

UM NEUROCONTROLADOR COM TREINAMENTO EM TEMPO REAL APLICADO A UMA PLANTA DE TEMPERATURA

ANDRÉ LAURINDO MAITELLI, JOSÉ AUGUSTO DANTAS DE REZENDE

LECA – DCA - CT - UFRN
59072-970 Natal/RN Brasil

E-mails: maitelli@dca.ufrn.br, jaugusto@telemar.com.br

Resumo— Neste artigo apresentamos uma metodologia de controle neural em um sistema de temperatura. O controlador usa uma série de métodos de treinamento otimizados para melhorar a velocidade de convergência. O treinamento é feito totalmente em tempo real e a arquitetura da rede neural é a Perceptron de Múltiplas Camadas. Apresentamos os detalhes do sistema de temperatura e do método de controle, bem como os métodos de otimização que permitem o treinamento em tempo real da rede neural. Os resultados apresentados mostram a capacidade do controlador neural em problemas que envolvem dificuldades como não linearidade e mudança na sua estrutura.

Abstract— In this paper we present a neural network control approach of a temperature system. The controller uses a serie of optimization training methods for improve the speed of its convergence. The training is made totally on-line and the neural network architecture is the Multi-layer Perceptron. We present the details of the temperature system and the control method, as well the optimization methods which permits the on-line training of the neural network. The results show in this article prove the capacity of the neural controllers in problems which involve difficulties like non-linearities and changes on its structure.

Keywords—Neural Networks, Neurocontrol, On line Training and Optimization.

1. Introdução

Dentre as técnicas de controle inteligente, uma que tem ganhado mais espaço e se mostra como uma das mais promissoras para as próximas décadas é a de Neurocontrole. Até pouco tempo atrás, aplicações que envolvessem treinamento *on-line* de RNA's não eram possíveis, devido ao lento processo de treinamento e convergência das mesmas. Com o desenvolvimento da tecnologia da informática e a proposição de novas técnicas de otimização dos processos de treinamento, aplicações com treinamento *on-line* passaram a ser possíveis.

O treinamento *on-line* de neurocontroladores confere a rede uma série de vantagens e desvantagens. Entre as vantagens, a principal está associada a capacidade de adaptatividade que passa a ser conferida ao controlador. Caso haja alguma mudança na estrutura da planta, a rede será capaz de absorver estas mudanças e continuar efetuando um controle eficiente. Entre as desvantagens, podemos colocar o desempenho do controlador durante o início do processo de treinamento. Não estando ainda suficientemente bem treinada, a rede não é capaz de executar um controle inicial eficiente.

Este trabalho pretende apresentar a aplicação de um sistema de controle digital proposto originalmente em [1] a uma planta de temperatura construída em nosso laboratório. O que se quer demonstrar é a capacidade do controlador em executar tarefas em um tempo de treinamento relativamente baixo e com uma boa performance de controle. Vale salientar que nesta técnica de Neurocontrole não fazemos uso de nenhum artifício como o pré-treinamento. Todo o trei-

namento da planta é feito entre os períodos de amostragem. A planta a ser controlada é não-linear e apresenta atraso de transporte que pode variar com o tempo.

Neste trabalho, foi desenvolvido um software que permite a aplicação do NeuroControlador a qualquer planta física. Todo o desenvolvimento foi feito utilizando a linguagem Borland C e a interface roda no MS-DOS, o que confere ao controlador características de velocidade adequadas para uma aplicação em tempo real.

2. Planta Física de temperatura

A planta física do controle de temperatura em questão pode ser entendida a partir do esquema representativo apresentado abaixo:

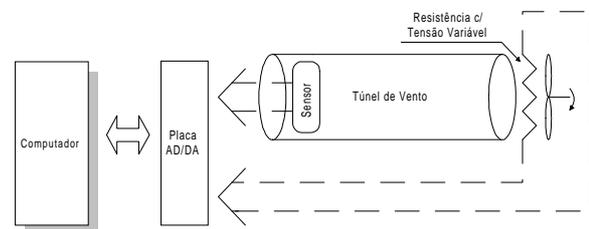


Figura 1: Esquema da planta de temperatura

A planta consiste basicamente de um ventilador alimentado por um fonte de 6 Vcc, que mantém a rotação praticamente constante. A frente do ventilador, temos uma resistência sobre a qual variamos a tensão aplicada. A variação desta tensão implica num maior ou menor aquecimento da mesma, variando dessa forma a temperatura ao longo do túnel de ven-

to. Ao final do túnel de vento temos um sensor formando por um arranjo que aciona um transistor do tipo 2N2222. Com a variação de temperatura, varia também a tensão coletor-emissor do mesmo, essa variação é ampliada através de um sistema de amplificadores operacionais que condicionam o sinal a uma faixa de 0 a 5V. Esse sinal é colhido através de uma placa analógica-digital do tipo *MultiQ*, da *Quanser Consulting*. O sinal é então entregue ao computador, onde é feita a implementação digital do Controlador Neural.

O controlador, através dos sinais lidos pelo sensor, gera o sinal de controle apropriado, também na faixa de 0 a 5V e o entrega de volta, via conversão digital-analógica, à planta, que irá, através de um amplificador operacional, aumentar a faixa do sinal para 0 a 10V e, através desse sinal, gatilhar o *gate* de tiristor que será responsável pela regulação da tensão aplicada sobre a resistência. Em ensaios realizados no laboratório, verificamos que a mínima temperatura obtida pela planta ficava em torno dos 26 °C (temperatura ambiente), quando não havia tensão aplicada sobre a resistência e, portanto, não havia aquecimento da mesma. A máxima temperatura obtida, quando a tensão aplicada sobre a resistência era de 220 V, ficou em torno dos 86 °C.

Um detalhe de projeto é a relação inversa entre a magnitude do sinal oriundo do controlador e o aquecimento da planta. Quanto maior o sinal de controle, mais tempo o tiristor passará sem conduzir e, portanto, menor a tensão média aplicada sobre a resistência, e menor a temperatura da mesma. De modo inverso, quanto menor o sinal de controle, mais tempo o tiristor estará conduzindo e maior a tensão média e a temperatura da resistência.

3. Redes Neurais Artificiais

Os estudos acerca das Redes Neurais Artificiais surgiram da tentativa de se desenvolver um modelo matemático que procurasse simular o comportamento do cérebro humano. O cérebro pode ser visto como um computador paralelo altamente complexo e não-linear. Ele tem a capacidade de organizar suas unidades básicas, os neurônios, em grupos que irão operar funções bastante específicas e complexas como a visão, reconhecimento de padrões e a coordenação motora. Todas estas tarefas, além de serem bastante complexas, são executadas pelo cérebro em um intervalo de tempo bastante baixo, além de ter uma eficiência energética extremamente alta. Portanto, modelos matemáticos que permitam a simulação, pelo menos em parte, de algumas dessas atividades desempenhadas pelo cérebro, são bem recebidos.

Como representação do modelo natural, a RNA possui uma série de características bastante semelhantes ao seu modelo inspirador. Entre essas características, podemos citar as apresentadas a seguir, que estão diretamente relacionadas às aplicações da rede em controle de sistemas, [2]:

1. *Não-Linearidade*. Um neurônio é basicamente um sistema não-linear. Conseqüentemente, uma rede neural consistindo da conexão de unidades não-lineares é também necessariamente não-linear. Esta característica é bastante importante, pois na maioria das aplicações práticas que surgem, os sistemas possuem não-linearidades.
2. *Mapeamento Entrada-Saída*. A Rede Neural absorve conhecimento, assim como em seu modelo inspirador biológico, através do processo de aprendizagem supervisionado. Apresentamos à RNA uma série de pares entrada-saída, chamados de exemplos de treinamento. Estas entradas são apresentadas à rede e sua saída é comparada com a saída presente no par de treinamento. A diferença entre a saída apresentada pela rede e a saída do par de treinamento é utilizada para a correção das sinapses. Uma série de exemplos de treinamento é apresentado à rede, de forma aleatória, até que os pesos da rede atinjam a estabilidade. Ao final desse processo, teremos a rede já treinada. Apresentamos, então, alguns pares ainda não apresentados à rede anteriormente. Poderemos observar que a rede terá realizado um mapeamento entrada-saída, apresentando em sua saída um valor coerente com a relação estabelecida pelos padrões entrada-saída dos exemplos de treinamento. Podemos imaginar a série de potenciais aplicações que surgem dessa característica da RNA.
3. *Adaptatividade*. A RNA tem a capacidade de adaptar suas sinapses, caso o ambiente ao qual está inserida modifique suas características. Uma RNA, treinada para operar em um determinado ambiente, pode ser facilmente re-treinada para operar em um outro sistema. Esta característica é bastante útil, especialmente se o ambiente em questão é variável e o treinamento é feito em tempo real. Essa característica alarga bastante as aplicações das RNA's em sistemas de filtragem e controle adaptativo, [3]-[4].

4. Neurocontrolador Implementado

O Neurocontrolador para sistemas SISO implementado, foi proposto originalmente em [1]. O esquema baseia-se no método de otimização em malha fechada. Consiste basicamente do uso de duas estruturas neurais, conhecidas como Neuro-emulador e Neuro-Controlador. O Neuro-Emulador é responsável pelo aprendizado da dinâmica da planta. Toda a informação absorvida pelo Neuro-Emulador será usada posteriormente no processo de treinamento do neuro-controlador que será a estrutura que efetivamente promoverá o controle da planta. O esquema abaixo ilustra o treinamento do neuro-emulador.

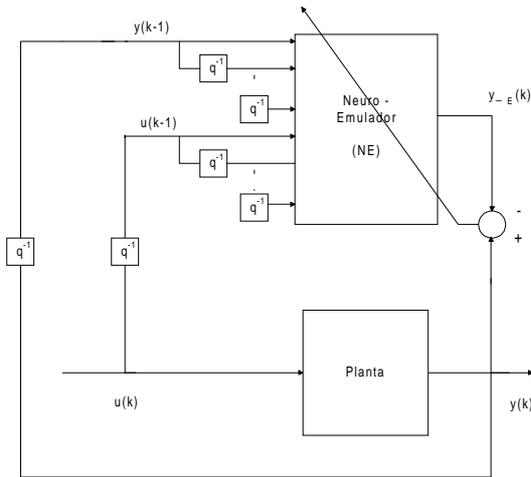


Figura 2: Treinamento do Neuro-emulador.

Primeiramente efetuamos o treinamento do NE junto a planta. Apresentamos em sua entrada os valores passados de entrada e saída da planta e comparamos a saída da rede com a saída atual da planta. Ao final do processo de treinamento, a rede deve ser capaz de repetir o comportamento da planta. O Neuro-emulador é então utilizado para o treinamento do Neuro-Controlador, como veremos na figura abaixo que ilustra o treinamento do Neuro-controlador.

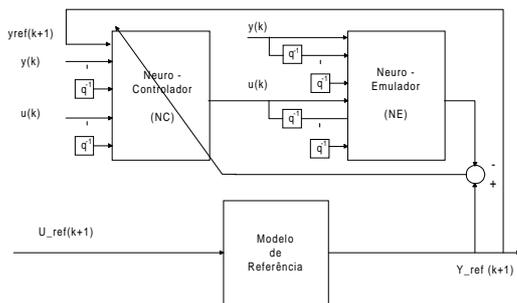


Figura 3: Treinamento do Neuro-controlador.

O Neuro-controlador aprende primeiramente a controlar o Neuro-emulador, fazendo-o acompanhar um sinal de referência. Após o treinamento, espera-se que o Neuro-Controlador esteja pronto para gerar os sinais de controle que serão colocados a entrada da planta para fazê-la seguir o modelo de referência.

Vê-se que o processo de treinamento do Neuro-Controlador só começa após concluído o processo de treinamento do Neuro-Emulador. O que ressalta ainda mais a necessidade de um algoritmo de treinamento rápido e eficiente, já que todo o treinamento se dará durante o período de amostragem. Veja abaixo o esquema da seqüência de treinamento das Redes Neurais entre duas amostragens.

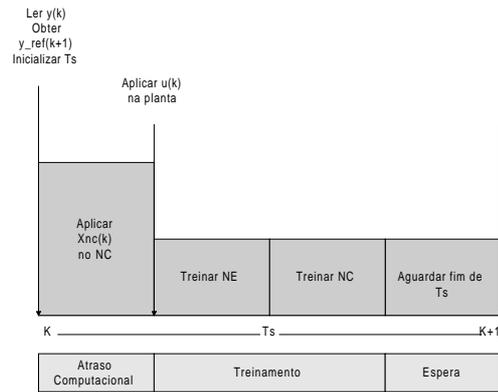


Figura 4: Diagrama temporal do treinamento do esquema de controle.

A arquitetura das Redes Neurais Artificiais utilizadas é do tipo Perceptron de Múltiplas Camadas com apenas uma camada intermediária e cinco neurônios na mesma.

5. Técnicas de Melhoria da Velocidade de treinamento

Como já colocamos anteriormente, o grande problema associado às Redes Neurais Artificiais é a questão de seu treinamento. O que se propõe nesse trabalho é a implementação de um Neurocontrolador com treinamento totalmente on-line, dessa forma, todo o treinamento deve ser feito entre as amostragens, ou seja, durante os intervalos de amostragem.

A questão da velocidade de processamento influi também diretamente sobre a qualidade do controle proporcionado. Caso o treinamento da rede seja rápido (em cada iteração), ao final do período de amostragem, teremos os pesos mais próximos do valor ótimo. O que pode significar uma rede com capacidade de gerar sinais de controle que levem a saída da planta mais próxima ao sinal de referência.

A arquitetura utilizada nesse esquema de neurocontrole é a Perceptron de Múltiplas Camadas, dessa forma, a maioria dos processos de treinamento fazem uso do método do gradiente para o treinamento dos pesos. Uma alternativa para a melhoria dos processos de treinamento é substituir o método do gradiente por outros métodos de treinamento que fazem uso de métodos de otimização de derivada superior, como o método de Newton e Marquandt-Levenberg. O problema desses métodos é que eles realmente conduzem a uma convergência em um menor número de iterações, porém o tempo necessário a cada uma das iterações é consideravelmente maior, além de uma maior complexidade nos algoritmos.

A outra alternativa é o uso dos chamados algoritmos adaptativos, nos quais se faz o ajuste automático de alguns dos parâmetros da rede em função do comportamento do erro na saída da rede. Normalmente o parâmetro ajustado é o coeficiente de aprendizado. Fizemos testes nesse problema com três algo-

ritmos diferentes, o Eta-Adaptativo, o Delta-Bar-Delta [5] e o SuperSAB [6]. Destes, o que obteve os melhores resultados, para este problema, foi o Eta-Adaptativo. São os resultados obtidos por esse método e os detalhes de sua aplicação que agora apresentamos.

5.1 Eta-Adaptativo

O método é bastante simples, consiste basicamente em aumentar o η (coeficiente de aprendizado) caso o erro global na saída da rede esteja diminuindo e diminuí-lo caso o erro esteja aumentando. Caso o erro global esteja aumentando, significa que o treinamento da rede está levando os pesos para uma direção errada, diminuimos então o η para que este treinamento errado não comprometa o valor dos pesos. Caso o erro esteja diminuindo, isto significa que possivelmente o treinamento está realmente levando os pesos para uma região próxima a um mínimo, aumentamos então o η para que este mínimo seja atingido mais rapidamente.

A grande vantagem desse método é sua simplicidade, não faz uso de refinamentos como a derivada segunda ou um η para cada um dos pesos, como é comum em outros métodos baseados em estratégias locais. Ao contrário do que possa parecer, apesar da simplicidade do método, os resultados obtidos são excelentes. Abaixo apresentamos o algoritmo para a implementação do método.

```

Erro_2=Erro;

Se (Erro_2 ≤ 1,04*Erro_1)
    Se (Erro_2 > Erro_1)
        η = η;
    Senão
        η = 1.05 * η;
    Senão
        η = 0.7 * η;
FimSe
Erro_1= Erro_2;
    
```

Algoritmo 1: Algoritmo Eta-Adaptativo

As variáveis $Erro_1$ e $Erro_2$ apresentadas são variáveis auxiliares do algoritmo, enquanto que a variável $Erro$ é o próprio erro global na saída da rede.

Apesar de existirem outros métodos para a aceleração da convergência, optou-se pelo método eta-adaptativo devido à sua simplicidade e ao seu bom funcionamento na prática.

6. Resultados Obtidos

Nesta seção apresentaremos alguns detalhes da aplicação do neurocontrolador ao sistema de temperatura.

A planta em questão já foi descrita na seção 2. Trata-se, basicamente, de um sistema não-linear de 1ª ordem com atraso de transporte. O neuro-emulador é o responsável pelo aprendizado da dinâmica da planta. Todas as informações da planta necessárias ao bom desempenho do controlador são fornecidas pelo neuro-emulador.

Uma importante questão na implementação em tempo real de qualquer controlador neural é a questão da escolha do período de amostragem. Por se tratar de um sistema lento (constante de tempo de 30 s), podemos escolher um período de amostragem de 0,75 s.

As duas Redes Neurais que compõe o Neuro-emulador e o Neuro-controlador são constituídas de 2 neurônios na camada de entrada, 1 na de saída e apenas uma camada oculta, com cinco neurônios na mesma. As funções de ativação dos neurônios na camada oculta são funções sigmóides bipolares e as da camada de saída é uma função do tipo linear.

Nesse primeiro exemplo, vemos a saída da rede para uma referência degrau unitário de valor igual a 55 °C. Veja na figura 5 o comportamento da planta, juntamente com o do sinal de controle gerado pelo neuro-controlador (figura 6).

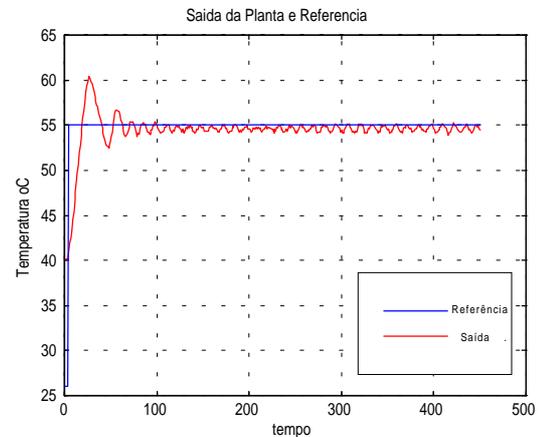


Figura 5: Saída da planta para uma referência degrau unitário.

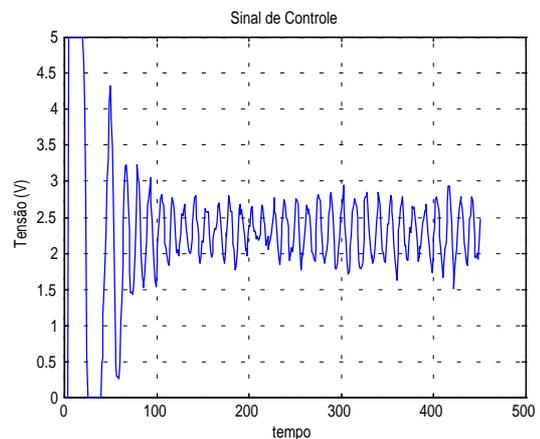


Figura 6: Sinal de controle gerado pelo neurocontrolador.

Na figura 7, apresentamos um gráfico que mostra o comportamento do erro, ou seja, da diferença, em módulo, entre a saída da planta e sua referência.

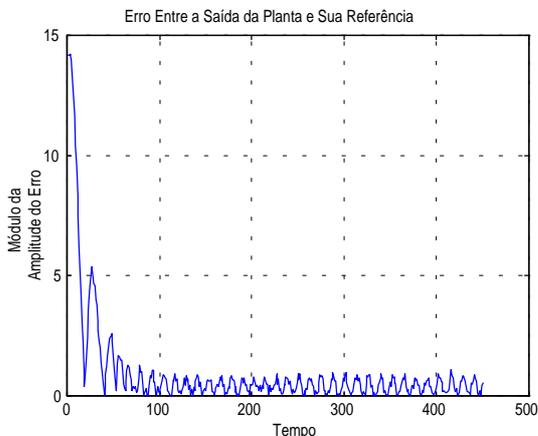


Figura 7: Módulo do erro (em graus) entre a saída da rede e o sinal de referência.

Pelos resultados, vemos que o desempenho do controlador é bastante satisfatório, visto que o máximo *overshoot* não chega a 10% do valor final de regime, o transitório não ultrapassa as 100 primeiras iterações. O desempenho final em regime pode se considerar bastante satisfatório, pois as oscilações não ultrapassam 1 °C. As oscilações presentes podem ser creditadas ao ruído presente na medida da temperatura e a variação na velocidade do ventilador.

Uma não-linearidade associada ao sistema é a relação quadrática entre a tensão de acionamento da resistência e a temperatura dissipada pela mesma. A própria resistência varia sua magnitude com o aumento da temperatura, inserindo outra não-linearidade ao sistema.

Qualquer oscilação da tensão que aciona o ventilador, provoca uma mudança no atraso de transporte da planta. Apesar disso, o comportamento na saída da planta é bom, o que ressalta também a capacidade de adaptatividade associada ao neurocontrolador.

A seguir apresentamos mais um exemplo, no qual variamos a referência do sinal. Observe a saída da planta e o seu sinal de controle.

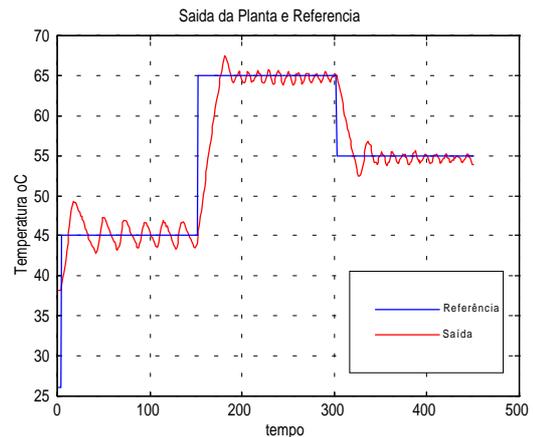


Figura 8: Saída da planta para uma referência variável.

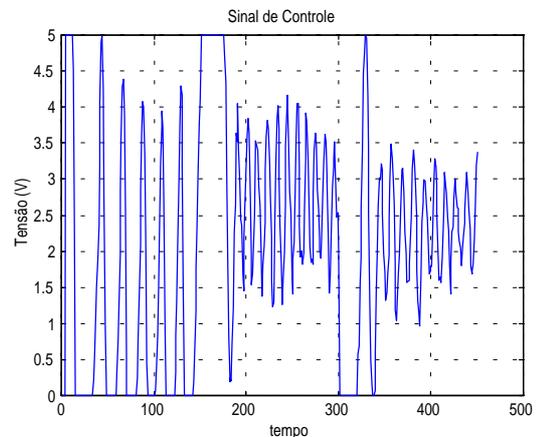


Figura 9: Sinal de controle gerado pelo neurocontrolador.

Apresentamos agora o gráfico do comportamento do erro entre a saída da planta e a referência.

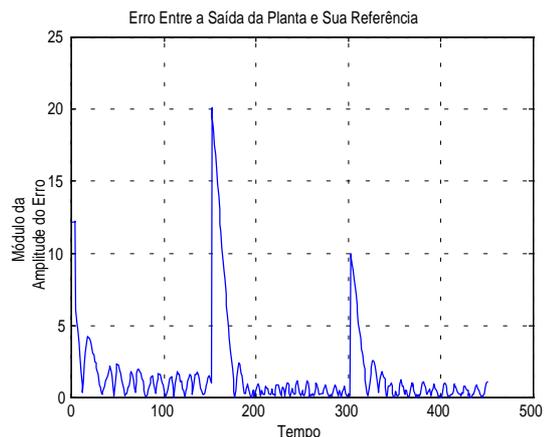


Figura 10: Gráfico do comportamento do erro (em graus) entre a saída da planta e sua referência.

Pelo que vemos pelos gráficos, o desempenho inicial do controlador não é muito bom, mas evolui com o passar do tempo em função de um maior treinamento da rede. Ao final do processo de ensaio na planta, o erro fica em torno de 1°C.

Aqui não são apresentados resultados de comparação com outros controladores, embora tenham sido

feitos testes com o controlador PID, o qual apresentou também um comportamento oscilatório, pois enfrentou os mesmos problemas relacionados à presença de ruído, além daqueles decorrentes de não-linearidades e a atraso de transporte variável.

7. Conclusões

Podemos ver nesse trabalho a capacidade do neurocontrolador aplicado em tempo real em uma planta de temperatura construída em nosso laboratório.

Os resultados práticos demonstram a boa capacidade do controle neural. A planta apresenta algumas não linearidades inerentes, como a relação quadrática entre a tensão na resistência e quantidade de calor fornecida pela mesma. A medição de temperatura apresenta ruído. Além disso, as variações de velocidade do ventilador podem ocasionar mudanças no atraso de transporte da planta. Mesmo assim, o Neurocontrolador conseguiu um bom desempenho.

Os neurocontroladores podem ser aplicados a uma grande classe de problemas, especialmente com o avanço da informática que permite o desenvolvimento de máquinas de maior capacidade de processamento. A maior capacidade de processamento permitirá uma ainda maior velocidade de treinamento e a aplicação dos neurocontroladores a plantas de dinâmica mais rápida.

Agradecimentos

A CAPES e Eletrobras, pelo apoio financeiro.

Referências Bibliográficas

- [1] Maitelli, A.L. e Gabriel, O.F. (1996). Um Esquema de Controle Adaptativo Neural com Treinamento "On-Line", *7.º Congresso Latinoamericano de Control Automatico – LACC, IFAC*, 2: 887-892. Buenos Aires. Argentina
- [2] Haykin, S. (1994). *Neural Networks*. Macmillan College Publishing Company. Ontario. Canada.
- [3] Hrycej, T. (1997). *Neurocontrol: Towards an Industrial Control Methodology*. John Wiley & Sons. Ulm. Alemanha.
- [4] Maitelli, A.L. e Rezende, J. A. D. (1998). Implementação em Tempo Real de um Esquema de Controle Neural On-Line em Plantas Físicas, *50.ª Reunião Anual da SBPC*, Sociedade Brasileira para o Progresso da Ciência, pp. 98. Natal. Rio Grande do Norte. Brasil.
- [5] Jacobs, R.A. (1988). Increased Rates of Convergence Through Learning Rate Adaptation. *Neural Networks*, 1:295-307.
- [6] Tollenaere T. (1990). SuperSAB: Fast Adaptive Backpropagation with Good Scaling Properties. *Neural Networks*, 3:561-573.