

## Implementação de um Controlador Preditivo Utilizando Redes Neurais Recorrentes e Algoritmos Genéticos

Carlos A. Claumann, Nestor Roqueiro, Ricardo A. F. Machado, Ariovaldo Bolzan  
Universidade Federal de Santa Catarina, Centro Tecnológico,  
Departamento de Engenharia Química e de Alimentos  
Cx.Postal 476 - CEP 88040-900 - Florianópolis - SC - Brasil  
Fone (+55) 48 331 9448 - Fax (+55) 48 331 9687  
E-mails: caio@enq.ufsc.br, nestor@enq.ufsc.br  
machado@enq.ufsc.br, abolzan@enq.ufsc.br

### Abstract

In this work a predictive controller was implemented for the control of a tank level. A recurrent network completely interconnected, trained by a genetic algorithm, was used as model for the process. The genetic algorithm was also used as an optimizer, in real time, of the controller's objective function.

### 1. Introdução

As redes neurais são amplamente utilizadas como modelos não lineares de processos, tanto para predição de comportamento de sistemas dinâmicos como para projeto de sistemas de controle. As redes mais usadas são as do tipo de *feedforward* multi-camadas treinadas com o método *backpropagation* [1]. No entanto, tem-se observado que as redes neurais recorrentes (*Recurrent Neural Networks* - RNNs) possuem capacidade de representação dinâmica superior [2], [3].

A maior diferença entre as RNNs e as redes *feedforward* está na topologia. As RNNs utilizam conexões *feedforward* (estáticas) e *feedback* (recorrências), enquanto que nas redes *feedforward* existem apenas conexões estáticas. A presença de recorrências transforma as RNNs em modelos com dinâmica intrínseca e, portanto, com possibilidade de identificar o comportamento dinâmico de sistemas não lineares.

Em geral, o treinamento de RNNs utilizando métodos tipo gradiente não pode ser realizado devido à realimentação das saídas. Por esta razão, técnicas provenientes da Inteligência Artificial tem recebido recentemente uma considerável atenção. Dentre estas técnicas podem-se ressaltar os algoritmos genéticos, (*Genetic Algorithms* - GAs). Os GAs, que são algoritmos com paralelismo intrínseco, estão sendo aplicados com sucesso onde outros métodos tradicionais falham.

Neste trabalho implementou-se um controlador preditivo para o controle de nível de um tanque constituído de duas seções, uma cônica e outra cilíndrica. Uma RNN completamente interconectada, treinada por um GA, foi utilizada como modelo do

processo. Uma demonstração da capacidade de representação dinâmica deste tipo de arquitetura pode ser encontrada no trabalho de [4]. O GA também foi utilizado para otimizar, em tempo real, a função objetivo do controlador. Os resultados conseguidos com o controlador preditivo, mostraram-se bastante superiores aos obtidos com o clássico PID, no controle de um processo com elevada não linearidade e presença de ruídos.

### 2. Descrição da RNN

Uma ilustração da RNN utilizada neste trabalho pode ser vista na figura 1. A RNN possui  $N$  neurônios (camada de processamento),  $M$  entradas externas e  $O$  saídas, onde  $X(n)$  representa um vetor ( $M \times 1$ ) de entradas externas aplicadas à rede no instante de tempo  $n$ , e  $Y(n+1)$  é o correspondente vetor com ( $N \times 1$ ) saídas dos neurônios da camada de processamento no instante de tempo  $n+1$ .

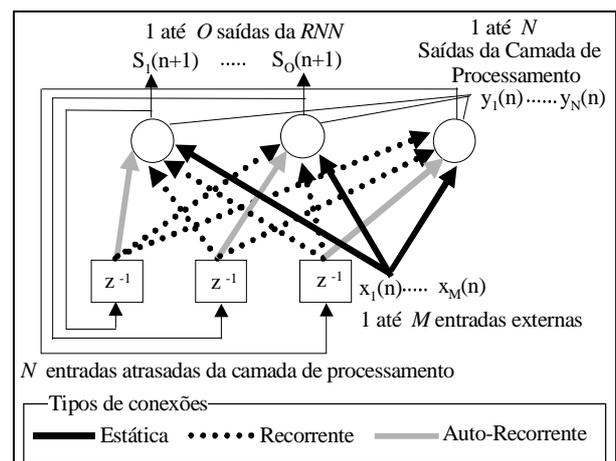


Figura 1 - Arquitetura de uma RNN completamente interconectada

As entradas da rede são formadas pelas saídas atrasadas  $Y(n)$  e as entradas externas  $X(n)$  para cada intervalo de tempo, definindo um vetor de dimensão  $(M+N) \times 1$ . A saída da RNN  $S(n)$  é formada por um vetor de  $O$  elementos. A RNN, portanto, possui um

total de  $M \times N$  conexões estáticas e  $N^2$  conexões recorrentes; das quais  $N$  são auto-recorrentes.

Um método de treinamento para a RNN mostrada na figura 1 pode ser encontrado em [5] e corresponde a uma modificação do método backpropagation.

### 3. Algoritmo Genético

Algoritmos genéticos são algoritmos de pesquisa e otimização baseados no mecanismo de seleção natural e adaptação. Estes algoritmos operam com um conjunto de possíveis soluções (indivíduos), denominado “população”, que são combinados, através de uma troca de informações, de modo a encontrar melhores soluções do que as correspondentes na população original. A otimização utilizando um GA deve seguir os seguintes passos:

1) Representação das variáveis do problema na forma de cromossomos. A codificação dos pesos da RNN em um cromossomo pode ser vista na figura 2. Neste caso um cromossomo é composto por pesos estáticos ( $W_{ij}^{est}$ ), pesos dinâmicos ( $W_{ij}^{din}$ ) e um bias por neurônio ( $B_i$ ).

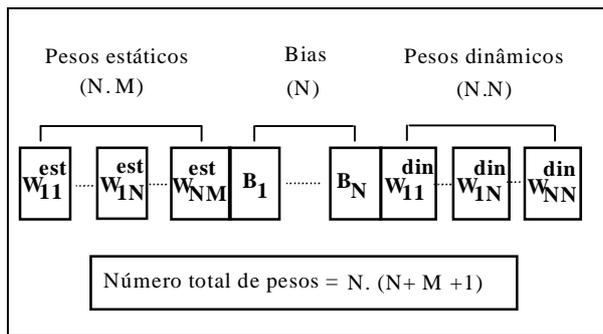


Figura 2 -RNN codificada na forma de cromossomo

2) Definição de uma função objetivo a partir da qual pode-se avaliar o desempenho dos cromossomos. A função objetivo utilizada no treinamento da RNN é o erro médio quadrático entre a saída predita pela RNN e saída real.

3) Definição de um conjunto de operadores que possam ser aplicados aos cromossomos de modo a alterar seus genes e melhorar seu desempenho. Os operadores utilizados no GA implementado neste trabalho são:

**Mutação Exponencial:** Este operador adiciona, de forma diferenciada, pequenas quantidades a cada gene, sendo que, o valor absoluto e o sinal destas são calculados aleatoriamente [6].

**Seleção:** A seleção empregada é a Rolleta [6].

**Crossover Direcional:** Dados dois pais escolhidos

por seleção, este operador verifica qual deles resulta no menor erro quadrático de treinamento. Em seguida, geram-se dois filhos na vizinhança do melhor pai [4].

**Elitismo:** O operador compara cada novo indivíduo gerado com o pior indivíduo da população, sendo que, o melhor é mantido

**Escalação:** Neste trabalho é adotado o escalação sigma [6], que utiliza a média e o desvio padrão dos valores da função objetivo de todos os indivíduos da população.

4) Definição dos parâmetros população (tamanho da população, taxa de mutação e crossover do GA). Nas simulações foram testadas várias combinações para os valores dos parâmetros do GA, os melhores resultados foram conseguidos com:

**Taxa de mutação** = 0.01

**Taxa de crossover** = 1

**População** = 300 indivíduos

5) Inicialização do processo de busca definindo uma população inicial. Todos os indivíduos da população recebem valores aleatórios no intervalo [-6,6] (experimentalmente os melhores resultados foram conseguidos com esta faixa de trabalho para os pesos). Os valores dos pesos foram distribuídos uniformemente no intervalo dado.

6) Finalização do processo de busca. Finaliza-se a busca quando a avaliação da função objetivo dos indivíduos for, em média, menor que uma tolerância ou, quando é atingido um número máximo de gerações.

### 4. Descrição do Tanque de Nível

O tanque cônico-cilíndrico faz parte de um conjunto de sistemas do Laboratório de Controle de Processos (LCP/UFSC). (Ver figura 3)

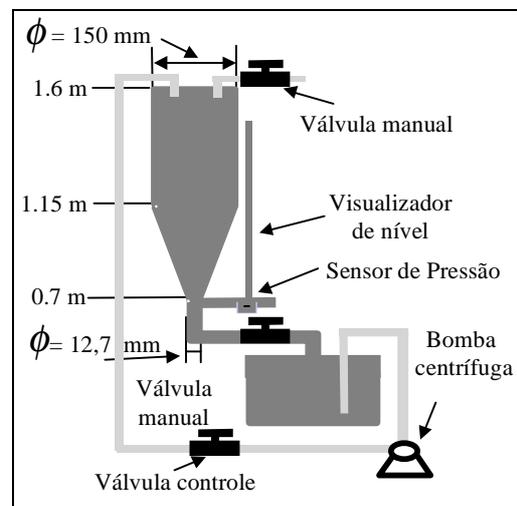


Figura 3- Sistema de Tanque Cônico-Cilíndrico

O tanque é constituído de duas seções de comprimento aproximadamente equivalentes (uma cônica e outra cilíndrica). A seção cônica começa em 0.7 m e acaba em 1.15m. A cilíndrica inicia em 1.15 e termina em 1.6m. O tanque é alimentado pela rede hidráulica do laboratório, constituída de tubulações em PVC de 12,7 mm de diâmetro, uma bomba centrífuga de 0.25 cv de potência, uma válvula de controle com acionamento pneumático para tubulações 19,05 mm de diâmetro e válvulas manuais.

O tanque cônico-cilíndrico apresenta a não linearidade de ganho inerente à mudança de geometria e há um forte ruído na medição do nível. Outras não linearidades estão presentes na válvula de controle, que é do tipo igual percentagem, ar-abre, falha-fecha e possui histerese.

## 5. Identificação

Dois grupos de dados foram utilizados: um para treinamento da RNN e outro para teste. Os grupos de dados foram coletados off-line e o tempo de amostragem utilizado foi de 5 s. Os grupos de dados foram obtidos efetuando perturbações aleatórias entre 1 a 5 Volts, no sinal enviado a válvula de controle. Estes limites correspondem aos estados de completamente fechado e aberto da válvula de controle. As perturbações efetuadas produziram variações no nível entre 0.75 a 1.55 metros.

Para determinar a melhor estrutura da RNN efetuou-se um estudo variando o número de neurônios escondidos e o número de entradas. A determinação melhor estrutura de RNN foi baseada no erro médio quadrático de treinamento, sendo escolhida a menor estrutura a partir da qual o erro médio quadrático de aproximação pouco variava. A melhor RNN obtida no treinamento foi validada em relação ao grupo de dados de teste. Na figura 4 pode-se observar a saída real e a predição da RNN. O erro médio quadrático de predição do teste foi de 0.00034.

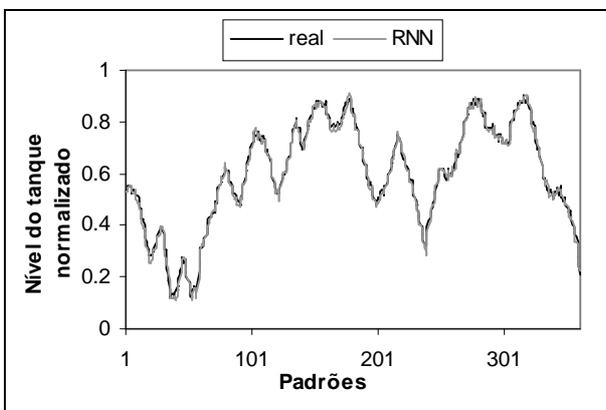


Figura 4- Comparação entre o nível do tanque normalizado e o predito pela RNN

## 6. Controle

Neste trabalho foi utilizada uma abordagem clássica de controle preditivo, [7], para comprovar a viabilidade do uso de RNNs treinadas por GAs. Nesta abordagem calcula-se o conjunto de futuras ações de controle otimizando um índice de desempenho de forma a manter o processo o mais próximo possível de uma trajetória de referência.

O critério de otimização mais utilizado em controle preditivo é o erro quadrático entre a saída predita e a estipulada pela trajetória de referência. Em muitos casos, pode-se também incluir o esforço de controle na função objetivo, [8]. O critério de otimização é mostrado na equação 1 e a trajetória de referência na equação 2.

$$J(N_1, N_2, N_u) = \sum_{i=N_1}^{N_2} \delta(i) \left[ \bar{y}(k+i|k) - y_{ref}(k+i) \right]^2 + \sum_{j=1}^L \sum_{i=1}^{N_u} \lambda(j,i) \left[ \Delta U_j(k+i-1) \right]^2 \quad (1)$$

$$y_{ref}(k+i) = \alpha y_{ref}(k+i-1) + (1-\alpha)y_{sp} \quad (2)$$

Onde:

Nu: horizonte de controle;

N1 e N2: horizonte inicial e final de predição;

L: número de variáveis manipuladas;

$\bar{y}$ : valor predito para a variável controlada;

y<sub>sp</sub>: ponto de operação;

y<sub>ref</sub>: trajetória de referência;

$\delta(i)$ ,  $\lambda(i,j)$ : fatores de penalização;

$\alpha$ : Fator de ajuste da trajetória de referência;

Para um controlador baseado em uma RNN, a otimização da função objetivo (equação 1) por métodos do tipo gradiente é uma tarefa difícil. A proposta deste trabalho é utilizar um GA e evitar, desta forma, o cálculo de gradientes. Neste caso deve-se codificar o vetor de ações de controle presentes e futuras na forma de um cromossomo. A figura 5 mostra um cromossomo formado pelo vetor de ações de controle presentes e futuras.

Na figura 5, o número de incógnitas da função objetivo do controlador é igual ao horizonte de controle Nu vezes o número de variáveis manipuladas L. Neste trabalho L é igual a 1(nível do tanque).

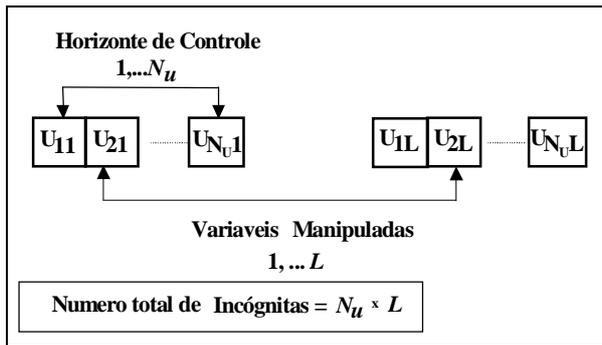


Figura 5-Vetor de ações de controle escrito na forma de um cromossomo

Na otimização da função objetivo do controlador foi utilizada uma população pequena (30 indivíduos), dado que, para aplicações em tempo real, é necessária uma rápida convergência. Os demais parâmetros do GA foram mantidos inalterados.

O controlador preditivo foi comparado a um PID em relação à transições no ponto de operação e às ações de controle tomadas. Os parâmetros dos controladores preditivo e do PID foram ajustados experimentalmente de forma a maximizar o desempenho de cada um. Os valores dos parâmetros são mostrados abaixo. Em todos os casos o intervalo de amostragem foi de 5s.

Parâmetros do Controlador Preditivo

$$\lambda = 0.025$$

$$\alpha = 0.65$$

$N_u = 4$  intervalos de amostragem.

$N_1 = 1$  intervalo de amostragem.

$N_2 = 4$  intervalos de amostragem.

Parâmetros do Controlador PID

$$K_c = 18$$

$$T_i = 30$$

$$T_d = 0.5$$

## 7. Resultados

O desempenho do controlador preditivo foi testado para perturbações no ponto de operação e comparado com um controlador PID. Nas figuras 6 e 7 são apresentadas as ações de controle do controlador baseado na RNN e do PID respectivamente, e nas figuras 8 e 9 é possível observar as transições de ponto de operação para os dois controladores na faixa de 0.8 a 1.45 m.

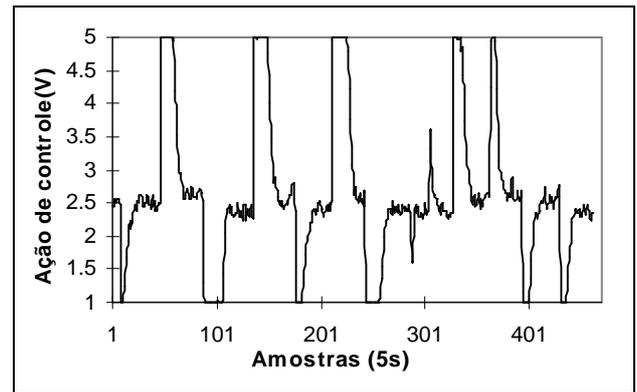


Figura 6- Ações de controle tomadas pelo controlador baseado na RNN

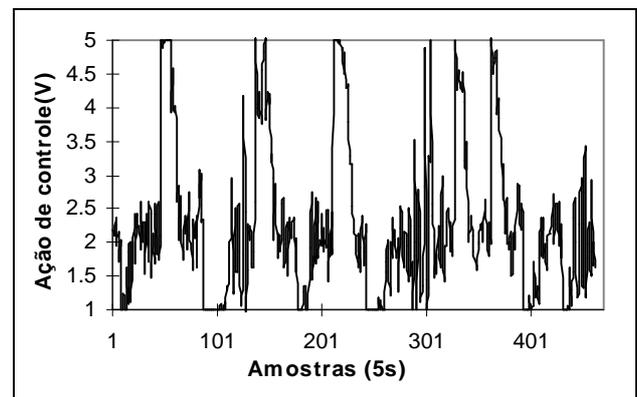


Figura 7- Ações de controle tomadas pelo PID

Pode-se observar nas figuras 6 a 9 que o controlador baseado na RNN foi mais rápido, não apresentou *overshoot* ou *off-set* e utilizou ações de controle mais suaves. O controlador PID, por outro lado, apresentou *overshoot* nas transições em que o valor do ponto de operação foi aumentado e variações significativas das ações controle. A diferença de desempenho é mais acentuada no início da seção cônica, onde a dinâmica do sistema é mais rápida, obrigando o controlador a tomar ações de controle precisas para produzir transições eficientes de ponto de operação.

Na otimização da função objetivo do controlador observou-se que o GA convergiu, em média, em um tempo de 500ms (um décimo do tempo de amostragem utilizado), portanto se comprovou a viabilidade da aplicação do GA no controle de nível do tanque cônico-cilíndrico.

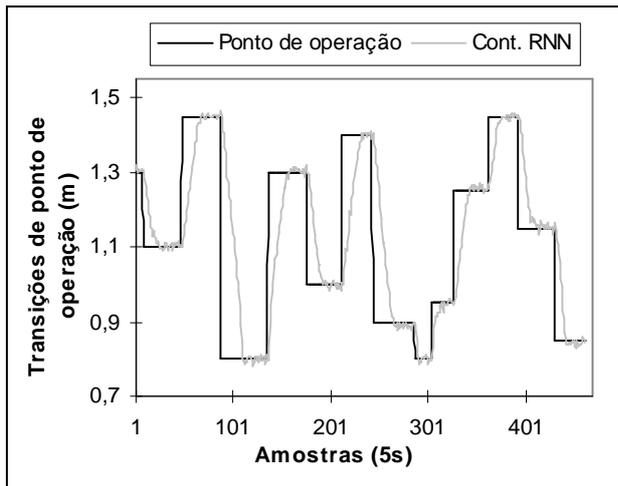


Figura 8- Transições de ponto de operação efetuadas pelo controlador baseado na RNN

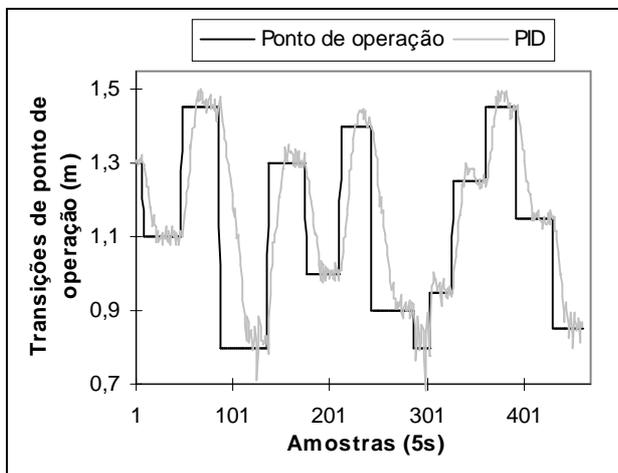


Figura 9- Transições de ponto de operação efetuadas pelo PID

## 8. Conclusões

Um GA foi empregado com sucesso no treinamento de uma RNN para identificação de um tanque de nível.

O modelo neural obtido foi utilizado em um controlador preditivo. O controlador foi testado no controle de nível de um tanque cônico-cilíndrico, obtendo melhores resultados que os atingidos com um PID.

O GA foi suficientemente rápido para otimização da função custo do controlador preditivo em tempo real.

Os bons resultados obtidos neste trabalho incentivam a aplicação de GAs e RNNs no controle e identificação de processos mais complexos.

## Referências

[1] Mazzuco, M. M.- "Implementação de um Controlador Preditivo Baseado em um Modelo Neural Associado a

um Sistema Especialista" - Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia Química, UFSC, Florianópolis, SC, Brasil, 1997.

- [2] Burrows, T. L. & Niranjana. - The Use of Feedforward and Recurrent Neural Networks for System Identification, Cambridge University Engineering Department, Technical Report CUED/F-INFENG/TR 158, December, 1993.
- [3] Henson, M. A. & Seborg, D. E. - Nonlinear Process Control, Prantice Hall, 1997.
- [4] Claumann, C. A. - "Modelagem e controle de processos não lineares: Uma aplicação de algoritmos genéticos no treinamento de redes neurais recorrentes" - Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia Química, UFSC, Florianópolis, SC, Brasil, 1999.
- [5] Haikin, S. - Neural Networks – A Comprehensive Foundation, Ontario: IEEE Computer, Society Press, 1994.
- [6] Goldberg, D. - "Genetics Algorithms in Search, Optimization and Machine Learning" - Addison-Wesley, Redwood City, CA, 1988.
- [7] Camacho, E. F. & Bordons, C. - Model Predictive Control in the Process Industry, 1994.
- [8] Cancelier, A. - "Controle Preditivo de Reatores em Batelada" - Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia Química, UFSC, Florianópolis, SC, Brasil, 1998.