesmos, foi utilizado um toolbox do Matlab, o “opctool”, capaz de fazer a comunicação entre os dois *softwares* via OPC.

6.2 Treinamento e Avaliação das Redes Neurais no Matlab

O primeiro passo dos procedimentos experimentais foi treinar algumas redes neurais, variando a quantidade de passos adiante a serem previstos. Como dito na seção 5.3 deste trabalho, a arquitetura de todas as redes neurais contava com três neurônios na camada oculta e um neurônio na camada de saída. Essa arquitetura foi escolhida, pois ela foi utilizada no trabalho escrito por [de Medeiros, 2009]. As redes tinham como função de ativação a tangente sigmóide.

Para realizar o treinamento das redes neurais, foi utilizado um toolbox para Matlab que faz o treinamento de redes neurais do tipo perceptron de múltiplas camadas (mlp) utilizando o filtro de Kalman estendido. Como esse toolbox não foi feito para treinar redes neurais recorrentes, foi criada uma rede perceptron de múltiplas camadas adaptada para ser treinada da mesma forma que uma rede NARX no modo série-paralelo. Tal adaptação consistiu em replicar a forma como a rede neural foi embarcada na rede Foundation Fieldbus, criando uma mlp com 6 neurônios na camada de entrada, sendo os 3 primeiros correspondentes às entradas atrasadas vindas do sensor e os 3 últimos correspondentes aos valores retornados da realimentação. Para fazer o treinamento, os conjuntos de entrada e saída eram compostos por um vetor de 6 números como entrada e 1 valor de saída correspondente a esse vetor.

Foi realizado o treinamento de redes neurais para a previsão de 1 até 10 passos na série temporal, com o sinal sem falhas, sendo utilizados 499 pontos para o treinamento e 168 pontos para validação, totalizando então 667 pontos. Depois de terminados os treinamentos, foram feitas simulações com as redes treinadas. Pode-se ver o resultado das simulações nas figuras 6.1, 6.2 e 6.3. Como critério de avaliação, foi feito um cálculo de correlação cruzada entre os valores de saída do sensor e do preditor. Os resultados serão expostos a seguir.

Tabela 6.1 – Correlações

Passos adiante utilizados na predição	Correlação com o valor real
1	0.9826
2	0.9621
3	0.9314
4	0.8875
5	0.8387
6	0.7848
7	0.7091
8	0.6374
9	0.5872
10	0.4930

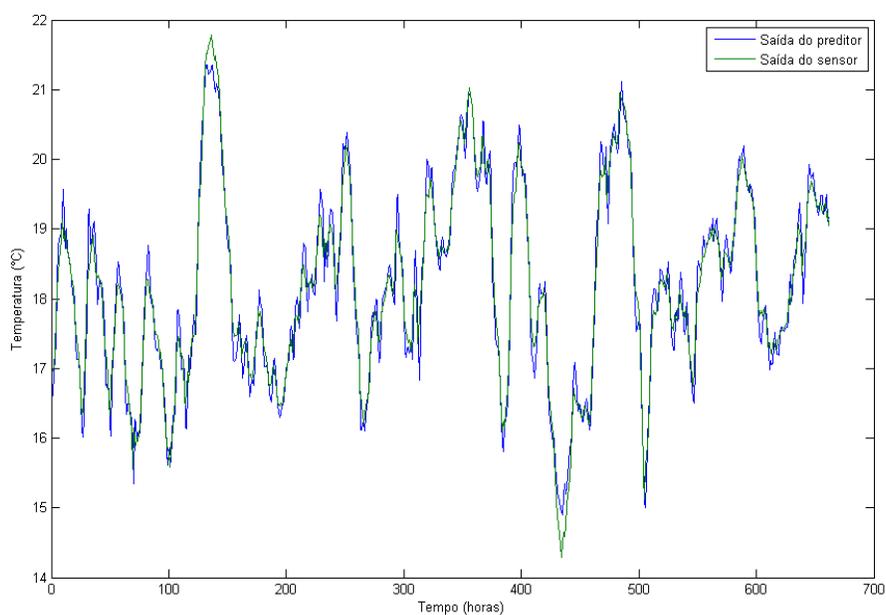


Figura 6.1 – Predição de 1 passo

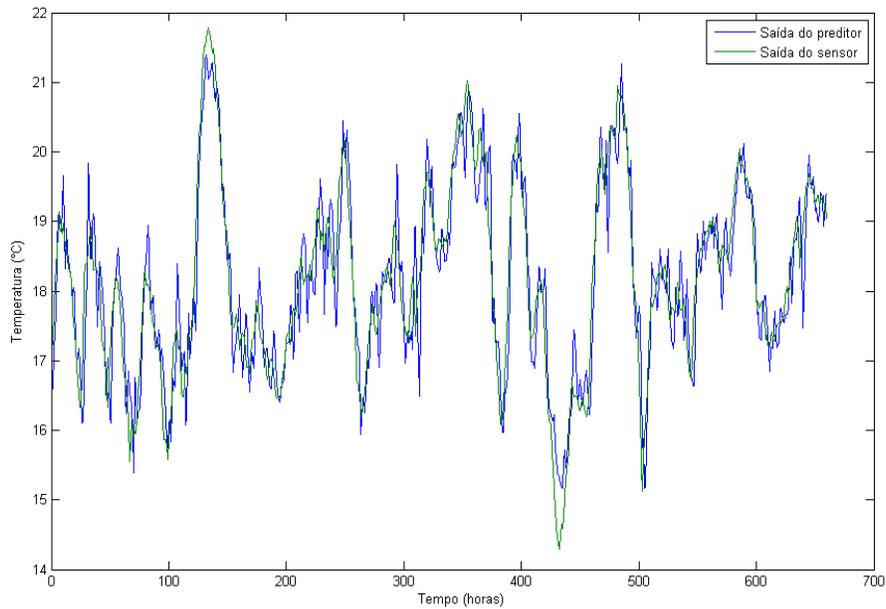


Figura 6.2 – predição de 3 passos

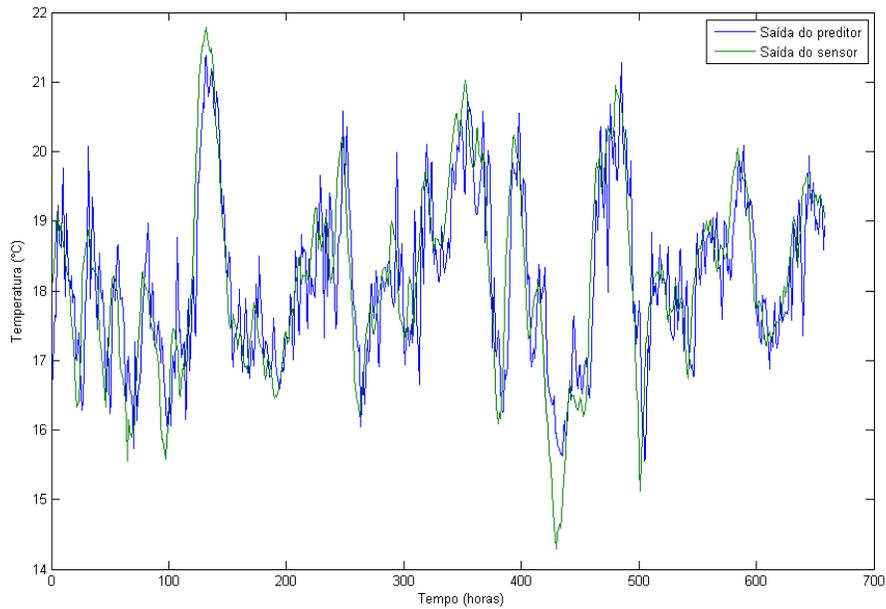


Figura 6.3 – predição de 5 passos

Pode-se ver pela tabela e pelas imagens que a rede neural é capaz de realizar a predição com uma forte correlação com a saída desejada para até 5 passos adiante na série temporal. O segundo passo foi fazer a simulação das redes neurais com falhas simuladas. Foram simulados quatro tipos de falhas, definidas como:

- Falha 1: Valor zero. $Y_t(n) = 0$.

- Falha 2: Valor de Fundo de escala. $Y_f(n) = 60$.
- Falha 3: Deriva de fator de escala. $Y_f(n) = Y(n)e^{-\alpha k}$, tal que $\alpha = 0.06$; a variável k assume valor 1 no instante em que se inicia o estado da falha e é incrementada a cada instante de tempo subsequente.
- Falha 4: Deriva de Offset. $Y_f(n) = Y(n) - \alpha k$, tal que $\alpha = 0.6$, correspondente a 1% do valor de Fundo de Escala; a variável k assume valor 1 no instante em que se inicia o estado da falha e é incrementada a cada instante de tempo subsequente.

Definidos os tipos de falhas a serem simuladas, foram feitos testes com cada tipo de falha, fazendo a predição de 1 e 2 passos adiante na série temporal. O resultado dos testes pode ser visto nas figuras 6.4 a 6.11.

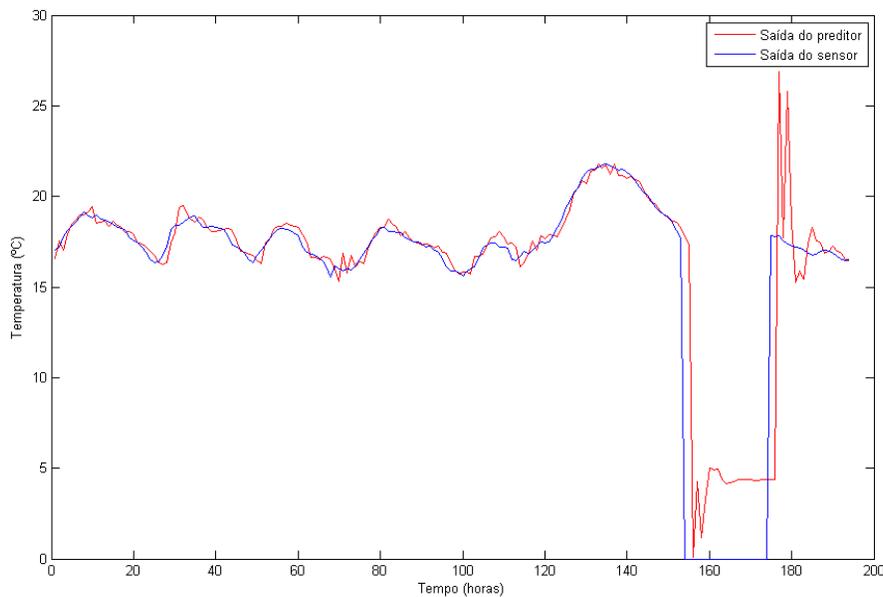


Figura 6.4 – Preditor de 1 passo para uma série com a Falha 1

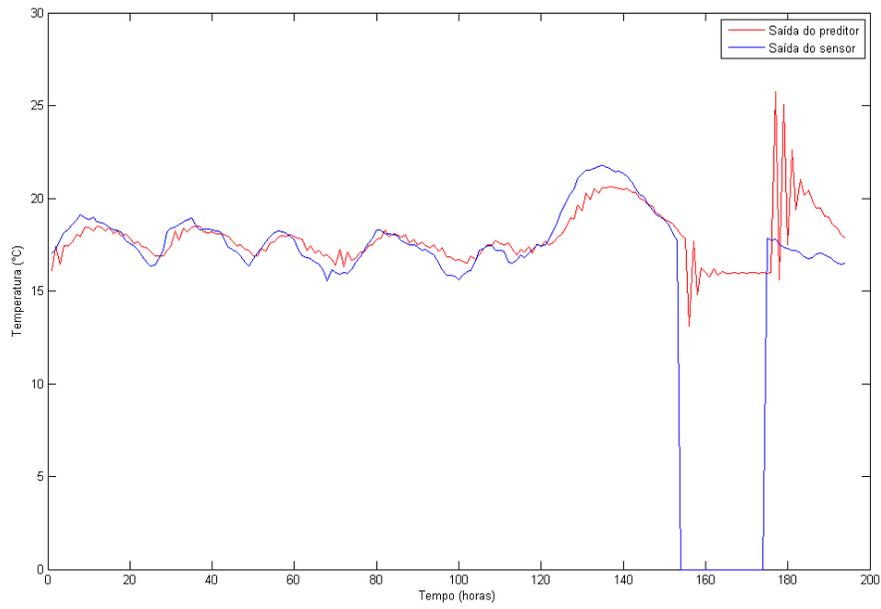


Figura 6.5 – Preditor de 2 passos para uma série com a Falha 1

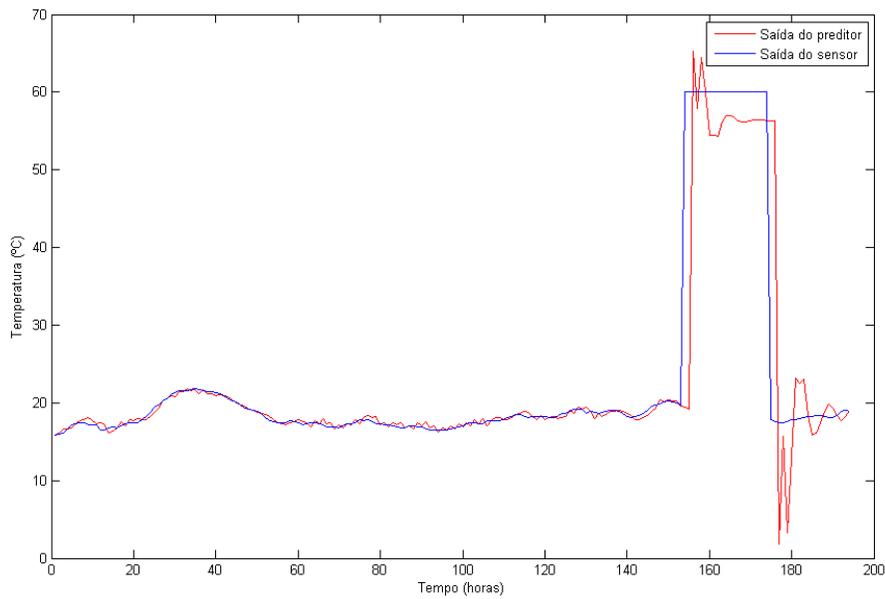


Figura 6.6 – Preditor de 1 passo para uma série com a Falha 2

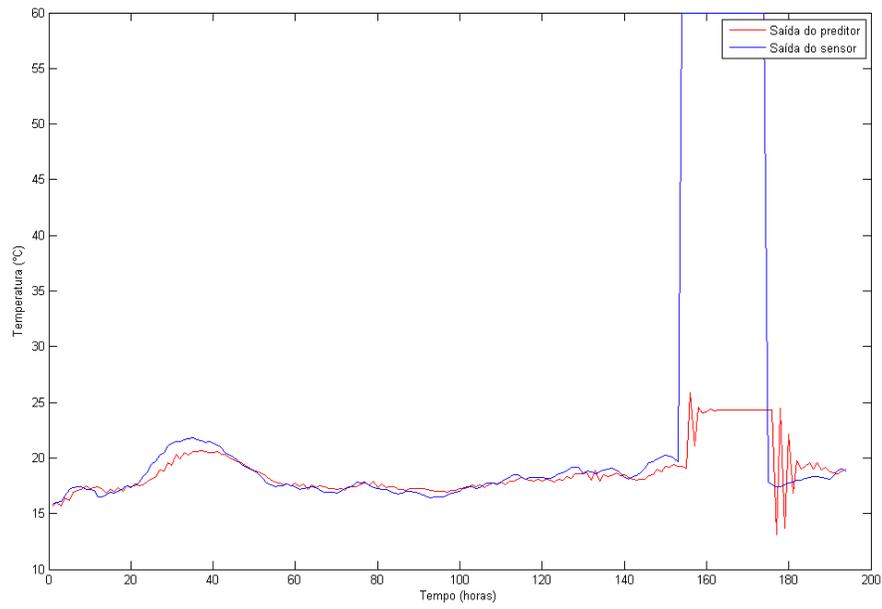


Figura 6.7 – Preditor de 2 passos para uma série com a Falha 2

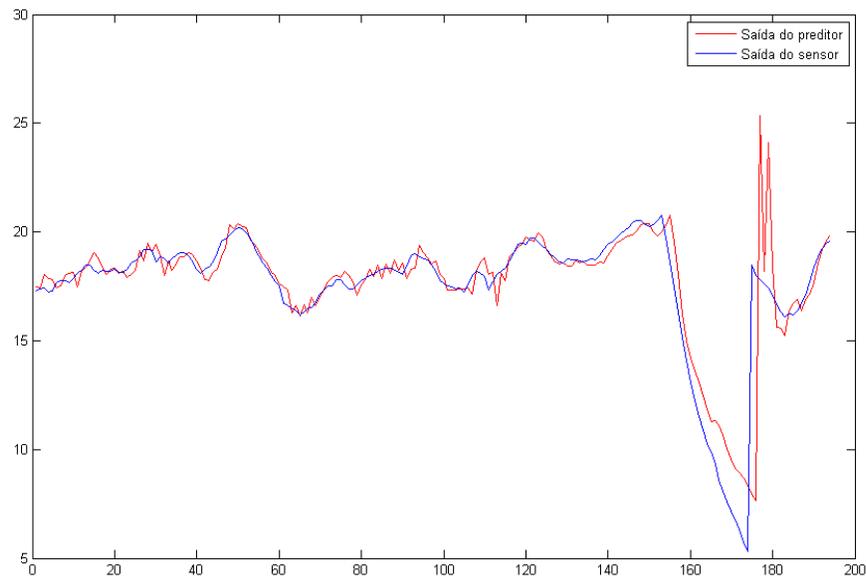


Figura 6.8 – Preditor de 1 passo para uma série com a Falha 3

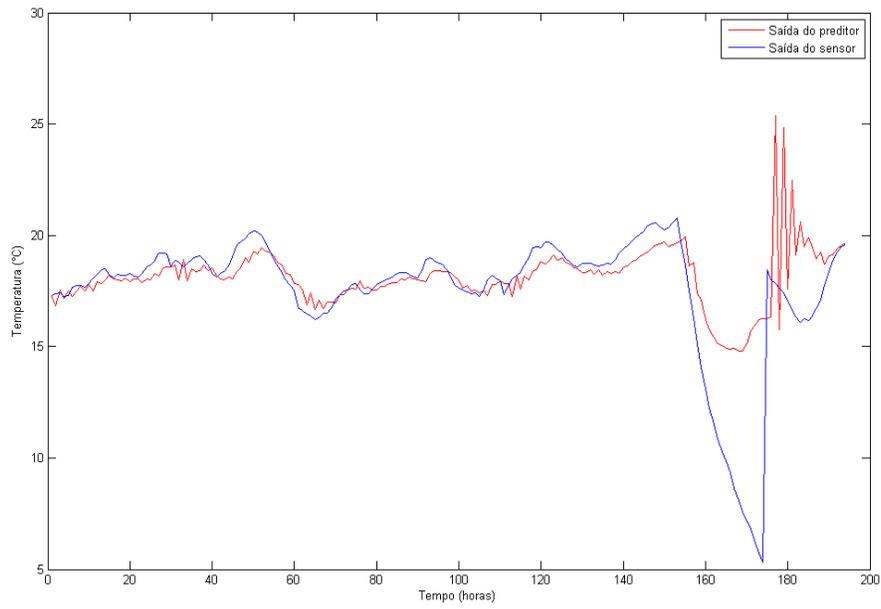


Figura 6.9 – Preditor de 2 passos para uma série com a Falha 3

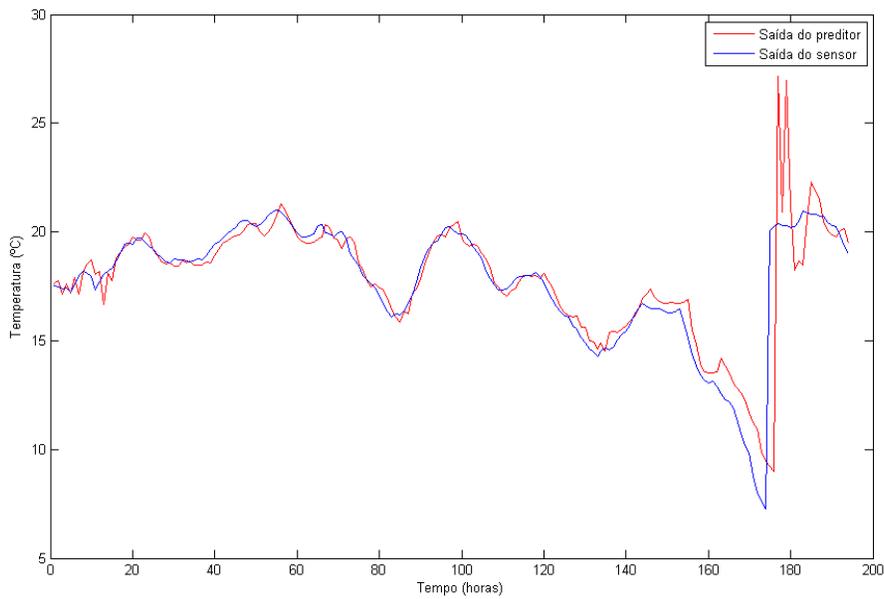


Figura 6.10 – Preditor de 1 passo para uma série com a Falha 4

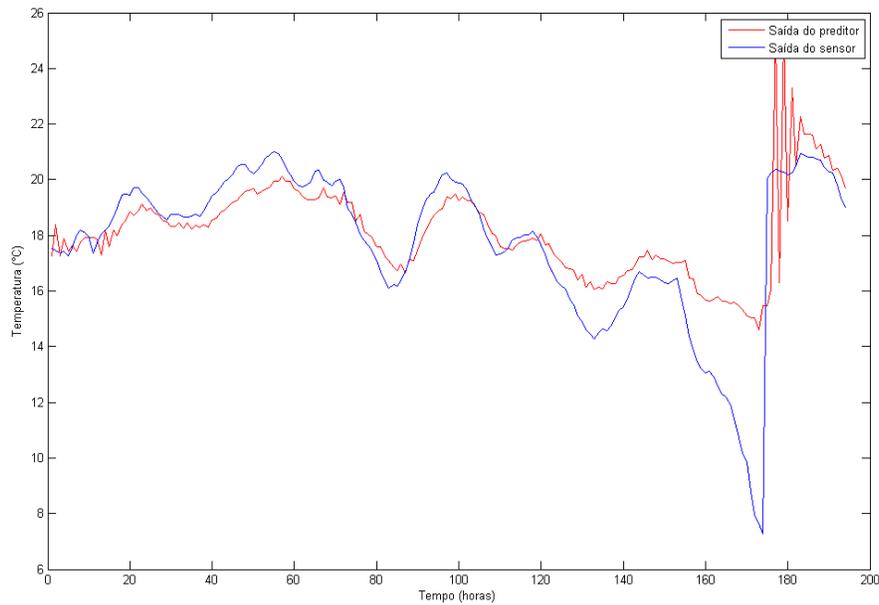


Figura 6.11 – Preditor de 2 passos para uma série com a Falha 4

6.3 Embarcar a Rede Neural na Rede Industrial

Depois feitos os testes, foi escolhida a rede neural com predição de 1 passo adiante, já que a mesma é mais estável e também apresenta um valor de saída diferente da saída do sensor quando o mesmo apresenta falhas.

Para que a rede neural previamente treinada seja embarcada no ambiente industrial, a rede Fieldbus foi configurada com os blocos funcionais de acordo com o esquema da figura 5.3, encontrada na seção 5.3. É importante notar que essa mesma configuração pode ser utilizada para a predição de qualquer quantidade de passos adiante na série temporal, já que apenas os valores dos pesos são modificados. Depois que os instrumentos foram configurados, os pesos da rede neural foram exportados para a rede industrial para que os testes pudessem ser realizados. Os resultados dos testes podem ser vistos nas figuras 6.12 a 6.14.

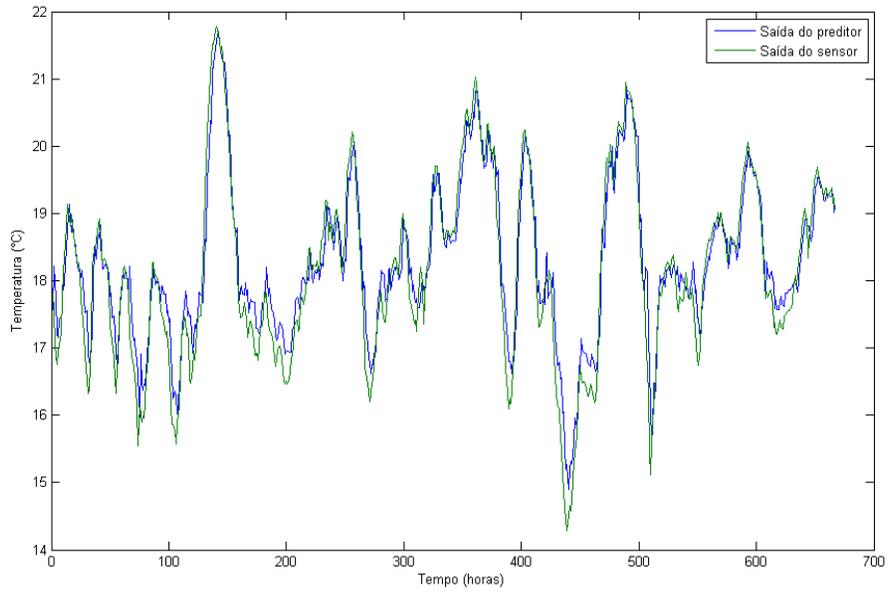


Figura 6.12 – Predição de 1 passo de uma série sem falhas, na rede FF

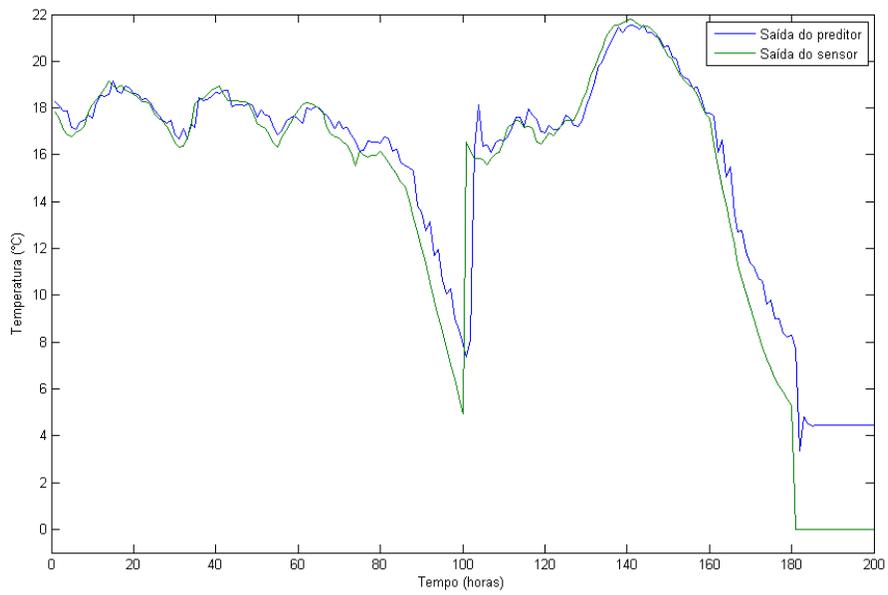


Figura 6.13 – Predição de 1 passo de uma série com falhas dos tipos 4 e 1, na rede FF

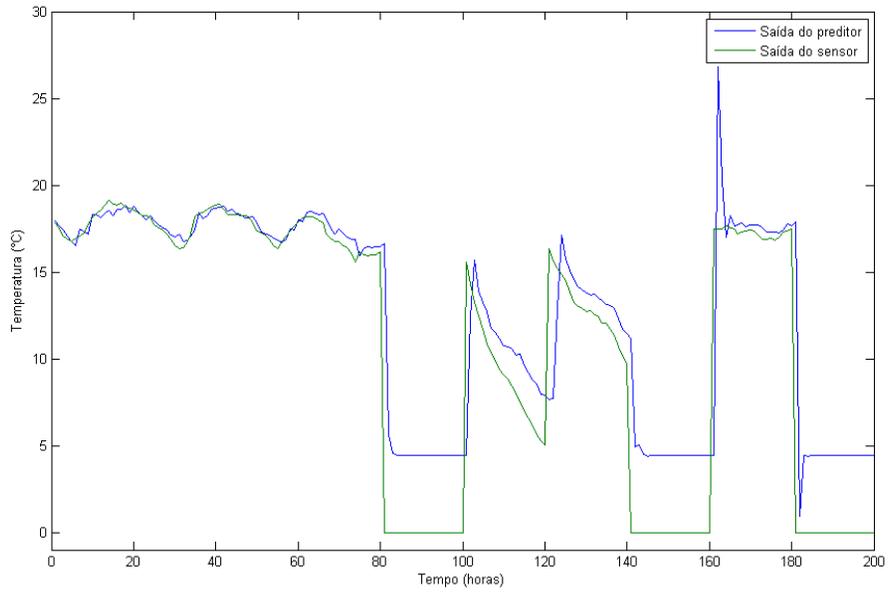


Figura 6.14 – Predição de 1 passo de uma série com falhas dos tipos 3 e 1, na rede FF

Capítulo 7

Conclusão

7.1 Considerações Finais

O gás natural é um produto que deve ter as algumas de suas variáveis, como temperatura, vazão e pressão, constantemente medidas, e essas medições devem ter uma precisão bastante acurada, já que qualquer erro de medição pode acarretar em prejuízos financeiros durante o processo de transferência de custódia.

Neste trabalho, foram desenvolvidos algoritmos inteligentes, baseados em redes neurais recorrentes, capazes de fazer a predição dos valores de saída de um sensor de temperatura. O valor retornado por esse preditor pode ser comparado com a saída real do sensor, de modo que, caso a diferença entre os dois sinais de saída seja significativa, existe a possibilidade de ocorrência de uma falha no sensor. Além de ser usada para detecção de falhas, a diferença entre os valores de saída do preditor e do sensor pode ser utilizada para a classificação de falhas, já que as diferentes falhas geram comportamentos diferentes no sensor e no preditor.

Além de ser implementada no computador, a rede neural recorrente foi embarcada em uma rede industrial Foundation Fieldbus, que é uma rede de instrumentos com capacidade de processamento instalados em campo, e que funcionam de modo paralelo e não necessitam de um elemento centralizador para que a tomada de decisões seja feita.

Foram feitos testes com algumas estratégias diferentes de predição e diversas falhas simuladas, onde foram obtidos resultados satisfatórios, que demonstraram que a rede neural pode realizar a função de preditor de séries temporais, assim como foi visto que é possível embarcar redes neurais recorrentes no ambiente industrial Foundation Fieldbus.

7.2 Perspectivas Futuras

Algumas sugestões para trabalhos futuros são o estudo mais aprofundado e melhoria do processo de simulação de falhas, gerando mais tipos de falhas simuladas e a aplicação aplicar desse trabalho na criação de sensores de *software* aplicados à produção de derivados de petróleo, utilizando o *software* Hysys para simular os processos estudados e a rede FF para realizar as operações com instrumentos reais.

Referências Bibliográficas

Adelson S. Carvalho, Ronald C. Silva, Dênis B. Nascimento (2009), Sistema de Comunicação OPC para uma Coluna de Destilação Piloto, 'Simpósio de Excelência em Gestão e Tecnologia', Resende, RJ.

Ehlers, Ricardo (2007), 'Análise de Séries Temporais'.

Fábio S. Lima, Luiz A. Guedes, Andres O. Salazar, André L. Maitelli (2004), Hybrid Environment for Tests and Training in Fieldbuses, 'VI Induscon 2004', Joinville, SC.

Fernandes, Raphaela Galhardo (2007), 'Detecção e Isolamento de Falhas em Sistemas Dinâmicos Baseados em Redes Neurais', Dissertação de mestrado, Universidade Federal do Rio Grande do Norte.

Filho, Constantino Seixas, 'Foundation Fieldbus', Apostila da disciplina 'Sistemas distribuídos para automação', Universidade Federal de Minas Gerais.

Haykin, Simon (2005), 'Neural Networks. A Comprehensive Foundation'.

Júnior, José Maria Pires de Menezes (2006), 'Redes Neurais Dinâmicas para Predição e Modelagem não-linear de séries temporais', Dissertação de mestrado, Universidade Federal do Ceará.

Manguinho, Diego Alessandro Paiva de Morais (2008), 'Sistema de Monitoração de Sensores Inteligentes em rede Foundation Fieldbus para melhoria dos processos de medição e controle na indústria do petróleo', monografia de conclusão de curso de Engenharia de Computação, Universidade Federal do Rio Grande do Norte.

Martins, Daniel Lopes (2008), 'Implementação de um demultiplexador aplicado ao ambiente Foundation Fieldbus', monografia de conclusão de curso de Engenharia de Computação, Universidade Federal do Rio Grande do Norte.

de Medeiros, Juliana Pegado (2009), 'Estudo e Implementação de Algoritmos Inteligentes para Detecção e Classificação de Falhas na Medição de Gás Natural', Dissertação de mestrado, Universidade Federal do Rio Grande do Norte.

Rodrigues, Bruno (2004), 'Gasoduto Virtual. Uma Alternativa para Expansão do Uso do Gás Natural', monografia de conclusão de curso de Administração de Empresas, Universidade Paulista.

Silva, Diego Rodrigo Cabral (2005), 'Redes Neurais Artificiais no Ambiente de Redes Industriais Foundation Fieldbus Usando Blocos Funcionais Padrões', Dissertação de mestrado, Universidade Federal do Rio Grande do Norte.